T.C.

ISTANBUL AYDIN UNIVERSITY

INSTITUTE OF NATURAL AND APPLIED SCIENCES

FEATURE SELECTION FOR LEAF DISEASES DETECTION USING OPTIMIZATION ALGORITHMS

THESIS

Lamyae GOUBRAIM

Department of Electrical and Electronics Engineering

Electrical and Electronics Engineering Program

Thesis Advisor: Assist. Prof. Dr. Necip Gökhan KASAPOĞLU

June, 2019

T.C.

ISTANBUL AYDIN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

FEATURE SELECTION FOR LEAF DISEASES DETECTION USING OPTIMIZATION ALGORITHMS

THESIS

Lamyae GOUBRAIM

(Y1613.300021)

Department of Electrical and Electronics Engineering

Electrical and Electronics Engineering Program

Thesis Advisor: Assist. Prof. Dr. Necip Gökhan KASAPOĞLU

June, 2019

ONAY FORMU

DECLARATION

I hereby declare that all information in this thesis document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results, which are not original to this thesis.

Eng. Lamyae GOUBRAIM

FOREWORD

After thanks to Allah our creator, I would like to express my gratitude to my advisor, Dr. Necip Gökhan Kasapoğlu. I thank him for framing, guiding, helping and advising me.

I extend my sincere thanks to all the teachers, speakers and all the people who by their words, their writings, their advice and their critics guided my thoughts and agreed to meet me and answer my questions during my research.

I would like to thank my dear parents, who have always been there for me. I thank my sister, and my brothers for their encouragement.

Finally, I would like to thank my friends who have always been there for me. Their unconditional support and encouragement have been of great help.

June, 2019   **Eng. Lamyae GOUBRAIM**

TABLE OF CONTENT

x

ABBREVIATIONS

ANN   :Artificial Neural Network
ICA   **:**Imperialist Competitive Algorithm
PSO   **:**Particle Swarm Optimization
PCA   :Principal Component Analysis
GA   :Genetic Algorithm
KNN   :K-nearest neighbor
SVM   :Support Vector Machine
SGDM   :Gray-level Dependence Matrix
RGB   :Red Green Blue
HSV   :Hue Saturation and Value channel
PRR   :Phytophthora Root Rot
CLR   :Coffee Leaf Rust
CWD   :Coffee Wilt Disease
CBD   :Coffee Berry Disease
BPNN   :Back Propagation Neural Network
CNN   :Deep Convolution Networks
MSE   :Mean Square Error

LIST OF FIGURES

LIST OF TABLES

FEATURE SELECTION FOR LEAF DISEASES DETECTION USING OPTIMIZATION ALGORITHMS

ABSTRACT

Modern techniques of computer science and machine learning become more and more important and useful in recent years. The cutting-edge techniques of artificial intelligence assisted humanity in different aspects of life, such as health-care, military and agricultural fields. However, a current solution of identifying leaves diseases totally based on visual inspection of farmers and agricultural engineers. Because this is a very time consuming manual method, its cost is also high as it requires a lot of personnel and risks a lot of plants. This work proposes a novel solution to identify the location and type of diseases on plant leaves, using imperialist competitive algorithm (ICA) for feature selection, and an efficient artificial neural network (ANN) algorithm for recognition. Moreover the comparison between two meta-heuristic optimization algorithms namely imperialist competitive algorithm (ICA) and particle swarm optimization (PSO) is given to demonstrate the effectiveness of the ICA.

Keywords: *Artificial neural network (ANN), feature selection, imperialist competitive algorithm (ICA), meta-heuristic, particle swarm optimization (PSO), plant leaf disease recognition.*

YAYILMACI REKABETÇİ ALGORİTMASINI KULLANARAK BİTKİ YAPRAK HASTALIKLARININ BELİRLENMESİ

ÖZET

Modern bilgisayar bilimi ve makine öğrenmesi teknikleri son yıllarda giderek daha önemli ve kullanışlı hale gelmiştir. Yapay zekanın en ileri teknikleri, sağlık, askeri ve tarım alanları gibi yaşamın farklı yönlerinde insanlığa yardımcı olmaktadır. Bununla birlikte, bitki yaprak hastalıklarının tespitinde mevcut çözümler tamamen çiftçilerin ve ziraat mühendislerinin görsel incelemesine dayanmaktadır. Bu çok yavaş bir yöntem olduğundan, maliyeti çok yüksektir, çünkü çok sayıda personel gerektirir ve çok sayıda bitkinin hastalanması riskini taşır. Bu tezde, bitki yaprakları üzerindeki hastalıkların yerini ve türünü tanımlamak için, öznitelik seçimi için yayılımcı rekabet algoritması (YRA) ve tanıma için verimli bir yapay sinir ağı (YSA) algoritması kullanarak yeni bir çözüm önerilmektedir. Ayrıca, yayılımcı rekabet algoritması (YRA) ve parçacık sürüsü optimizasyonu (PSO) gibi iki sezgisel optimizasyon algoritması arasındaki karşılaştırma, YRA' nın etkinliğini göstermek için verilmiştir.

Anahtar Kelimeler: *Yapay sinir ağları (YSA), öznitelik seçimi, yayılımcı rekabetçi algoritması (YRA), sezgisel yöntemler, parçacık sürüsü optimizasyonu (PSO), bitki yaprak hastalıklarının tanınması.*

1. INTRODUCTION

State-of-the-art automated techniques can be used to achieve more advanced and accurate solutions that can reduce costs to detect leaves diseases in a shorter time. However, in some previous studies some methods proposed to solve the same problem, but in these studies proper features have to be used to achieve more accurate results have been ignored. In this thesis imperialist competitive algorithm (ICA) is used with cascade feed forward neural network to extract the most suitable features to be used in detecting diseases in leaves. Diseases considered in this study are alternaria alternate, anthracnose, bacterial blight, septoria brown spot and cercospora.

Alternaria alternata is a fungus or mushroom which has been recorded causing leaf spot and other diseases on over 380 host species of plant. It is an opportunistic pathogen on numerous hosts causing leaf spots, roots, and blights on many plant elements [1]. Moreover, anthracnose is a fungal disease that develops when the humidity is too high, especially in spring and autumn, but also in summer [2].

Bacterial blight is the most common soybean disease in cold, wet weather. This disease usually occurs at low levels that do not lead to a loss of yield. The bacterial blight may be mistaken for the septoria brown spot. An error can be made in the distinction between bacterial blight and septoria brown spot. These two diseases can be distinguished by the presence of a halo around bacterial lesions. Same plant can be exposed to both diseases at the same time, However bacterial blight are more common in young leaves, as long as brown spots usually occur on older leaves and less leaves. [3].

On the other hand, cercospora fruit spot is a common disease of citrus fruits but it also affects many other crops. The disease is fungal and survives on any affected fruit in soil from the previous season [4].

These diseases can infect several types of plants, such as bean, beetroot, capsicum, okra, silver beet, watercress, carrot, avocado and coffee in different

regions around the world, especially in South East Asia and Africa. These diseases are usually detected manually by farmers or agricultural engineers who survey the fields. However, due to the advanced technologies of computer science and machine learning, the cutting-edge methods of image processing are becoming widely used to replace traditional methods efficiently.

1.1 Machine Learning and Artificial Intelligence

Machine Learning is the science of getting a computer to act based on a model built during the training. It has two sub-fields and artificial intelligence is the most important one. Machine Learning is the simulation of human intelligence by means of machines. Those approaches consist of learning (the acquisition of information and rules for the use of the information), reasoning (the use of the rules to reach approximate or definite conclusions) and self-correction. Moreover, artificial intelligence has mains applications such as:

- Prediction is the ability to learn and predict future events, without being explicitly programmed.

- Sample recognition is a branch of machine learning that emphasizes the recognition of data patterns or data regularities in a given scenario.

- Clustering: is a machine learning technique that involves the grouping of data factors. Given a set of statistics factors, we can use a clustering algorithm to categorize each data point into a specific group.

In artificial intelligence several algorithms are used such as fuzzy logics a form of many-valued logic in which the real values of variables may be any real number between 0 and 1. It is hired to deal with the concept of partial reality, where the real value may range between completely true and completely false. Additionally, artificial neural network popularly known as neural community is a computational model based on the structure and functions of biological neural.

1.2 Problem Definition

There are several types of diseases affect plants and their leaves. However, traditional methods that are based on individual's observations are accurate, but

may take long time. In addition to this as in large fields where there are huge range of plants or trees, this traditional method may take very long time which can cause the plants to reach a worst situation before the disease being detected. Currently, computer science and machine learning techniques are used in this field and these techniques supply promising results. Nevertheless, the existing solutions lack the accuracy and speed of the detection procedure, which is usually due to the wide variety of features available at the back of the plant leaf image. Additionally, the previous works may not be able to identify the most suitable feature to be used, which reflects on the overall system performance. Therefore, it is important to discover a solution that provides the processing algorithm with suitable features, to have accurate disease detection within relatively shorter time.

1.3 Previous works

In this section, we review some previous works related to the plant leaf disease recognition in order to outline the performance of the algorithms. Singh and Misra [5] have tackled the problem of diseased plants and the delay usually experienced before imparting proper medication using traditional techniques. They have proposed an algorithm to provide proper segmentation for the leaf image and automatic detection for plant leaf disease. They have focused on one disease which is the little leaf in pine trees. They have conducted their research on samples collected from the United States. In order to implement their proposed solution, they have used genetic algorithm (GA) for image segmentation and made comparison between two methods which are K-means and support vector machine (SVM). They have extracted features using gray level co-occurrence matrix. In this paper smoothing filter is implemented as in pre-processing stage and to increase the contrast image enhancement. A digital camera is used for the image capturing. They have concluded that using genetic algorithm reduces the performance of this work. Max 15 images per sample were used in this study may be not sufficient to build a model and making performance evaluation.

Sabanb and Bayramin proposed a hybrid neural network for segmentation of leaf images [6]. Automatic segmentation method is proposed using four unique

color components. In order to implement their proposed method, various illumination conditions is used for image segmentation and the image is converted into RGB, HSV, XYZ and YIQ channels. After the transformation of the color components, B, S, Z and I components are used to train hybrid neural network and color features are extracted from each color region. Trained artificial neural network using gray wolf algorithm on the selected features was obtained. They have reported outperformed results according to sensitivity, specificity and accuracy.

Khirade and Patil handled the problem of monitoring plant diseases [7]. They proposed a method for detecting plant diseases using their leaves images. They focused on some specific disease such as powdery mildew yellow rust and aphids diseases. They have collected their images using a digital camera. In their proposed solution they have used the boundary and spot detection to identify infected parts of a leaf. They employed co-occurrence texture futures in K-means clustering algorithm for image segmentation. In this work the classification is applied using artificial neural network (ANN) with back propagation learning. Clipping smoothing and enhancement are applied as preprocessing step to remove noise in images.

Image processing is used for detection of diseases in cotton leaves [8]. In this work principal component analysis (PCA) and K-nearest neighbor (KNN) classifier are used in order to identify diseases. In this study the aim is to recognize diseases such as blight, leaf nacrosis, gray mildew, alternaria and magnesium deficiency. The most substantial capabilities from 110 samples were extracted and compressed to keep only the significant information. The accuracy of the proposed PCA-KNN based classifier is reported as 95% in this work.

In [9], the authors have tackled the problem of distinguishing plant leaves from their environment. Their technique is based on leaf image classification using deep convolution networks (CNN). The model built in this study is able to recognize thirteen different types of plant diseases such as powdery mildew, taphrina de formans, pear, erwinia amylovora. All the images used in the work are collected from the internet. The precision of the proposed method were reported between 91% and 98%.

The problem of diseases plants is the subject of the study in [10]. In this study neural network with back propagation (NN-BP) learning, k-way clustering and spatial graylevel dependence matrix (SGDM) have been used to analyses healthy and diseased plant leaves. In order to implement proposed scheme k-method clustering algorithm are applied for image segmentation and the masked cells inside the boundaries of the infected clusters are removed. The infected cluster is converted from RGB to HSI space. In this study neural networks (NN) have been used for recognition. Total 118 diseased leaf samples were used in the experiments. Proposed method is reported good potential with an ability to detect plant leaf diseases with some limitations.

Yahya et al.[11] studied the problem of detecting plant diseases. They have proposed a spectroscopy technique that has grown to be one of the most available noninvasive methods utilized in detecting plant diseases. They collected the images using digital camera to enhance the accuracy of their work. They have centered on a lot of diseases such as yellow rust, bacterial canker, downy mildew, apple scab, they have tested the 3 major categories for noninvasive monitoring of plant diseases which might be the visible and infrared spectroscopy (VIS/IR) technique. These categories are impedance spectroscopy (IS)and fluorescence spectroscopy (FS) techniques. They have concluded that spectroscopy strategies have the potential to be applied on plants, as a noninvasive disorder detecting tool.

In [12] the authors have tackled the problem of Phytophthora root rot (PRR) which infects the roots of avocado trees. In their paper they proposed two image analysis methods that can serve as surrogates to the visible assessment of canopy decline in large avocado orchards. These methods are combination of Canny edge detection and

Otsu"s method. Coinciding with the on-floor measure of cover porosity, they have used a total of 80 bushes to gather samples from three avocado blocks. In this study a smart phone camera was used to collect RGB color images.

Identification of coffee plant diseases that attack coffee plants using hybrid approaches of image processing and decision tree has been introduced in [13]. The authors have focused on the three foremost of diseases such as coffee leaf rust (CLR), coffee berry disease (CBD) and coffee wilt disease (CWD). They

have proposed back propagation artificial neural network (BPNN) as technique to recognize diseases. In their work they have used K-means clustering for image segmentation within the first stage of their paper coffee plant diseases are given as input to the system. The second step is that pre-processing of images, such as median filtering is used in this step for reducing noise. They have reported 94.5% of accuracy in their work.

Sivakamasundari and Seenivasagam have worked to provide a fast and cheap solution for detecting apple leaf diseases at the earlier stage. The authors have proposed a system aims to identify a number of the diseases in apple leaf based on color and texture feature. The diseases chosen are alternaria, apple scab, and cedar rust. In order to put in force their work; they have used K-means clustering for image segmentation; gray-level co-occurrence matrix (GLCM) or grey level spatial dependence matrix is used for texture evaluation. They have concluded that the accuracy in their paintings according to texture and color future is high therefore in the future the same system can be used to test other kinds of plants.

JUse of plant models in deep learning is proposed in [15]. The authors have proposed a new technique for augmenting plant phenotyping datasets using rendered images of synthetic plants by using computer-generated model of the arabidopsis rosette to enhancing leaf counting performance with convolutional neural networks (CNNs) and they have mainly focused on rosette plants.

Sunny and Gandhi have tackled the problem of citrus plants such as lemon which are mainly affected by citrus canker disease which affects the fruit production of the plants [16]. They have proposed an image processing technique to detect plant leaf diseases from digital images. The proposed method involves stages using contrast limited adaptive histogram equalization (CLAHE) enhancement and SVM classifiers to improve the clarity of leaf images. They have mainly focused on citrus canker disease. Images are captured by high resolution digital camera and they have used 100 samples in their work. In this approach, leave image features include texture, color, shape are taken into consideration for disease detection. They have concluded that the pre-processing step with CLAHE enhancement achieves efficient result integrated

with SVM classifiers. The SVM classifiers achieve the good prediction rate for differentiating the canker disease.

1.4 Motivation

The aim of this thesis is to select the most distinctive features that increase the recognition accuracy of the leaf diseases. An enhanced technique is proposed based on optimization algorithm to accurately discover the most suitable features to be extracted in the fields of diseases image classification. In this thesis to identify the most efficient algorithm to be used in selecting the disease features in leaf images is implemented and an approach that provides an enhanced detection accuracy and recognition quality is proposed. In order to reach thesis aims this work tries to find solutions for research questions below.

Q1- Is it possible to use an optimization algorithm to identify the most optimal features to be used by ANN for the classification process?

Q2- Does the use of an optimization algorithm, as a feature selector for the ANN, improves the accuracy of leaf diseases detection?

2. PRE-PROCESSING STEPS:

2.1 Feature extraction

Features are defined as a function of one or more measurements, each of which specifies some quantifiable property of an object. There are various features which are classified into:

- General features: Independent features such as color, texture , and shape.

According to the abstraction level, they can also be divided into:
- Pixel-level features: Features calculated at each pixel, such as color.

- Global features: Calculated over the entire image or on a regular sub-area of an image.

- Domain-specific features: Applying dependent features such as human faces, fingerprints, and conceptual ones.

- Local features: They are calculated on the results of the subdivision of the image segmentation or edge detection.

All features can be roughly classified into low-level and high-level features. Lowlevel features can be extracted directly from the original images, while extraction of high-level features depends on low-level features.

In this work gray level co-occurrence matrix (GLCM) feature extraction method is used to extract nine features after converting the segmented image to gray level. A Matlab code of the method is given below:

GrayImg=rgb2gray(ImgSeg);

GrayCooccurance=graycomatrix(GrayImg);

ImageFeatures=graycoprops(GrayCooccurance,'Contrast   Correlation   Energy Homogeneity');

Asymmetry=[asymmetry,skewness(double(ImgSeg(:)))];

Avg=[Avg,mean2(ImgSeg)];

ISTD=[ISTD,std2(ImgSeg)];

Shaprness=[Shaprness,kurtosis(double(ImgSeg(:)))];

RMS=[RMS,mean2(rms(ImgSeg))];

ImgVar=[ImgVar,mean2(var(double(ImgSeg(:))))];

ImageE=[ImageE, entropy(ImgSeg)];

ImgSummation=[ImgSummation,sum(double(ImgSeg(:)))];
ImageSmooth=[ImageSmooth,1-(1/(1+ImgSummation(n)))];

The extracted features for this work are as follows:

- **Texture features:**

RMS: The root mean square is a measure of the magnitude of a set of numbers. It gives a sense for the typical size of the numbers [16].

Image Smoothing: Smoothing is often used to reduce noise within an image or to produce a less pixelated image. The most of the smoothing methods are based on low pass filters [17].

Skewness: Skewness is a measure of the asymmetry of the gray levels around the sample mean. If skewness is negative, the data are spread out more to the left of the mean than to the right. If skewness is positive, the data are spread out more to the right [18].

Kurtosis: Kurtosis is a measure of how outlier-prone a distribution is. It describes the shape of the tail of the histogram [19].

Entropy: Entropy of the image is the measure of randomness of the image gray levels [20].

- **Color features:**

Standard deviation ISTD: Standard deviation of the intensity values (ISTD) is calculated for this color feature as follows: [21]:

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (xi - \mu)^2}$$

$$\textbf{(2.1)}$$

where x_i is intensity value for pixel i and μ is the mean value of an image or subimage.

Average: The mean gives the average gray level of each region and it is useful only as a rough idea of intensity not really texture [22].

$$A = \frac{1}{n} * \sum_{i=1}^{n} xi$$

$$(2.2)$$

where A is average or arithmetic mean., n is the number of samples and x_i is pixel intensity value for pixel i..

Image Variance: The variance gives the amount of gray level fluctuations from the mean gray level value [23].

$$\sigma^2 = \frac{\Sigma(X - \mu)^2}{N}$$

$$(2.3)$$

Image Sum: Image sum is to take the sum of the intensities of the pixels in an image.

2.2 Pre-processing

Generally to create a neural network in Matlab, the network is composed of n inputs, n outputs and a number of hidden layers.

The first step is to set the input and output nodes and the network is designed as follow, setting the number of hidden neurons and outputs once the activations functions are also cited the training function is defined with some training parameters such as number of epochs, validation checks…etc, Furthermore once the network is trained in the training process then all the performance measures

such as the training performance, are given. Moreover the execution time and the training process to reach the goal performance are also given by neural network.

In our case, our aim is to identify some of the diseases in plant leaves based on selected features by imperialist competitive algorithm (ICA) and Particle Swarm Optimization (PSO). The code is written in Matlab.

Alternaria Alterana Anthracnose   Bacterial   Bright Cercospora   Leaf
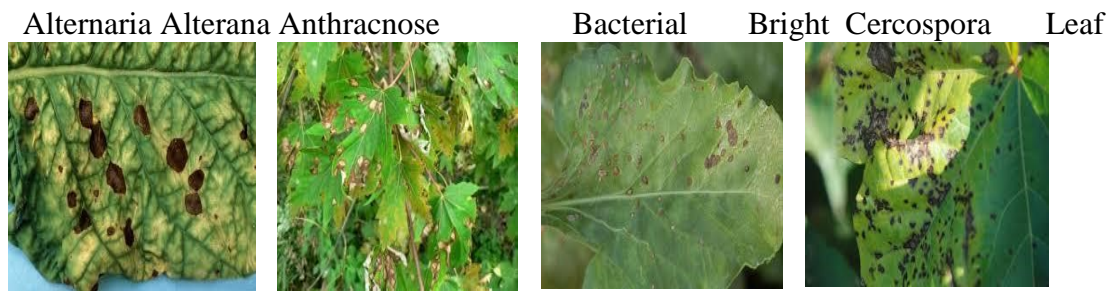
Figure 2.1: Various diseases in plant leaf.

In image acquisition step, the images are read one by one by using `IMREAD` function, then in the image pre-processing step some adjustments are applied on the images to enhance their quality in order to make it more clarify for processing ,

`IMRESIZE` function is also used to convert images to 256*256, `IMADJUST` function is used to have all the images in the full-screen of the figure. Moreover we convert the images from RGB domain (see Fig. 2) to LAB color domain using `SRGB2LAB` function as shown in Fig 3.
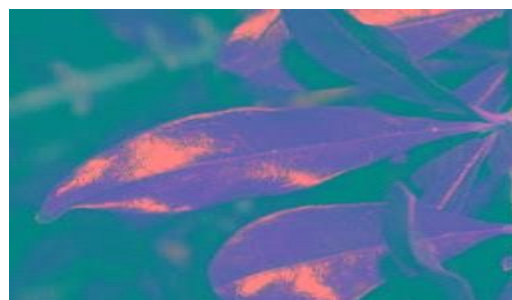
Figure 2.2: Original image in RGB	**Figure 2.3:**Image in LAB color domain

After the images has been transformed to LAB color domain (CForm) the segmentation is implemented using K-means clustering algorithm. However,

Kmeans works on data not images. It clusters the data based on the distances of similarities between the neighbors. Thus, the goal is to reshape the image into vector data not image and then it is clustered with K-means algorithm. Knowing that two column vectors (layers) the layer A and the B from LAB color space, are included from the four clusters, the aim is to put each layer in a column in this vector of data in order to isolate the region of interest. The details of the method is given as follows:

- The layer A and B are extracted then the image are reshaped into vector with two columns.

- K-means clustering is done by choosing the number of clusters, the selected layers will be divided in four because more accurate result appears when K=4 clusters are chosen on layers A and B otherwise the region of interest are mixed with the background.

- Then the K-means clusters for the selected layers which are 2 and 3 are stored the segmented pixels in Matlab variable `CIndex`. Obtained clusters are shown in Figure 4.

(a)   **(b)**   **(c)**

(d)   **(e)**
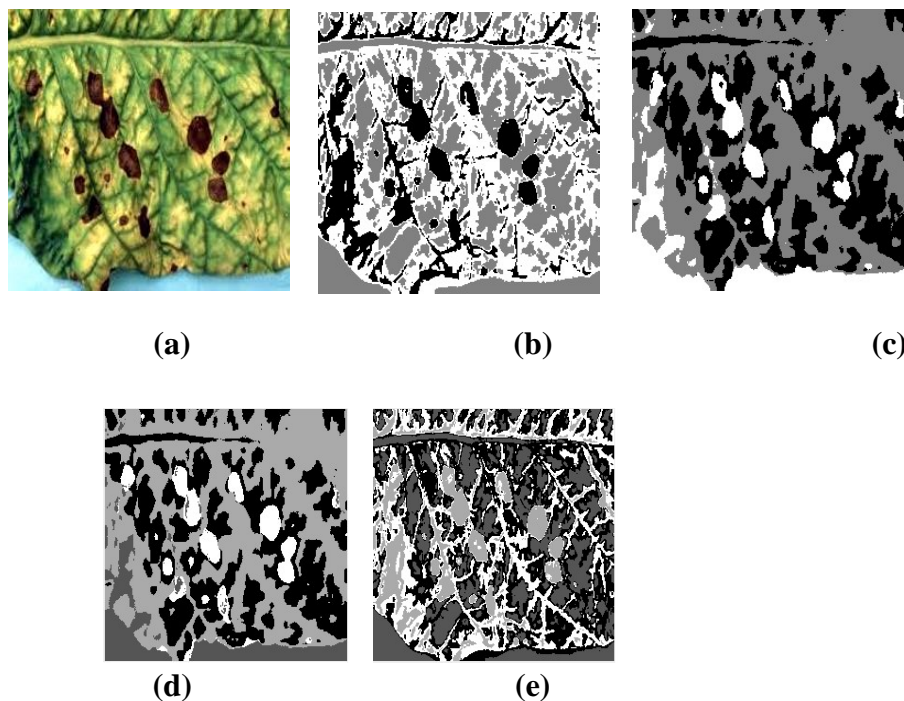
Figure 2.4: K-means clustering and layer selection process. (a) Original image, (b) Kmeans results for K=3 and 1st and 2nd layers of LAB (Layers L and A), (c) K-means results for K=3 and 2nd and 3rd layers of LAB (Layers A and B), (d) K-means results for K=4 and 2

Hence, two layers are conducted from 3 layers of LAB color space, important results appeared when layers A and B was divided into four clusters, as shown in Figure 4(d).
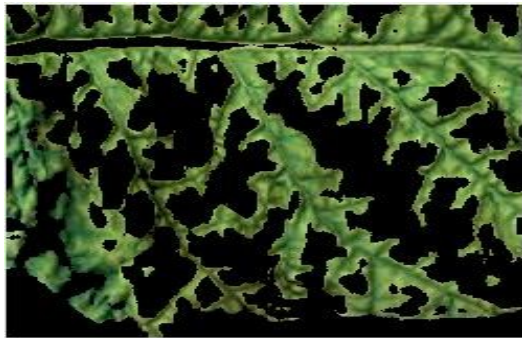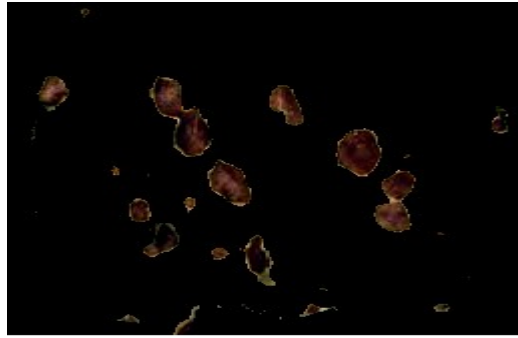
<div align="center">(a) Cluster 1   (b) Cluster 2</div>
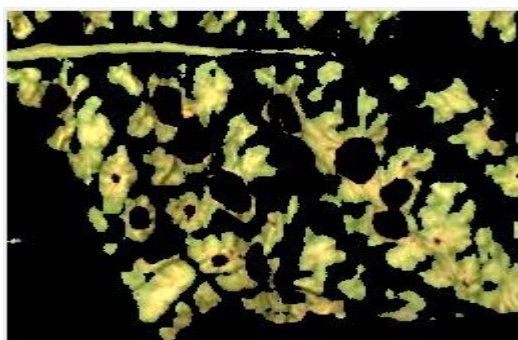
<div align="center">(c) Cluster 3   (d) Cluster 4</div>

Figure 2.5: Segmented image in color.

Figure 5 shows results of segmented image in color obtained by K-means clustering, the four classes are sited and each ones represent a separate range of colors, hence cluster 2 shows only the region of interest which is the disease part.

2.3 LAB color space

This space is also known as CIELAB (see Figure 6) and CIELAB contains three layers: The L axis reproduces the brightness, the A axis is responsible for the orangered hues and greens while the B axis describes the colors from blue to yellow. The color model is hardware independent. Colors are defined regardless

of how they are produced and their reproduction technique. The LAB color space is a uniform color space, thus the greater the distance between the colors in the color space, the more the color difference becomes clear for the eye. All the color information is in the A and B layers.
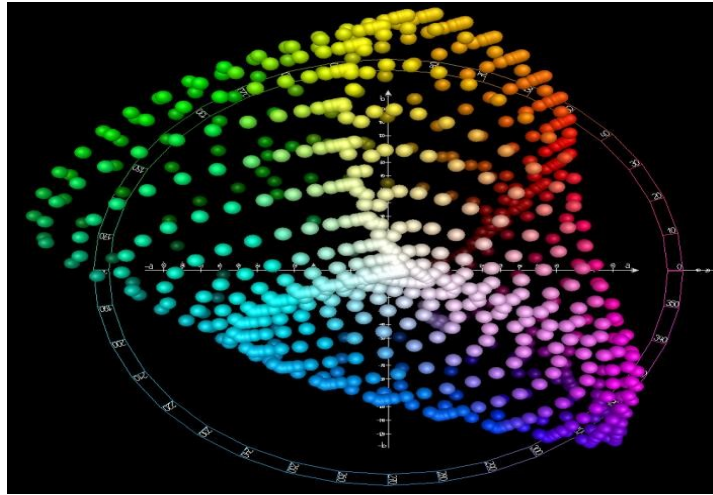
Figure 2.6: CIELAB color space

2.4 K-means clustering method

K-means is one of the most common clustering algorithms. It makes possible to analyze a set of data characterized by a set of descriptors, in order to group the "similar" data into groups or clusters.

The similarity between two data can be inferred from the "distance" separating their descriptors, thus two very similar data are two data whose descriptors are very close. This definition makes it possible to formulate the data partitioning problem as the search for K "prototype data", around which the other data can be grouped.
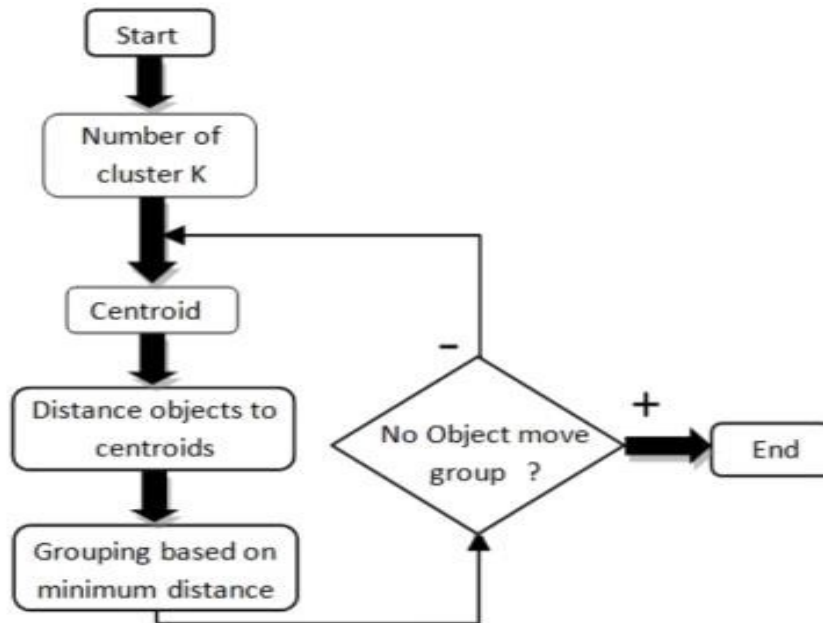
Figure 2.7: Flow chart of K mean algorithm.

The algorithm starts by choosing K initial cluster centers of K groups then place the objects in the group of nearest center then recalculate the center of gravity from each group. In addition of those steps the algorithm iterates until the cluster centers stops to change anymore groups (see Figure 7).

In the proposed method there are 3 steps and the following flowchart explains the first step of our work which is pre-processing step, starting by initializing all features intended to be used in this study after that some adjustments are applied to the size of images and their quality to have all the images in a full screen and also to unify the images size to 256*256, then the images are converted from RGB domain to LAB color domain. Moreover the segmentation step is implemented using K-means clustering which classify the data on the distances of similarities between neighbors, in the end of segmentation step the k-mean cluster the selected layer and stored the segmented pixels in `CIndex` (variable chosen to store the segmented pixels on Matlab code). By the end of the preprocessing step the nine features that will be processed by ICA are extracted.

16
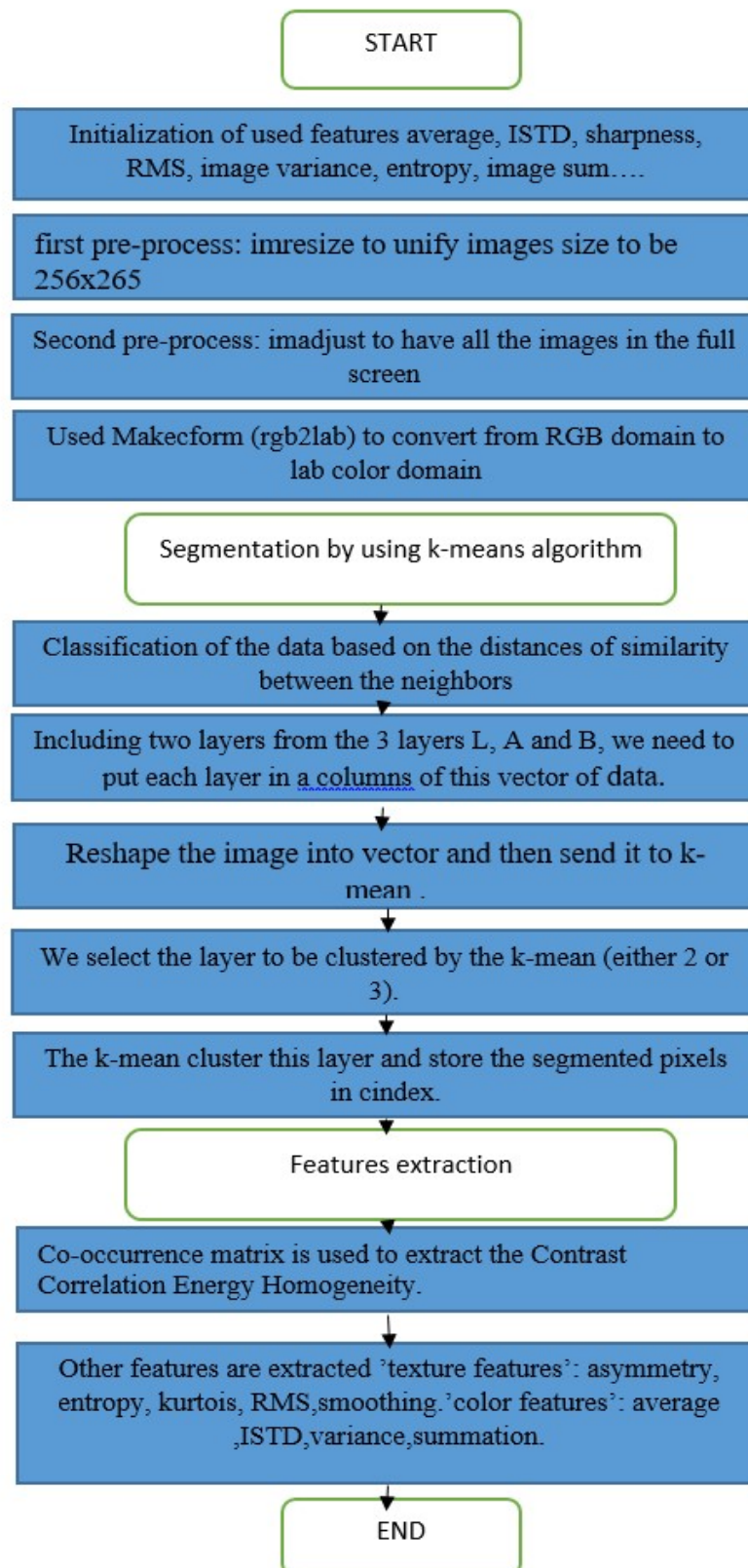
Figure 2.8: Flow chart of the preprocessing phase.

3. FEATURE SELECTION

In order to select the most discriminative features, an optimization algorithm is used. In this thesis imperialist competitive algorithm (ICA) is chosen for selection of discriminative features. The role of ICA is to choose the optimal features within 9 used features and the algorithm consists of two competition mechanisms:

- The intra-empire competition is competition between members of an empire,

- The inter-empire competition is the competition within empires, according to the overall objective value of the empire. Within the empire, the strongest country will be the imperialist (the leader) of the whole empire; It is the intra-empire competition.

- The optimization aim is to find an optimal solution in terms of problem variables. We form an array of variable values to be optimized.

- In Genetic Algorithm (GA) terminology, this array is called "chromosome", but here the term "country" is used for this

array. In our proposed method 9 features are chosen as variables such are:

- color features: average, standard deviation, variance, summation
- texture features: entropy, asymmetry (skewness), kurtosis, RMS, smoothing

In this thesis the Imperialist competitive algorithm (ICA) and particle swarm optimization (PSO) are used to specify the best features within 9 features to reach the best recognition accuracy.

Employing ICA in this thesis is demonstrated in Figure 9 ,the mean component of the ICA is the agents whose functionalities is to look for new solutions to optimize the problem under study ,in our work the problem is to find the

optimal solution to be used in the Artificial neural network in order for the agents to define the optimal features we have to convert the solutions of these agents to be binary which contains zero and one only ,while 1 means the selected features and 0 means the discarded features each one of the agent in the colony of an empire suggest a solution ,this solution then sent to the cost function which contains our ANN model ,then the MSE

($z=$ `perform(net,Target,y)`)obtained by the ANN model for the best selected solution will be sent back to the same agent and which will sent it back to the colony that it belongs to ,then the colonies will compare this cost value with others in order to update the imperialist value ,over inside the cost function ,the ANN takes the samples of the selected features and then it is trained on it and test the identified weights values with the testing data set in order to calculate the value of the MSE which then will be identified as a set value of the selected solution for these corresponding agents. Continuing on this way and completing the required iterations it will insure the proposed system to achieve the best position of the optimal solution which presents the optimal combination of features to be used by ANN for the same problem under study.
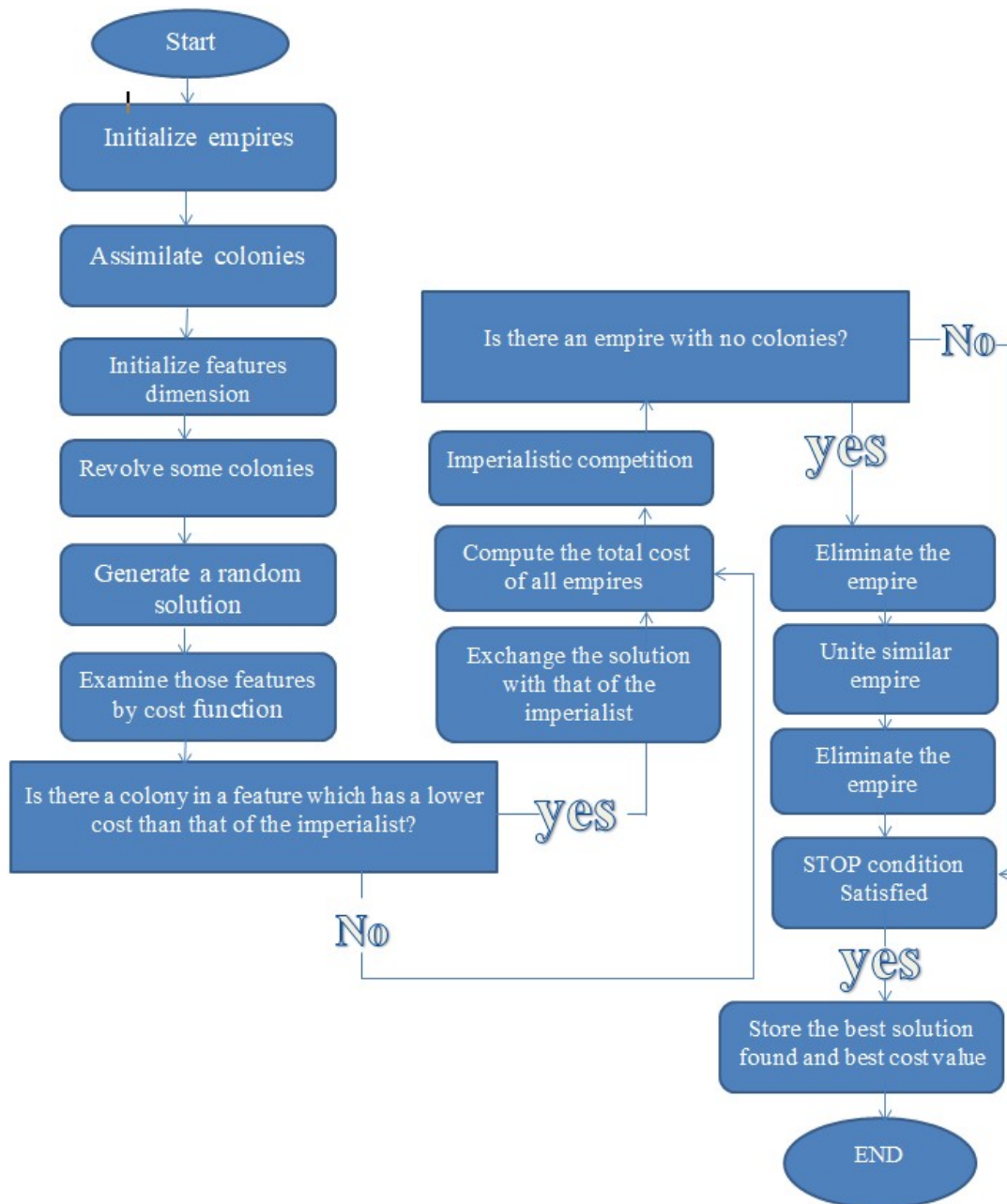
Figure 3.1: ICA workflow to select discriminative features

Using these parameters illustrated in Table our proposed designed can identified the optimal combination of features to be used in detecting diseases under study.

Table 3.1:parameter of ICA used in this work

Parameter	Value
Cost function	ANN
N of variables	9
Max iterations	1000
Number of population	10

3.1 Introduction of Imperialist competitive algorithm (ICA)

The ICA algorithm was inspired from the idea of imperialism, Hence in this process the powerful countries takes control of the weakest one. Moreover this meta-heuristic algorithm was proposed by Atashpaz-Gargari and Lucas [24], Thus ICA has been used in various applications.

Imperialist competitive algorithm (ICA) is a socio-politically strategy proposed to solve optimization problems, Moreover it is a new meta-heuristic algorithm used to solve some engineering issues which is similar to many other developed algorithm, it starts with a random initial population or empires, each agent in this empires is called country. Countries are categorized into two types as of imperialist and colony. ICA is based on a competition between those, in this competition the powerful empires take the position of their colonies and the weakest ones fall in.

- **Imperialist competitive algorithm (ICA)**

Initialize and evaluate the empires   **while** Stop condition is not satisfied **do**

Move the colonies toward their relevant imperialist

Revolution. Make some changes in the characteristics of some of the countries **if** there is a colony in an empire which has lower cost than that of imperialist then

Exchange the positions of that colony and the imperialist   **end if**

Compute the total cost of all empires

if the distance between two empires is less than Uniting Threshold **then**

Unite the two empires   **end if**

Imperialistic competition

if there is an empire with no colonies then

Eliminate this empire   **end if** **end while**
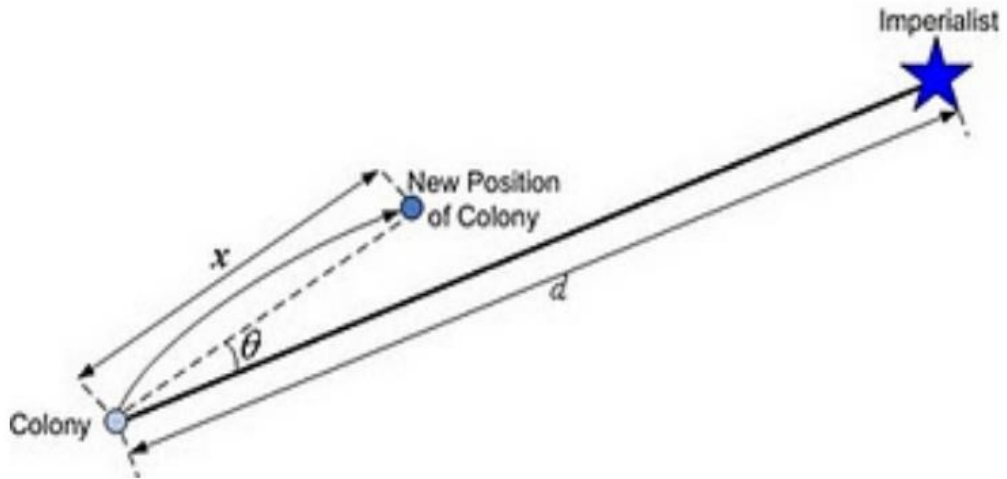
Figure 3.2: Movement of colony

Initialization: In the algorithm each individual of the population represents a country, the first step is to initialize and evaluate them. The best countries are selected and become the imperialists and those who are not become imperialist are colonies. Colonies are then spread over the empires according to their power.

Movement: After having divided the colonies among the different imperialists, each colony moves towards its imperialist. Figure 10 illustrates this movement, d is the distance between the colony and the imperialist. θ and x are random numbers. The number θ, drawn randomly, allows slightly deflect the colony. After each movement of a colony, it may be that the cost of a colony is less than the cost of its imperialist, in which case the colony becomes the imperialist.

Total cost of an empire: The total cost of an empire depends on the cost of the imperialist and it is colonies. The eqn. (1) gives the expression which makes it possible to calculate the total cost of an empire *n*.

$$TC_n = Cost\{imperialist_n\} + \xi.mean\{Cost\{colonies\ of\ empire_n\}\} \quad (3.1)$$

23

where ξ is less than 1.

Imperialist competition: Imperialist competition allows a stronger empire to seize of a colony of the weakest empire. This competition makes it possible to increase the power of strong empires and eliminate weak empires. Figure 11 illustrates this competition; the weakest empire 1 is on the verge of losing a colony for the benefit of one of the empires. Each empire has a probability of possession P_i calculated from its power.
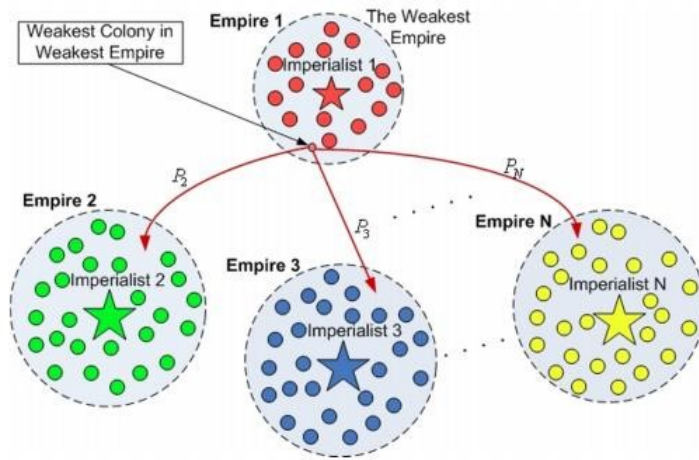
Figure 3.3: The competition of empires: the most powerful empire is more likely to have the weakest colony of the weakest empire

The imperialist competitive algorithm is an evolutionary algorithm, which works on the same way like other population-based algorithms, it starts by initializing the population where the best countries are selected as the imperialist another makes part of the colonies imperialist. All the colonies of the population are distributed on the imperialist countries according to their power.

After dividing all the colonies among the imperialists, the colonies begin to move towards their imperialist countries. The total power of all the empires depends on the power of the imperialist countries and the power of their colonies (the countries with the least cost function value become powerful imperialists and start controlling other countries called colonies, hence the initial imperialist is formed. This fact is defined with the full power of an empire by the power of the imperialist state in addition to the percentage of the

average power of its colonies. The imperialist competition will finish by an increment in the power of the powerful empires and decrement in the weaker empire"s power. However the empires that cannot remain in this competition and cannot increase its power will be ignored or removed. Nevertheless the weakest empire will collapse one by one while all countries will definitely become a state in which there is only one empire and all the remaining countries are colonies of that empire. General flow chart of ICA is depicted in Figure 12 [24].
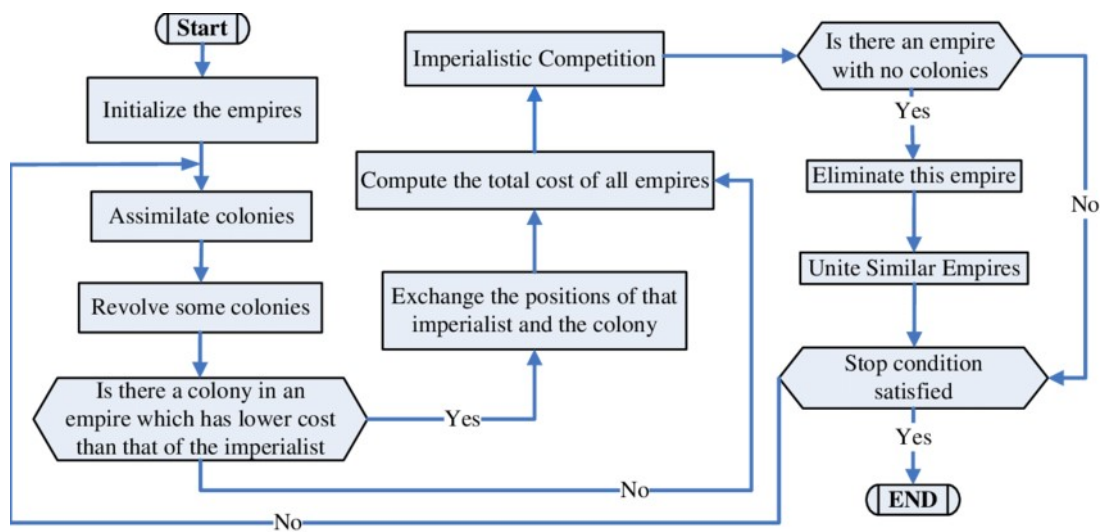
Figure 3.4 General flow chart of ICA.

3.2 Particle swarm optimization (PSO)

Particle swarm optimization or PSO is an intelligent optimization algorithm, it belongs to a class of optimization algorithms called meta-heuristic algorithms, and inspired by social behavior of flocking birds and schooling fish. PSO is a simple powerful optimization algorithm applied successfully to an enormous applications in various field in science and engineering such as Machine learning, image processing, data mining and many other fields .Initially PSO is introduced by James Kennedy and Russel Eberhat in 1995 originally, they were working to developed model to describe a social behavior like flock of birds and school of fish. However, they released that their model is capable of doing optimization test so they proposed a new optimizer based on their model just called Particle Swarm Optimization .in those last 20 years PSO became one of the most useful and most popular algorithms to solve various optimization

25

problems in various field. Despite its simplicity PSO is also a powerful algorithm.

- **Algorithm of Particle Swarm Optimization PSO**

The basic algorithm is very simple:

We denote g the best known position of the swarm and f (x) the function that calculates the criterion of x.

For each particle:

We initialize our position

We initialize its best known position p as its initial position

If f (p) <f (g), we update the best position of the swarm The speed of the particle is initialized.

As long as we have not reached the maximum iteration or a certain value of the criterion: For each particle i:

We draw random c2 and c3

We update the speed of the particle according to the formula seen previously

We update the position xi

If f (xi) <f (pi),

We update the best position of the particle

If f (pi) <f (g), we update the best position of the swarm.

The principle of the algorithm can be more easily visualized from the **Figure 13**.

PSO uses a number of particles called the swarm of candidate solution called particles. These particles are allowed to move around and traverse the search-space.

These particles move to a new position depending to those rules:

- The particle''s own previous velocity (*Inertia*)

- Distance from the individual particles'' best known position.

- Distance from the swarms best known position (*Social Force*)

Based on those simple rules the particles will corporate to find the best location in the search space and best solution for the optimization problem.
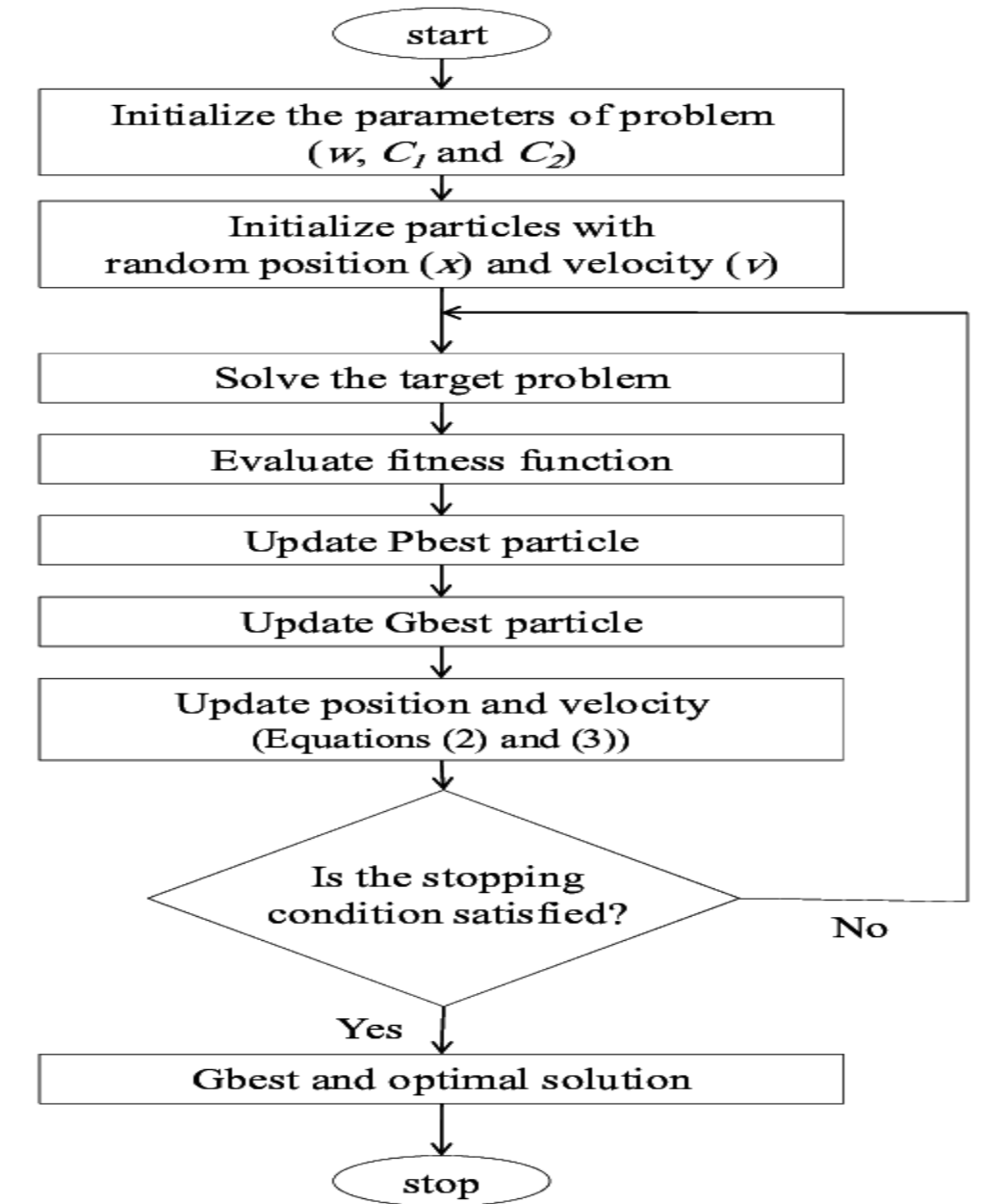
Figure 3.5: General Flow chart of Particle Swarm Optimization (PSO).

3.3 Mathematical Model of PSO
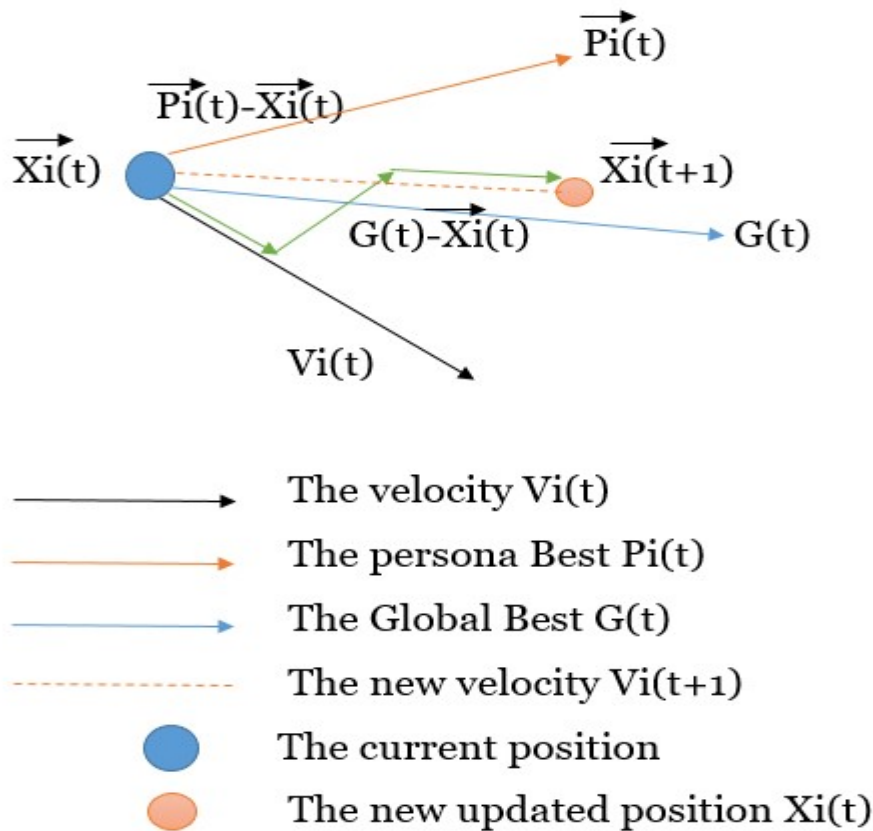
Figure 3.6: Mathematical model of PSO

We denote the position of Particle i by $\overrightarrow{Xi(t)}$ and having also velocity for every particle denoted by $\overrightarrow{Vi(t)}$ in a same space as position, the dimension of X and V are same ,the velocity describes the movement of particle (i) in the sense of direction and the sense of distance .thus we have a particle in time step **t** which is located in the position $\overrightarrow{Xi(t)}$ and it moves towards a vector such $\overrightarrow{Vi(t)}$ as shown in (**Figure 14**). These particles are interacting and learning from each other or obeying some simple rules to find the best solution for an optimization problem .In addition to position and velocity every particle has a memory of its swarm best position or best experience this is denoted by personal best $\overrightarrow{Pi(t)}$ shown in (**Figure 14**) We have also a common best experience among a member of swarm denoted by G(t) it hasn"t index of i because it belongs to the whole swarm ,thus we have personal best for every particle and a global best which is the best experience of all particles in the swarm .The mathematical model of

28

PSO is very simple by defining these concepts in every iteration of PSO P and V are updated according to this simple mechanism. The particle moves to a new position by using all those three component or three vectors shown in (**Figure 14**).Tthe particles moves somewhat parallel to the vector of velocity, same as to the vector connecting $\overrightarrow{Xi(t)}$ and $\overrightarrow{Pi(t)}$ and moves somewhat to the vector connecting the $\overrightarrow{Xi(t)}$ to the G(t) (**Figure14** in green),thus the new updated position is created as shown in **Fig 15**.
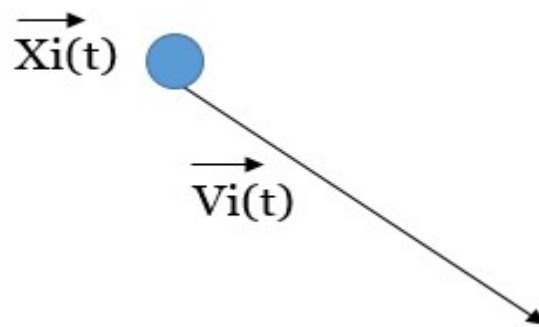
Figure 3.7: Simple Model of Moving Particle Some considerations are necessary to complete the standards of PSO:

The equations of updating the position of the particles are as follows:

$$\overrightarrow{Xi(t+1)} = \overrightarrow{Xi(t)} + \overrightarrow{Vi(t+1)}$$

$$\overrightarrow{Vi(t+1)} + w\overrightarrow{Vi(t)} + C1\overrightarrow{(Pi(t) - Xi(t)} + C2\overrightarrow{G(t) - Xi(t))}$$

The equations of updating Velocity are as follows:

$$\overrightarrow{Vij(t+1)} = W\overrightarrow{Vij(t)} + r1C1\overrightarrow{Pi(t)} - \overrightarrow{Xij(t)} + r2C2(G(t) - \overrightarrow{Xij(t))}$$

$$\overrightarrow{Xij(t+1)} = \overrightarrow{Xij(t)} + \overrightarrow{Vij(t+1)}$$

With:

$\overrightarrow{Vi(t+1)}$: updated velocity $\overrightarrow{Vi(t)}$:the current velocity

$\overrightarrow{Xi(t+1)}$: updated position $Xi \overrightarrow{\ \ (t)}$:the current position

W=Interia coefficient   **r1, r2:Random** variable uniformly distributed in the range of [0.1]

C1, C2: accelaration coefficient.

3.4 The comparison between ICA and PSO

For many problems, there is no deterministic solution that gives the result in a reasonable time, and this despite the creation of more and more powerful computers. To overcome this problem, we use methods known as heuristics, that is to say methods that provide an approximate solution. However, it is necessary to reproduce the process on several iterations to move towards an acceptable solution. Among these heuristics are some algorithms that have a generic principle adaptable and therefore applies to several optimization problems. They are called meta-heuristics.. Such as Imperialist competitive algorithm(ICA)and Particle Swarm optimization (PSO), in the beginning of this chapter a definition of each of those algorithm is given and looking to do a simple comparison between its .

3.5 Using Imperialist competitive algorithm

The algorithms used in this work are optimized algorithm their concept is similar which is looking for the best solution in order to solve an optimization problem.

in this work artificial neural network is used as decision making for applying the training and testing process, thus the finest optimized neural network is obtained by varying various parameters of hidden nodes of network ,training percentage for training ANN ,number of epochs, as soon as the developed network became successful ,the proposed algorithm starts by suggesting features from the extracted ones implementing ICA agents ,then send them to the cost function which contains our Artificial neural network to train on the samples for the selected features and test them ,Moreover the output will be calculated by ANN and compare it with the target ,Hence ANN calculate the MSE then the value of MSE obtained by the ANN model for the best selected solution will be sent back to the same agent and which will sent it back to the colony that it belongs to, then the colonies will compare this cost value with others in order to update the imperialist value. Completing the required iterations will insure the

designed model to achieve the optimal solution which presents the optimal combination of features that can be easily used to determine the types of diseases under study and classify them accurately.
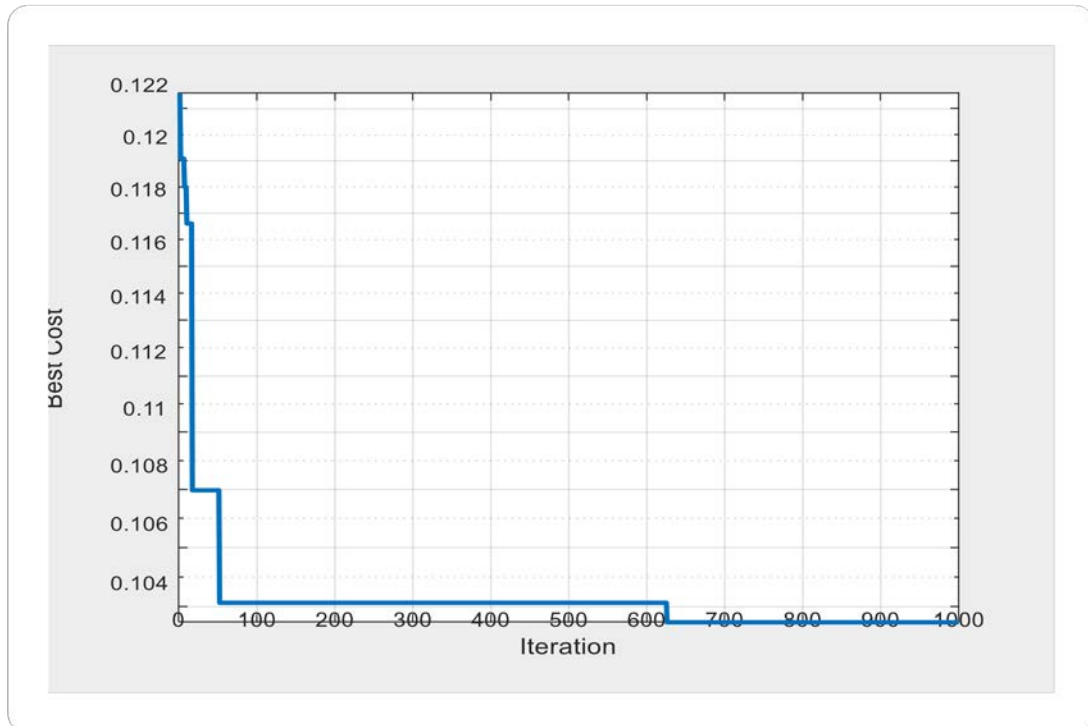
Figure 3.8: Example of convergence curve of best cost giving by ICA in each iteration from 0 to 1000

The best solution is the selected features by ICA algorithm, in each run of ANN

1000 iterations are done, the results is giving in the 1000 iterations with best cost (MSE value) Figure16 (the result that has the lowest cost function (MSE) is taken as final result)

3.6 Using Particle swarm optimization PSO

In our work also Particle swarm optimization is also implemented to find a best solution for our problem which are the optimal features, the PSO algorithm works in the same way as ICA , However PSO has particles which moves around in the search-space where the solutions is located , our PSO tries to find a solution between it .The cost function of PSO was also built using ANN algorithm ,thus the selected solutions by PSO particles are sent to the cost function ,where the ANN is trained on them and then it calculates the MSE

value to compare the ANN output with the required target .then this value is returned back to the PSO in order to determine a new solution.
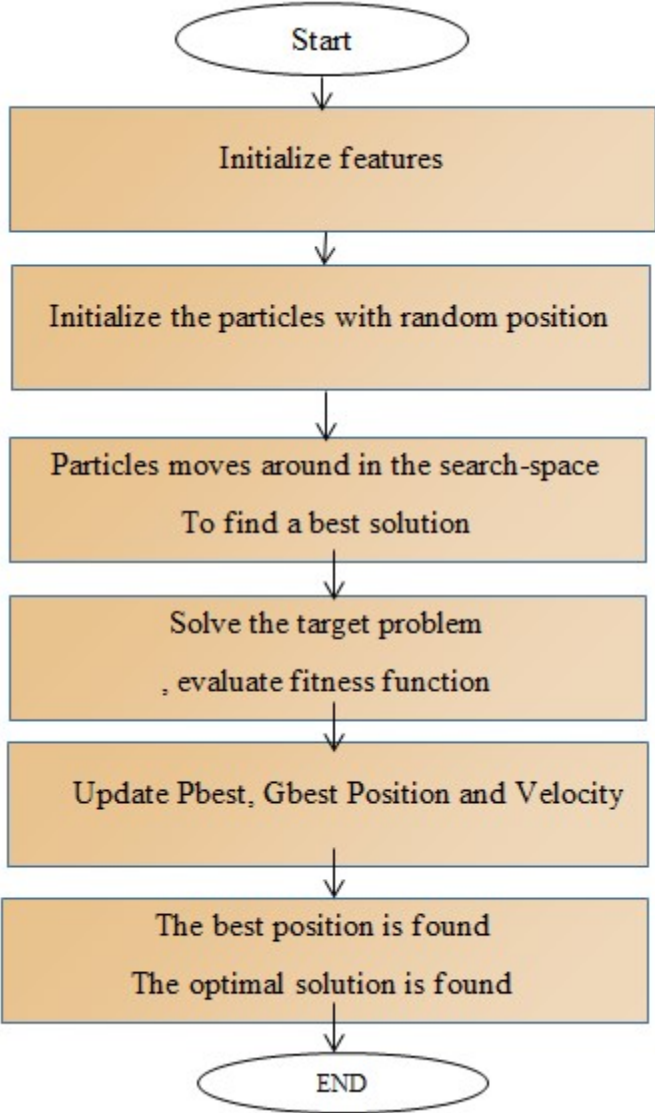
Figure 3.9: The way PSO works to select optimal features
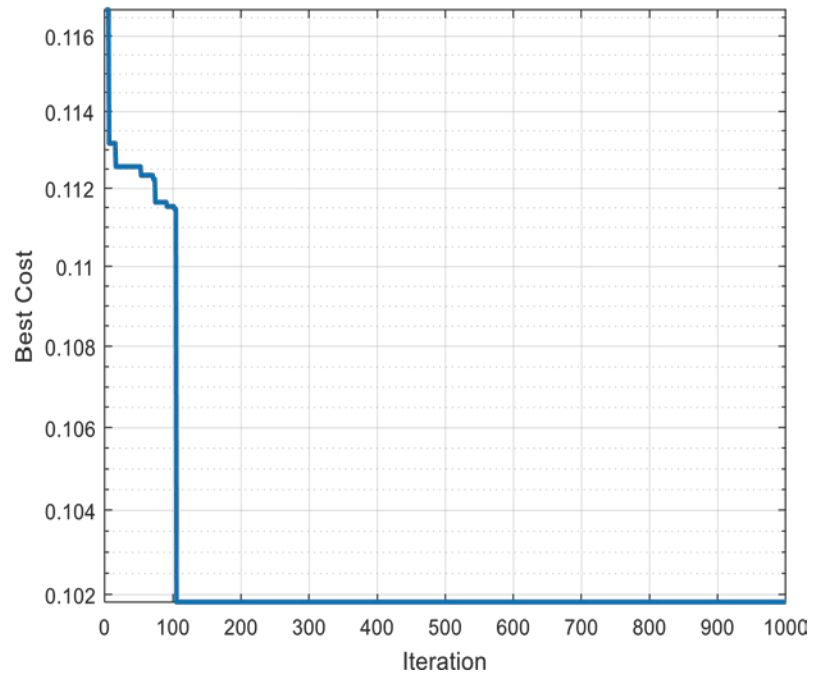
Figure 3.10: Example of convergence curve of best cost giving by PSO in each iteration from 0 to 1000

4. DECISION MAKING

There are two methods of classification technique in machine learning which are supervised and non-supervised methods, however in this part focus has been on the supervised method used in our work, therefore supervised classification is categorized into two as parametric and non-parametric methods.

In the supervised context we already have examples whose class is known and labeled. The data are therefore associated with class labels denoted $\Theta = \{q1, q2, ..., qn\}$. The goal is then to learn using a learning model of the rules that allow to predict the class of new observations which amounts to determining a Cl function that from the descriptors (D) of the object associates a class qi and to also be able to affect any new observation to a class among the classes available. This comes back to the end to find a function we note Ys that associates each element of X with a Q element. then a template for classifying the new data. Among the supervised methods one quotes: k-nearest neighbors, decision trees, neural networks, machines with vector support (SVM).

Moreover, In this study Artificial neural network is used which is an example of nonparametric classifier. Non parametric classifiers are used in case of unknown density function and used to estimate the probability density function. Thus, there are large number of classifiers/methods come to existence that are used for classification task. Such as K- nearest Neighbor, Kernel Density Estimation, Artificial, Logistic Regression and neural network.

Hence the cascade forward neural network is implemented   in this work ,which works as a cost function of one of optimization algorithm used ,60 number of hidden layer is used ,There can be multiple hidden layers in the model depending on the complexity of the function which is going to be mapped by the model. However, when there are many hidden layers, it takes a lot of time to train and adjust wights.More details about neural network used in this study are giving in Table 2.

Table 4.1: Parameters of ANN used in this work

Parameter	Value
Hidden layer	60
Performance function	Mean square error
Training function	„trainscg"
Epochs	1000

Using these settings illustrated in table 2 our proposed designed identified the optimal combination of features that will be used in detecting diseases under study.

4.1 Artificial neural network (ANN)

Neural networks are one of the most important fields of control engineering and artificial intelligence, which reflects a significant development in the human way of thinking. The idea of neural networks revolves around the simulations of the human mind using the computer.Artificial neural networks Are computational techniques designed to mimic the way the human brain performs a task by massively distributed parallel processing consisting of simple processing units. These units are computational elements called neurons or nodes, which have a neurological characteristic in that It stores practical knowledge and empirical information to make it available to the user by adjusting the weights.
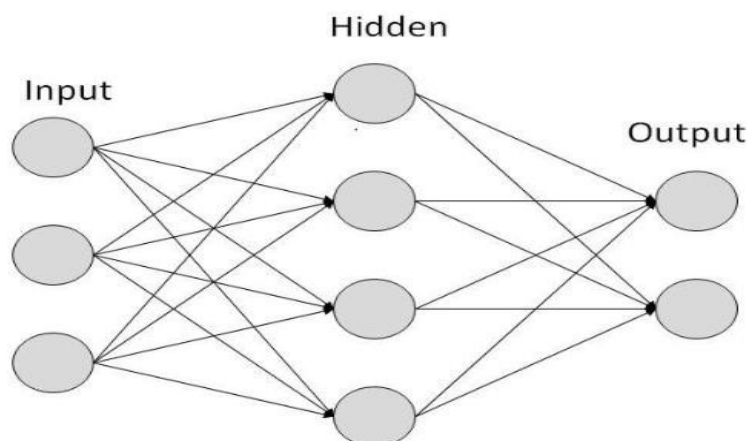
Figure 4.1: Artificial neural network system

The Artificial neural network (ANN) is similar to the human brain in that it acquires knowledge of training and stores this knowledge by using connecting forces within neurons called tangential weights.

Just as a human has input units that connect it to the outside world and its five senses, so do the neural networks, which need input units, and processing units where calculations are determined by the weights and through which the reaction is appropriate for each input of the network. Input units are a layer called the input layer, and the processing units are the processing layer that outputs the network outputs.

- **Introduction of Neural network used in this work (feed forward)**

In this work cascade feed forward network is used, it is an acyclic artificial neural network, thus distinguishing itself from recurrent neural networks. The best known is the multilayer perceptron which is an extension of the first network of artificial neurons, the perceptron invented in 1957 by Frank Rosenblatt2.

- **Artificial neural network in our work**

The feed forward neural network was the first and simplest type of artificial neural network designed. In this network, information travels only in one direction, forward, from the input nodes, through the hidden layers (if any), and to the output nodes. There are no cycles or loops in the network.

Our Neural Network has Input which are the samples for the selected features (extracted features by ICA),and in the output has 5 classes which are the 4 diseases and healthy leaves ,Moreover the Algorithm used in this work is Imperialist competitive algorithm ,ICA"s role is to suggest features from the extracted ones send it to cost function which send them to ANN and then it is trained on it and test the identified weights values with the samples set in order to calculate the value of the MSE ,this value will sent back to the cost function which will give it back to its agent to compare it with the latest value in order to update the cost value.
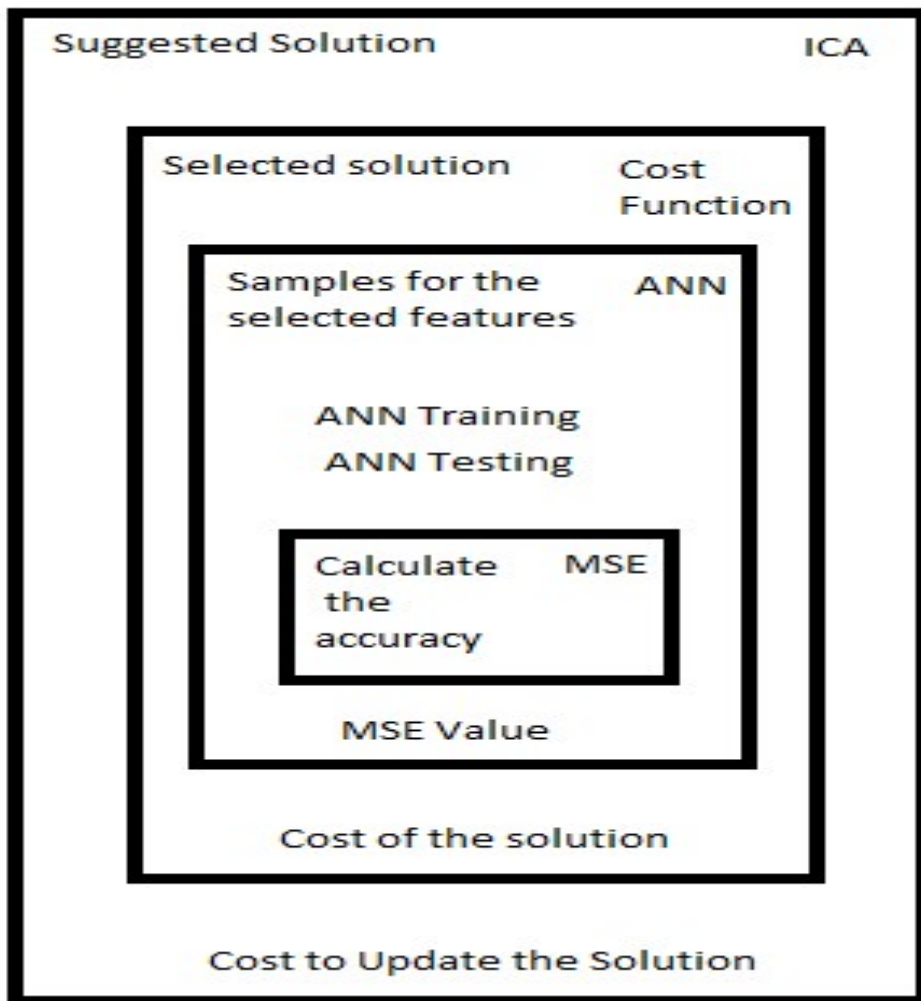
Figure 4.2: The proposed method of selecting optimal features

This systems works in the form of three connected rings between them it is role is to minimize the cost function, to determine the solution as shown in figure20,firstly ICA algorithm suggest random solution, sent to the cost function as selected solution where our ANN model is located .The ANN takes the samples of those selected features as input train on them using 60% (experiment 1) or 70% (experiment 2)of data After that the testing is done by ANN using 15% (experiment 1) or 20% (experiment 2) remaining data the MSE is calculated in order to compare the output with the required target .this operations is done in each run until the maximum iterations is reached .
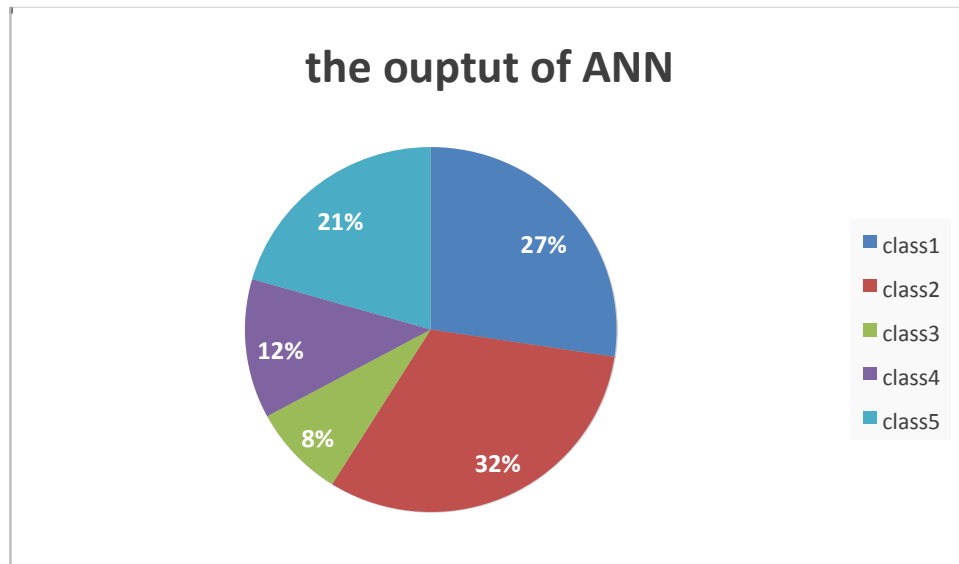
Figure 4.3: The percentage of samples in each class (The ANN"s output)

Our input will be classified according to 5 classes as shown in Figure 21, four classes are diseases images as explained before first class describes diseases called Alternaria Alternata which represents 21% of our datasets , from the second disease which is anthracnose 21% of images are manipulates, furthermore 8% of our data represents the third disease named Bacterial Bright , the forth class describes the Cercospora Leaf disease which is 12% ,in addition to those diseases extra class is included which is the class 5 (as shown in Figure 20) represents 32% of healthy leaves.

5. DESIGNED EXPERIMENTS AND RESULTS

5.1 Designed experiments

Forth experiments were conducted for the study of PSO and ICA algorithms in our work, according to the percentage of samples used in each experiment with both algorithms significant results were conducted which will be explained in details in the next section.

In order to collect enough results using two algorithms implemented in this study, two experiments are done for each algorithm operating the same datasets, as mentioned before both algorithm PSO and ICA have the same aim (finding optimal solution for optimization problem) However each in a different way .thus in this work 4 experiments are done, the number of samples has been changed according to testing, training and validation.

In each of these experiments approximately ten results are collected, the results obtained using artificial neural network in each run thus we have gathered around twenty different results for each algorithm.

Data sets used in this study:

RGB images with size of 256*256 pixels and four categories of diseases images are implemented; with healthy leaves. Total number of samples are used in the experiments is 73.

The features extracted from those images after segmentation and getting the region of interest represents the input of our neural network in each experiment, which the aim of our study is finding the optimal ones between them.

- **Experiments of ICA algorithm**

Experiment 1:

In this experiment 20% of samples are used for validation 20% for testing and 60% for training, our proposed system try to find a solution to an optimization problem , the ICA algorithm is implemented to give the optimal features

between the nine features. The ICA suggests features from the giving ones by comparing between them using its own algorithm(explained before) the selected features is sent to the artificial neural network to test them ,at the same time it calculate the output and compare it with its target in order to get the MSE value (cost value). Furthermore, the error is calculated which will be sent back to the ICA as z-value, if this value is the lowest value founded it retains the features otherwise it looks for others this process is done until the ANN attend the maximum iterations which are 1000.

Table 5.1:10 trails done by ANN with ICA algorithm 20% testing 60% training and 20% validation.

Trail	Test samples	Train. samples	Validation per	Training per	Testing per	Best cost	Features	Grad.
1	20	60	0.1997	0.12	0.21	0.10158	111110100	0.0442
2	20	60	0.1644	0.11	0.25	0.10784	001110001	0.0400
3	20	60	0.1925	0.11	0.17	0.10566	100101111	0.0370
4	20	60	0.2065	0.11	0.21	0.10584	111111100	0.0317
5	20	60	0.2379	0.13	0.23	0.09550	000101000	0.0203
6	20	60	0.1593	0.11	0.2	0.10013	010101100	0.0308
7	20	60	0.1135	0.12	0.18	0.11133	100111001	0.0443
8	20	60	0.2027	0.13	0.15	0.10729	101010111	0.0282
9	20	60	0.1305	0.12	0.18	0.10278	000100111	0.0446
10	20	60	0.1833	0.16	0.29	0.0963	101010010	0.0551

Table 3 contains results of ten trails done by our system with ICA algorithm, in each trail 20% samples for validation 60% samples for training and 20% testing are taken, as shown in the table different features and best cost are given in each of those trails also the gradient of error , Therefor in this experiment the testing performance in the eighth trail was the lowest one but according to validation it cannot be taken as a best results in this experiment. Moreover the testing performance of the trail 5 was approximately to the validation and also the cost value or MSE was the lowest one thus it can be taken as final result until all experiments being completed .

Graphics for each element are implemented in this experiment in order to see the extent of change in the pattern the graphics are as follows:
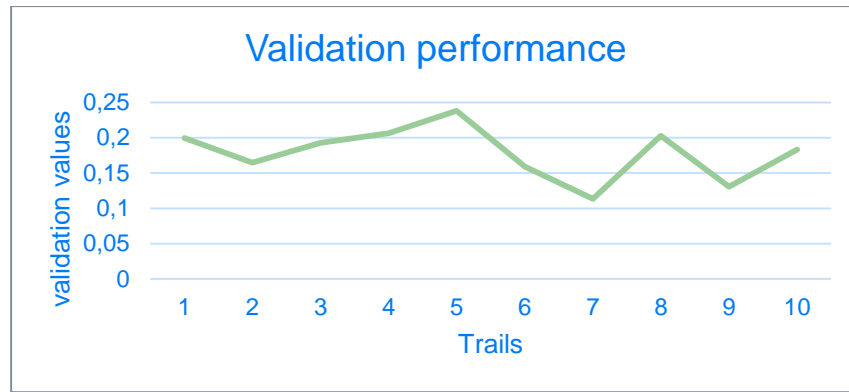
Figure 5.1: Validation performance of Experiment 1 by ICA in each trail

In first experiment of ICA The maximum value of validation performance is shown in trail number 5 (0.12792) and the Minimum value was in trail 7 (Figure) ,While the pattern was vibrated from the beginning until the end .
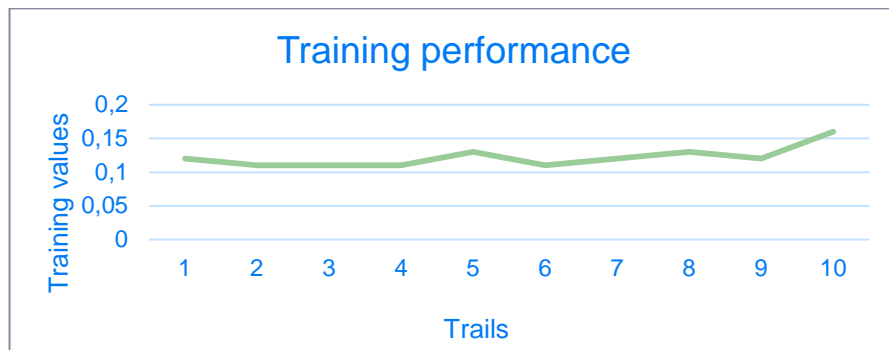
Figure 5.2: Training performance of Experiment 1 by ICA in each trail

The Figure 23 shows the values of training performance in experiment 1 of ICA which the high value was in trail 10 and the lowest one in trails 2,3 and 4 ,in addition to this the pattern of training performance in this experiment was also vibrated from the beginning until the end .
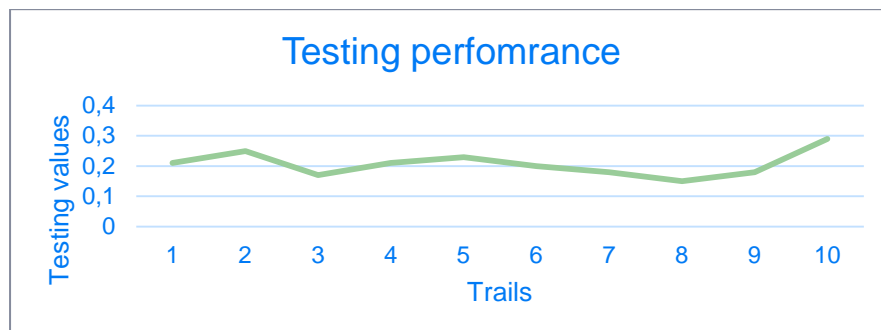
Figure 5.3: Testing performance of Experiment 1 by ICA in each trail

The figure 24 above shows the values of testing performance of experiment 1 which the pattern was vibrated and the maximum value was noticed in the last trail while the minimum one was in the trail number 8
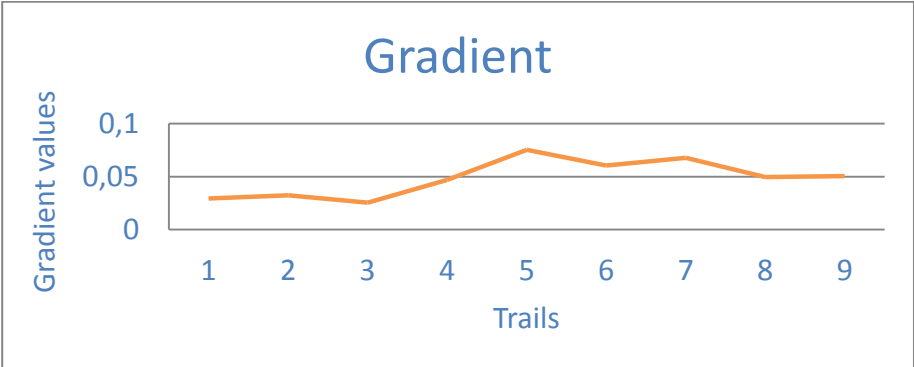
Figure 5.4:: gradient of Experiment 1 by ICA in each trail

The pattern of gradient was also vibrated in this experiment as the figure shows, while the maximum value was in trail 10 and the minimum one was in trail 5 (figure 25) and we can also noticed that the gradient of error was very low in trail number 5.

Figure 5.5: The best cost of Experiment 1 by ICA in each trail

 Figure 26 shows the pattern of last element of this experiment which is Best cost, while the maximum value was shown in Trail 7 and the lowest one was in trail5.

Experiment 2

In this experiment same algorithm is used ICA with ANN but with different percentage of samples, for validation 15% of samples is used ,70% for training and 15% for testing .our system work on the same way as mentioned before in

the first experiment. The below table contains the results which will be explained in details below it.

Table 5.2: Nine trails done by ANN with ICA algorithm 15% testing 70% training and 15% validation

Trail	Testing samples	Train. Samples	Validation per.	Training per	Testing per	Best cost	Features	Gradient
1	15	70	0,38	0,15	0,3	0,0941	000111100	0,02928
2	15	70	0,1673	0,09	0,16	0,0803	001111100	0,03242
3	15	70	0,1331	0,12	0,28	0,1042	100000110	0,02544
4	15	70	0,254	0,25	0,5	0,1038	010111101	0,04678
5	15	70	0,1822	0,15	0,25	0,1075	101111000	0,0753
6	15	70	0,2303	0,15	0,2	0,1059	111110111	0,06047
7	15	70	0,2044	0,12	0,2	0,1078	101101001	0,0676
8	15	70	0,1638	0,12	0,163	0,1089	111010111	0,04948
9	15	70	0,1793	0,14	0,179	0,1046	001111110	0,0506

Table 4 shows results of experiment 2 the number of samples is changed ,each trails gives different results as in the first experiment but here we noticed that the lowest best cost was giving in trail number 1 but the testing and training error was high , However the result founded in trail 2 shows a lowest number of testing performance and also validation Moreover its cost function or best cost is lower according to the other ones ,Furthermore if we compare this result with the one in the experiment 1 ,this results is the best result or solution in both experiment because it has a lower cost function and testing performance .

The graphics for the Experiment 2 are as follows:
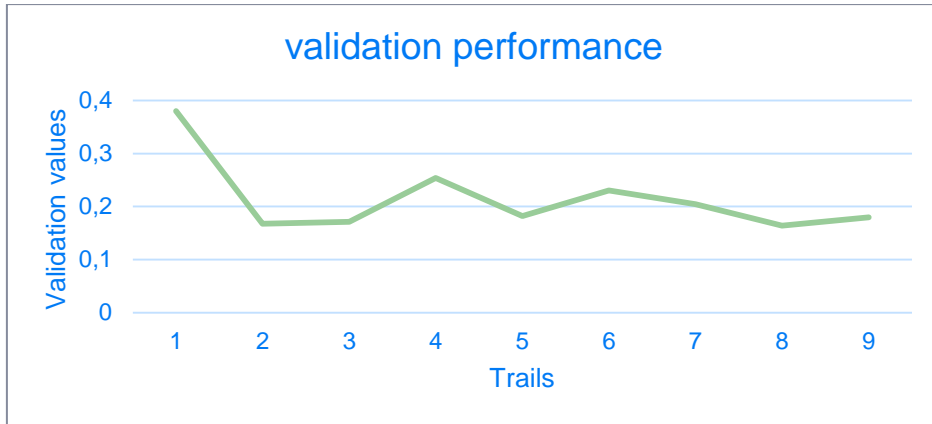
Figure 5.6: Validation performance of Experiment 2 by ICA in each trail

the validation performance of experiment 2 of ICA is shown on the (figure 27) above, which the pattern was also vibrated as in the first one, Moreover the maximum value was in first trail and the lowest one was in trail 3.

Figure 5.7: Training performance of Experiment 2 by ICA in each trail

The pattern of training performance of second experiment for ICA is shown in Figure As shown in the figure the performances vibrated from the beginning until the end but the high value was in trail 4 while the lowest one was in trail 2.
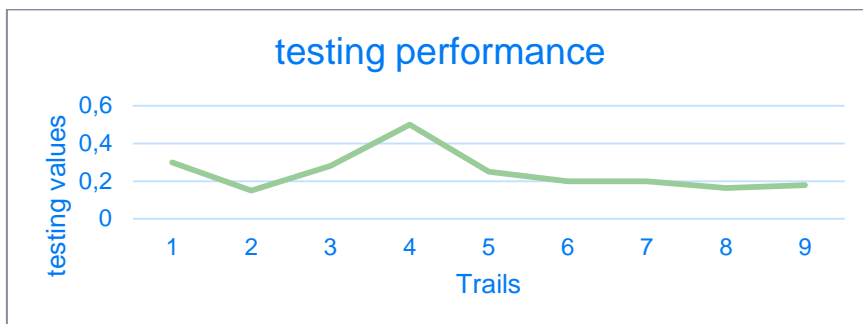
Figure 5.8: Testing performance of Experiment 2 by ICA in each trail.

Testing performance of this experiment was also vibrated as other elements while the maximum value was in trail 4 and the lowest one was in trail 2 as it is shown in figure 29 .
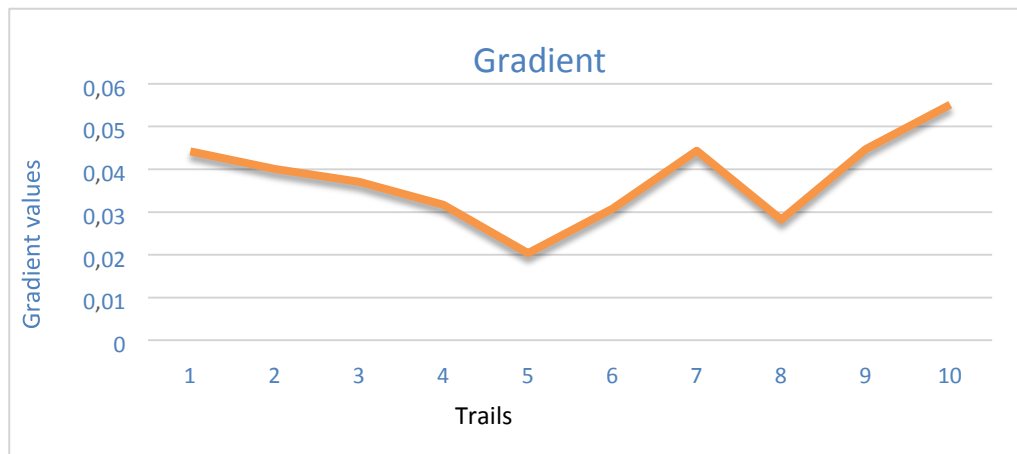
Figure 5.9: The gradient of Experiment 2 by ICA in each trail.

The figure 30 above shows the values of gradient of error in this experiment which the high value was in trail 7 and the lowest one in trail 5, Moreover the pattern was descending from the beginning trail until the fifth one after that it takes another turn however this pattern is also vibrated.

Figure 5.10: Gradient of Experiment 2 by ICA in each trail

This Figure 31 shows the best cost values giving in each trail which the high value was in trail 8 and the lowest one was in trail 1(0.0941) but as the testing error was high we have taken the result of trail number 2(0.10315) which has also a lower best cost and lowest testing error, however the pattern was oscillated from the beginning until the end.

Experiments of PSO algorithm

Experiment 1

In this experiment the same Process is done as with ICA 20% of samples for validation ,60% for testing and 20% for training, the PSO process the images while working with Artificial neural network in order to find the optimal features the particles moves around in the search-space where 9 features are located, our PSO tries to find the optimal one between those ,the distributed particles change their position according to some rules, until the best position is found where the optimal solution is located(optimal features).the following table contain the result of each trail in this experiment .

Table 5.3: 10 trails done by ANN with PSO algorithm 20% testing 60% training

Trail	Testing samples	Train. Samples	Validati on per.	Traini ng per	Testing per	Best cost	Features	Gradi ent
1	20	60	0.3315	0.19	0.28	0.09856	011101110	0.032
2	20	60	0.13148	0.1	0.29	0.09412	101100000	0.031
3	20	60	0.12659	0.11	0.18	0.09406	101000010	0.060
4	20	60	0.16449	0.12	0.1644	0.10697	001110000	0.028
5	20	60	0.17693	0.13	0.1769	0.10531	100001000	0.057
6	20	60	0.1893	0.15	0.19	0.10043	010110100	0.055
7	20	60	0.15597	0.13	0.14	0.1068	000010001	0.018
8	20	60	0.18507	0.12	0.165	0.10575	011110101	0.071
9	20	60	0.14905	0.11	0.21	0.10492	010111110	0.021
10	20	60	0.22019	0.14	0.18	0.10693	110001100	0.074

Table 5 contains the number of samples used in this experiment and also the results of each element such as, testing, training, validation performance ,gradient of error and the given features in each result, from those result we can see that the lowest value of cost function is in the trail number 2 but the testing and validation error was high so we have supposed the results of trail number 7 as final result giving by PSO algorithm because it has a lowest testing error 0.14 ,until the second experiment is done and also the comparison between results giving by both algorithms.

The graphics for the Experiment 1 of PSO are as follows:
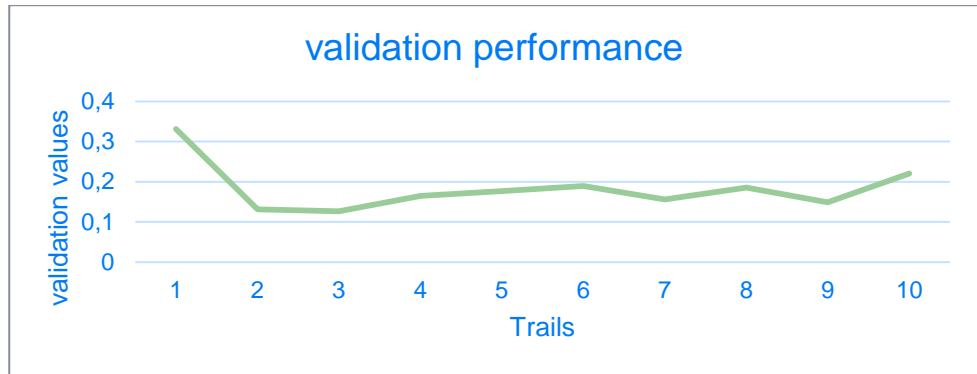
Figure 5.11: Validation performance of Experiment 1 by PSO in each trail

The Figure 32 above shows the values of validation performance in each trail of this experiment which the maximum value was in first trail and the minimum was in trail 2, Moreover the pattern of this element was oscillated.
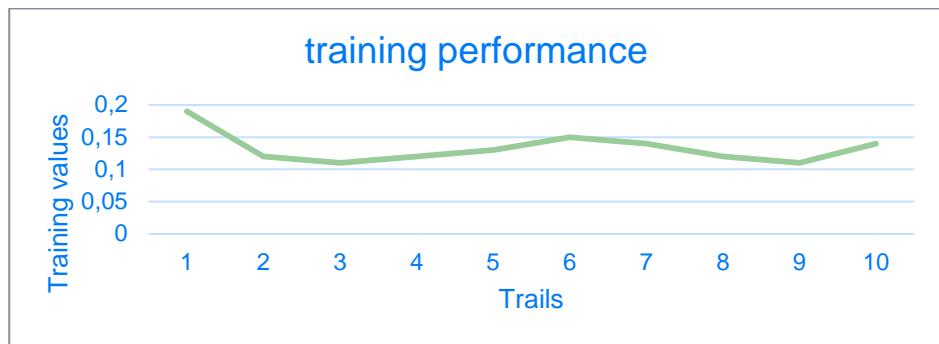
Figure 5.12: Training performance of Experiment 1 by PSO in each trail

The values of training performance are shown in (Figure 33) where the maximum was in trail 1 and the minimum was in trail 2, while the pattern was also oscillated.

Figure 5.13: Testing performance of Experiment 1 by PSO in each trail

The figure 34 shows that testing performance in this experiment was high in trail 1 otherwise the lowest value is shown in trail 7, Moreover the pattern was oscillated.

Figure 5.14: Gradient of Experiment 1 by PSO in each trail

The gradient of error of this experiment is shown in Figure 35 which the maximum value was in trail 8 and the minimum value in trail 7 while the pattern was vibrated From the beginning to the end.

Figure 5.15: Best cost of Experiment 1 by PSO in each trail

The highest value of best cost in this experiment was in the last trail which the lowest one in trail 2; However the pattern of this element was oscillated figure 36.

Experiment 2

In this experiment same algorithm is used PSO with ANN but with different percentage of samples, for validation 15% of samples is used, 70% for training and 15% for testing. Our system work on the same way as mentioned before in the first

Experiment with PSO. The below table contains the results which will be explained in details below it.

Table 5.4::9 trails done by ANN with PSO algorithm 15% testing 70% training and15% validation.

Trail	Testing samples	Training samples	Validation per.	Training per.	Testing per.	Best cost.	Features	Gradient
1	15	70	0.27	0.20122	0.15	0.10413	011110100	0.037766
2	15	70	0.12	0.14494	0.17	0.1028	010010100	0.031195
3	15	70	0.17	0.17055	0.21	0.10813	011110101	0.043681
4	15	70	0.14	0.16564	0.19	0.10665	101011110	0.049739
5	15	70	0.12	0.14829	0.1483	0.10205	001110110	0.048727
6	15	70	0.15	0.15	0.21	0.10068	010111101	0.080709
7	15	70	0.12	0.19374	0,16	0.10264	010100100	0.052054
8	15	70	0.13	0.21437	0.12	0.10395	010111110	0.041298
9	15	70	0.12	0.15071	0.21	0.10925	100111111	0.057854

Figure 5.16: Training performance of Experiment 2 by PSO in each trail

Training performance values of this experiment are shown in the figure above which the maximum value was in trail 1 in the minimum was in the last one, In addition to that the pattern of this element was oscillated figure 37.
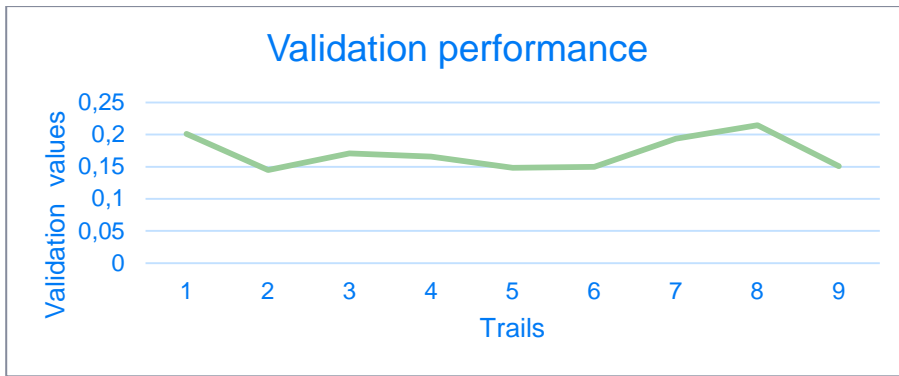
Figure 5.17: validation performance of Experiment by PSO in each trail

The pattern of validation performance in this experiment was also oscillated while the maximum value was in trail 8 and the lowest value was in trail 2, 5, 6 and 9 as shows figure 38.
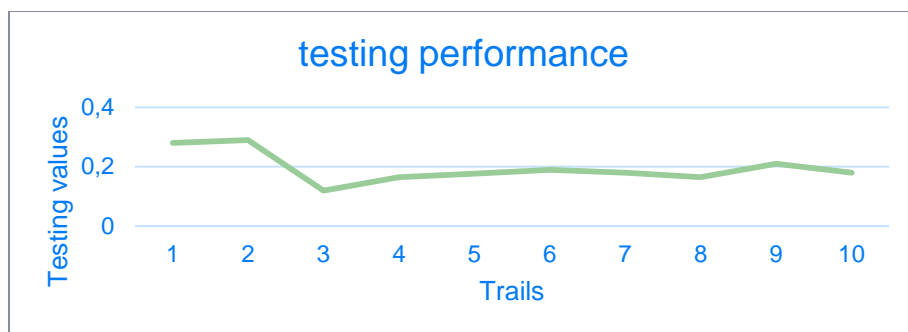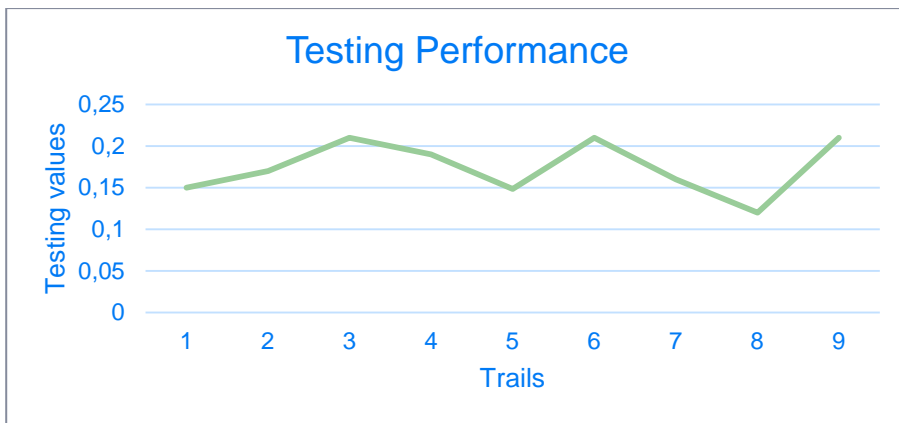

Figure 5.18: Testing performance of Experiment 2 by PSO in each trail

The Figure 39 shows the values of testing performance in this experiment which the maximum was in the last trail and the minimum was in trail 8 while the pattern was also vibrated.

Figure 5.19: Best Cost of Experiment 2 by PSO in each trail

The maximum value of best cost is was in trail 3 while the minimum was in trail 7 and 9, However the pattern of best cost in this experiment was vibrated figure 40.
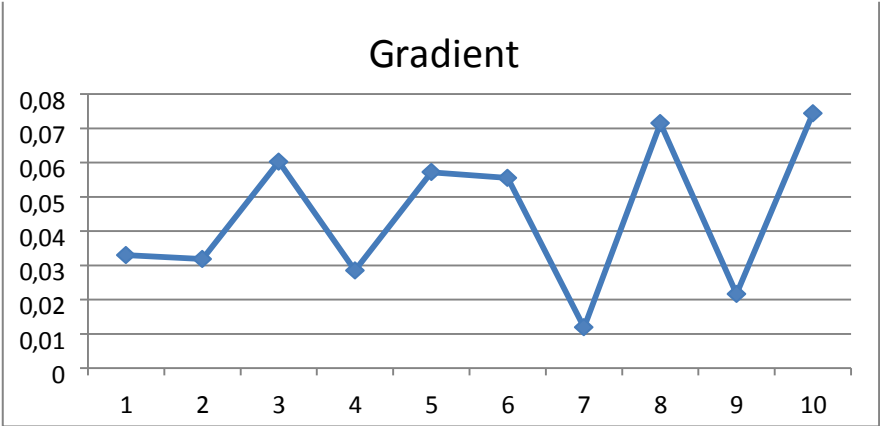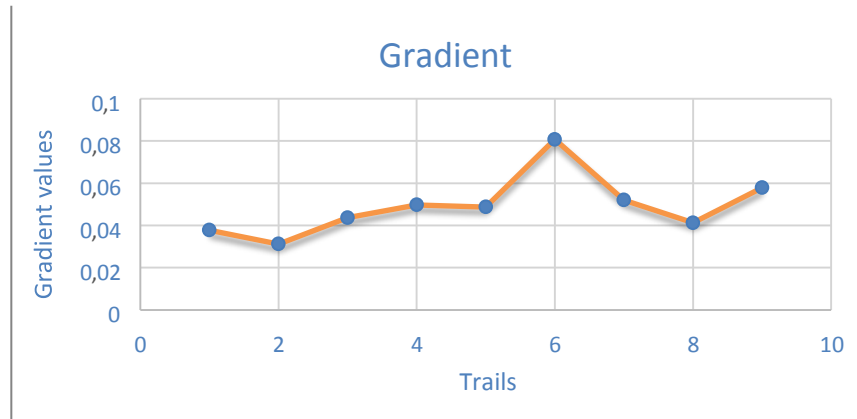

Figure 5.20: Gradient of Experiment 2 by PSO in each trail

The gradient of error is shown in figure above which the maximum value was in trail 6 and the minimum was in trail 2, furthermore the pattern of gradient in this experiment was vibrated.

After finishing all experiments for both algorithms a comparison is done between the results giving by each one using average method, thus the calculation of the average of Testing, training, validation and best cost is done for each trail in each experiment, in order to find out the best solution (optimal features) from the obtained results. The details are as follows:

Table 5.13: The average of validation, testing, training and best cost in each trail of both experiments with PSO algorithm

Trail	Testing samples	Training samples	Features	Average
1	20	60	0 1 1 1 0 1 1 1 0	0,18928
2	20	60	1 0 1 1 0 0 0 0 0	0,29
3	20	60	1 0 1 0 0 0 0 1 0	0,13903
4	20	60	0 0 1 1 1 0 0 0 0	0,13573
5	20	60	1 0 0 0 0 1 0 0 0	0,14112
6	20	60	0 1 0 1 1 0 1 0 0	0,14521
7	20	60	0 0 0 0 1 0 0 0 1	0,13344
8	20	60	0 1 1 1 1 0 1 0 1	0,13537
9	20	60	0 1 0 1 1 1 1 1 0	0,15746

53

10	20	60	1 1 0 0 0 1 1 0 0	0,14346
1	15	70	0 1 1 1 1 0 1 0 0	0,15178
2	15	70	0 1 0 0 1 0 1 0 0	0,13917
3	15	70	0 1 1 1 1 0 1 0 1	0,16289
4	15	70	1 0 1 0 1 1 1 1 0	0,15409
5	15	70	0 0 1 1 1 0 1 1 0	0,13287
6	15	70	0 1 0 1 1 1 1 0 1	0,15356
7	15	70	0 1 0 1 0 0 1 0 0	0,17687
8	15	70	0 1 0 1 1 1 1 1 0	0,14610
9	15	70	1 0 0 1 1 1 1 1 1	0,18035

Table 7 contains the average of results giving in 2 experiments by using PSO algorithm , in order to select the optimal solution from those results , the average of testing ,training ,validation and best cost are taken in each trail of both experiments of PSO ,and the lowest value from the obtained values was taken and its features was selected as optimal solution,.

Hence ,The lowest value of average is obtained in trail 3 of first experiment when 60% of sample for training and 20% for validation are used .the optimal solution giving by using PSO algorithm is "101000010" ,thus 3 features are selected which are : **Asymmetry - Standard deviation - Image summation**

Table 5.5: The average of validation, testing, training and best cost in each trail of both experiments with ICA algorithm

Trail	Testing samples	Training samples	Features	Average
1	15	70	"000111100"	0,1970715
2	15	70	"001111100"	0,126575
3	15	70	"100000110"	0,192145
4	15	70	"010111101"	0,3019
5	15	70	"101111000"	0,17875
6	15	70	"111110111"	0,15295
7	15	70	"101101001"	0,1539
8	15	70	"111010111"	0,135955
9	15	70	"001111110"	0,14034
10	15	70	"111110100"	0,15579
1	20	60	"001110001"	0,17892
2	20	60	"100101111"	0,13783
3	20	60	"111111100"	0,15792
4	20	60	"000101000"	0,1627545
5	20	60	"010101100"	0,150065
6	20	60	"100111001"	0,145665
7	20	60	"101010111"	0.128645
8	20	60	"000100111"	0,14139
9	20	60	"101010010"	0,19315

Table 8 contains the average of results giving by ICA algorithms; the same calculation is done as with PSO, in order to select the best solution from the obtained results.

As the table shown the lowest value of average is giving in trail two (0.126575), thus according to ICA algorithm the best solution is located in trail 2. ("001111100"), the selected features are:

Standard deviation - Sharpness - Root mean square –Variance - Entropy

- **Final result**

After comparing the lowest average giving by both algorithms, consequently the optimal features according to ICA algorithm are:

001111100-**Standard.Deviation, Sharpness, Root-mean-square, Variance** and **Entropy**

And according to PSO algorithm 3features are selected which are :

101000010-Asymmetry, Standard deviation and **Image summation**
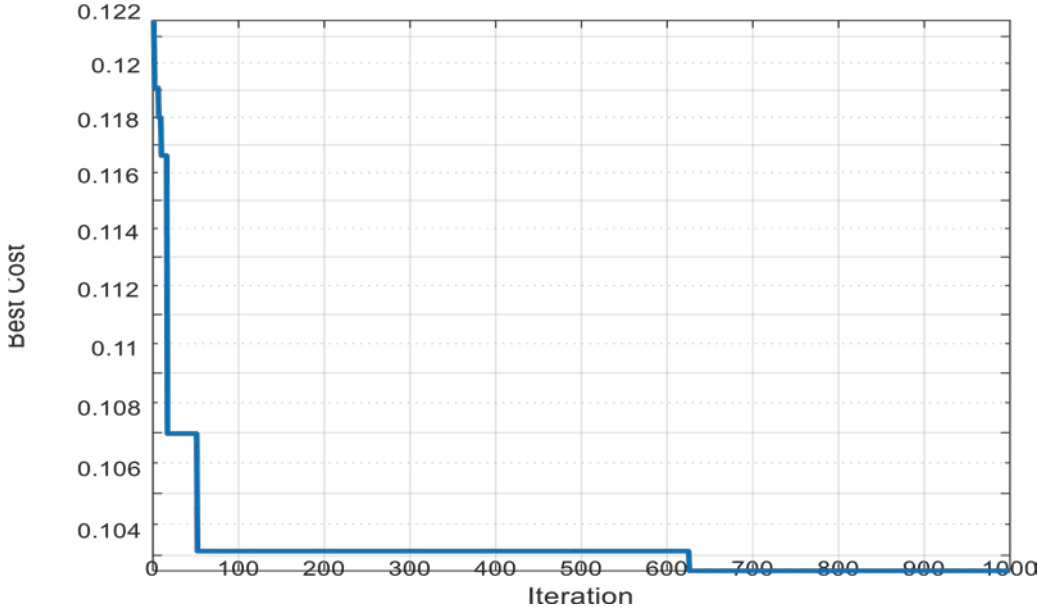
Figure 5.21: ICA mean convergence for the selected features
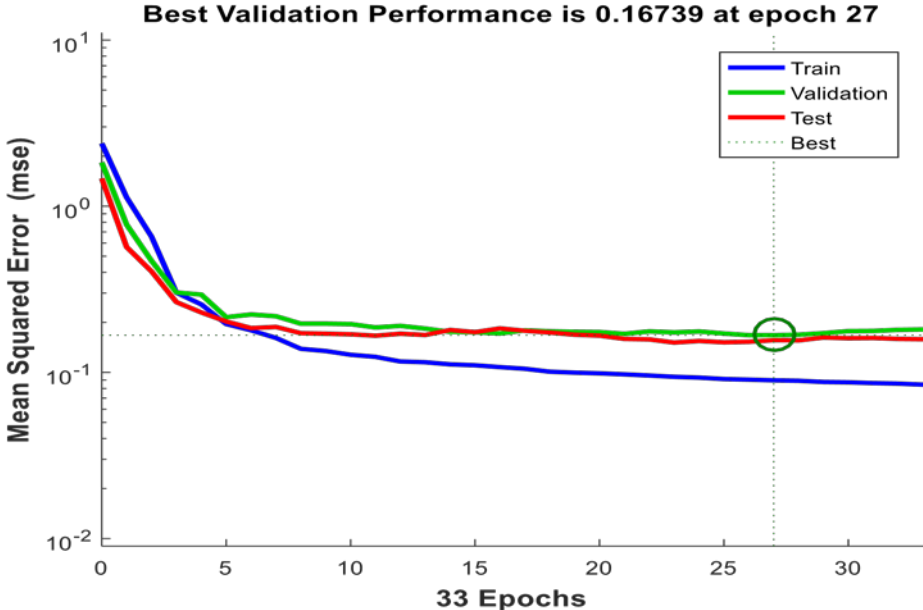
Figure 5.22: ANN training Performance for the selected features

56

After implementing the proposed system, the model detected the optimal number of features which the lowest mean square value of 0.0803 Is reached using ICA algorithm , However the convergence curve of ICA for the selected features is illustrated in Table 9,the best cost value reached during the 1000 iterations is start after the iteration 600 ,the value of cost function was improved when it reached the maximum iteration 1000 Furthermore The mean square values of The ANN is converged before the epochs 30 as shown in Table 10, where the minimum MSE value is reached , this value still fixed until the maximum number of epochs, that means this is the best value that can ANN obtain also in this trail the minimum value of testing performance reached 0.16 was approximately to the validation .

5.2 Accuracy of PSO and ICA

In addition to find a solution to our optimization problem by using optimization algorithm ICA and PSO, the both algorithm was compared according to their accuracy.
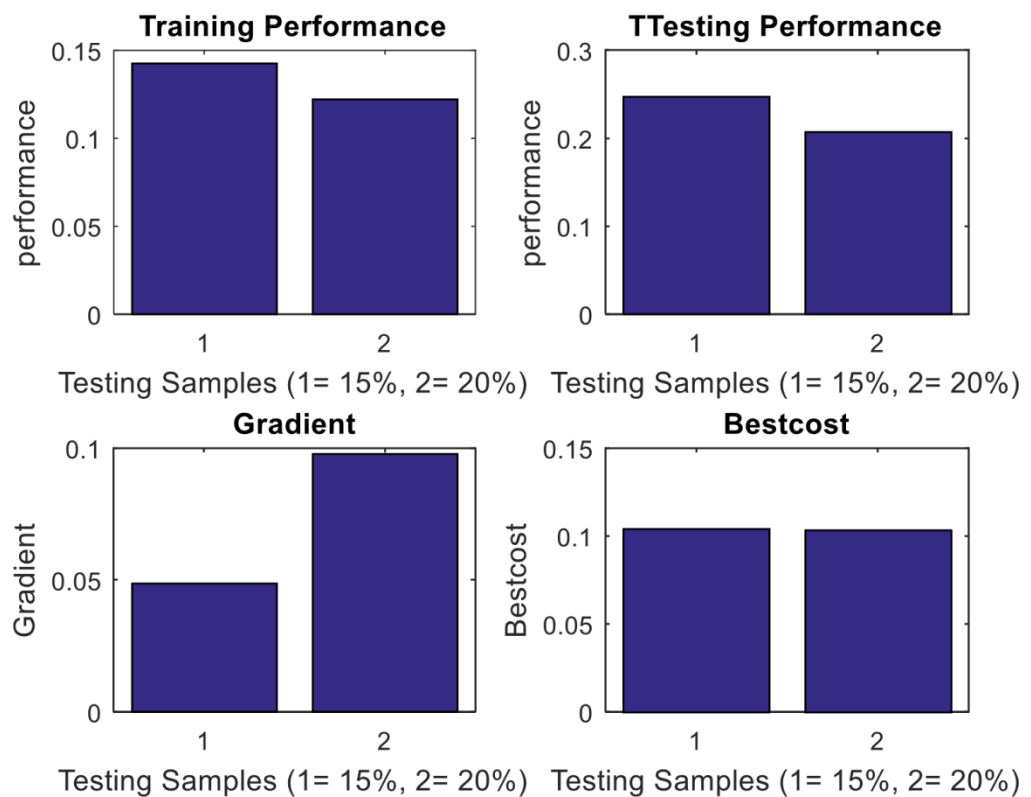
Figure 5.23: The Performance of ICA in each experiment

The figure 44 shows the Performance in both experiment when we used 15 % of samples in testing and also 20 %, we have seen that with 15% of testing the Performance was high, Moreover, the gradient of error was lower in the first experiment according to the second that means the ratio of error was low when the number of testing samples was 15 %. Furthermore, the best cost doesn't change a lot although the changes of samples.

We conclude that ICA result does not get affected according to the number of samples used in training or testing; however it is a positive thing in ICA algorithm especially if a low number of samples are implemented, because in other algorithms the results cannot be guaranteed if number of samples was insufficient.
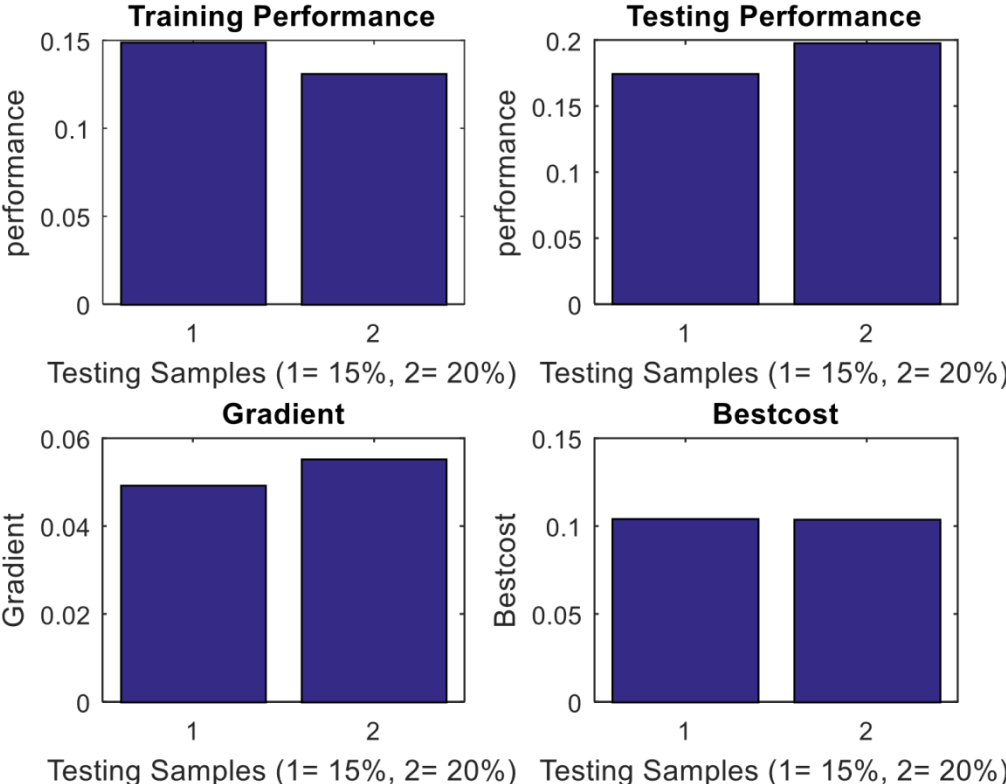
Figure 5.24: The Performance of PSO in each experiment .

The Figure 45 shows the Performance of PSO algorithm, in first experiment when the number of testing samples was 15% and the training was 70%, the Performance of training and testing was high according to the second one when the number of training samples was 60%.and also the gradient of error was lower in the first experiment which means that the ration of error was small in

58

first experiment. Moreover the pattern of best cost was the same in both experiments that can conclude that PSO does not get affected with number of samples used, it can give result even if the number of samples was small reversed other algorithms.

Confusion matrix

A confusion matrix method is applied to determine the image classification accuracy of algorithms used in this study. Hence the confusion matrix values is high when a combination of features giving by ICA (Table 10)and PSO (Table 11) are used

Table 5.6: Confusion Matrix using 9 features

N=73	1	2	3	4	5	Classification overall	Producer Accuracy (Precision)
Class1	11	5	1	1	2	20	55%
Class2	3	16	2	0	2	23	69.565%
Class3	4	0	2	0	0	6	33.333%
Class4	2	1	2	4	0	9	44.444%
Class5	4	3	0	0	8	15	53.333%
Truth overall	24	25	7	5	12	41/73	
Recall	45.833%	64%	28.571%	80%	66.667		

Total Accuracy: 41/73=0.561643=56%

Kappa=0.417

Table 5.7: Confusion Matrix using 5 features (ICA result)

N=73	1	2	3	4	5	Classification overall	Producer Accuracy (Precision)
Class1	14	4	1	0	1	20	70%
Class2	5	17	0	1	0	23	73.913%
Class3	0	0	6	0	0	6	100%
Class4	0	3	0	6	0	9	66.667%
Class5	0	2	0	0	13	15	86.667%
Truth overall	19	26	7	7	14	56/73	
Recall	73.684%	65.385%	85.714%	85.714%	92.857%		

Accuracy: 76.712% Kappa=0.6932

Table 5.8: Confusion matrix using features (PSO result)

N=73	1	2	3	4	5	Classification overall	Producer Accuracy (Precision)
Class1	16	2	1	0	1	20	80%
Class2	2	19	0	0	2	23	82.609%
Class3	2	0	4	0	0	6	66.667%
Class4	0	3	0	6	0	9	66.667%
Class5	0	1	0	0	14	15	93.333%
Truth overall	20	26	5	6	16	59/73	
Recall	80%	76%	80%	100%	82.353%		

Total Accuracy =80.822% Kappa=0.745

5.3 Conclusion and future work

In this study a lots of techniques and algorithms are used to find optimal solution of our optimization problem which is finding the optimal features extracting from the images to be used in detecting leaves diseases and classify them correctly.

in the first chapter of this work ,pre-processing step is done which we have applied some adjustments to our datasets ,after that we have selected k-mean clustering algorithm for segmentation to cluster our images after being transformed to Lab color domain ,our images was divided into 4 classes (colors) , first layer L contains the brightness thus ,our work was focused on the other layers which are the A and B layer those layers contains information about range of color (red ,green, blue and dark brown) ,after segmenting the images we got output with just the region of interest which is the diseases part then the features was extracted and nine ones are chosen to be processed by our designed model. From this part we have concluded that dividing the lab colors images into 4 classes including the layer 2 and 3 from the lab color space gives more results than using three classes, thus the region of disease was clear and was not mixed with the sunlight or the background when 4 classes are used and also using LAB colors is better than RGB.

Furthermore in the second step of this study Two meta-heuristics algorithms are used implementing cascade feed forward NN as cost function to select the best solution from extracted features, the samples of the selected features by the algorithm was the input of our artificial neural network and the 5 classes was the output ,Moreover two experiments are done using our proposed method for each algorithm ,in those experiments our algorithm was tested 10 times and we have collected a lot of results ,in order to find the optimal solution from the results founded we have compared between the results giving by those two algorithms using average method and we calculate the overall accuracy using confusion matrix method.

Consequently, the imperialist competitive algorithm is accurate with an accuracy of 76.71% and overall accuracy of particle swarm optimization was

also high 80% thus we can conclude that both algorithm are accurate and both results can be taking as final result

Moreover the optimal solution or selected features by both algorithms are:

Table 5.9: The combination of features obtained by ICA and PSO

ICA	PSO
Standard deviation-Root Mean Square-Image variation and the entropy of the image	Asymmetry-Standard deviation Image summation

Future work

The proposed possible future work is listed as follows:

-It is possible to increase the datasets, for example using a large number of datasets for training and separate ones for testing

-It is possible to use another algorithm instead of neural network to increase the accuracy and the speed.

-It is possible to use other methods of features extraction instead of methods used in this study.

-It is possible to use other meta-heuristics algorithm as genetic algorithm to find More accurate results if it exists.

- Another possible field to be studied is to use the same methods but on other kind of images like: high spectral images.

-It is possible to apply this method to extract the most suitable features in detecting human diseases on images.

6. REFERENCES

[1]   **Troncoso-Rojas ,R. and Tiznado-Hernández, M. E** "Alternaria alternate (Black Rot, Black Spot)" in Postharvest Decay, 2014. [Online] Available:

https://www.sciencedirect.com/topics/agricultural-and-biologicalhttps://www.sciencedirect.com/topics/agricultural-and-biological-sciences/alternaria-alternat**sciences/alternaria-alternat** [Accessed Nov. 18 2018].

[2]   **Patterson, S.** and **Gardener, M.** "Anthracnose disease info and control – What plants get anthracnose" May 2018 .[Online] Available:

disease.htm [Accessed Nov. 20 2018].

https://www.gardeningknowhow.com/plantproblems/disease/anthracnose-

[3]   **Yen, H. "**Rice bacterial blight**" Nov. 2016** [Online].Available: https://www.britannica.com/science/rice-bacterial-blight   **[**Accessed Nov. 25 2018].

[4]   **Thomas, R.R.** and **Prakash, M. K. C.** "Cercospora fruit /leaf spot: Symptoms of flower" February, 2016. [Online]. Available:

https://expertsystm.wixsite.com/pomegranate/singlepost/2016/02/08/Cercospora
https://expertsystm.wixsite.com/pomegranate/singlepost/2016/02/08/Cercospora-fruit-leaf-spot-5**fruit-leaf-spot-5** [Accessed Nov. 25 2018].

[5]   **Singh, V.** and **Misra, A.K.** "Detection of plant leaf diseases using image segmentation and soft computing techniques" Information Processing in agriculture, vol.4, issue 1, pp.41-49, March 2017 .

[6]   **Ozturk, S.** and **Akdemir, B. "**Automatic leaf segmentation using grey wolf optimizer based neural network" in Proceedings of 2017 Electronics, 19-21 June 2017, Palanga, Lithuania.

[7]   **Sachin, D. K. and Padil, A. B.** "Plant disease detection using image processing" International Conference on Computing Communication Control and Automation 26-27 Feb. 2015, Pune, India.

[8]   **Gulhane, V.A.,** and **Kolekar, M. H.** "Diagnosis of diseases on cotton leaves using Principal Component Analysis classifier "Annual IEEE India Conference, 1113 Dec. 2014, Pune, India.

[9]   **Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D.,**and **Stefanovic, D.** "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification" Computational Intelligence and Neuroscience, vol.2016, Article ID 3289801, 11 pages .2016.

[10]   **Shivkumar, B.Swaranjali,P. Snehal,P .**and **Poonam, P.** "Artificial Neural Network Based Plant Leaf Disease Detection ",International Journal of Computer Science and Mobile Computing, Vol.4 Issue.4, April-, pg. 900-905.2015**.**

[11]   **Alfadhl ,Y.K., Samsuzana ,A. Siti ,K.B., Nazmi ,M.N., Idris ,A.** and **Daniel ,I.** "Early detection of diseases in plant tissue using

spectroscopy" – applications and limitations, Applied Spectroscopy Reviews, 53:1, 36-64, DOI:
10.1080/05704928.2017.1352510.2018.

[12] **Arachchige ,S. A. S. , Andrew, J.R. , David ,W. L. Elizabeth, K.D .** and **Christopher ,S.** "Quantifying the Severity of Phytophthora Root Rot Disease in Avocado Trees Using Image Analysis Remote Sensing"., 10, 226; doi:10.3390/rs10020226.2018.

[13] **Abrham ,D .M. Seffi, G.** and **Dagnachew,M.** March "An Automatic Coffee
Plant Diseases Identification Using Hybrid Approaches of Image Processing and Decision Tree" Vol. 9, No. 3, pp. 806~811 ISSN: 2502-4752, DOI: 10.11591/ijeecs.v9.i3.pp806-811.2018.

[14] **Sivakamasundari, G. and Seenivasagam, V.** "Classification of leaf diseases in apple using support vector machine", International Journal of Advanced Research in Computer Science, vol. 9, no. 1, pp. 261-265, January 2018.

[15] **Jordan ,U. Mikolaj ,C. Przemyslaw, P. and Ian ,S.** "The use of plant models in deep learning: an application to leaf counting in rosette plants "14:6.2018.

[16] **Shoby ,S.** and **Indra ,G. M. P.** "An Efficient Citrus Canker Detection Method based on Contrast Limited Adaptive Histogram Equalization "Enhancement International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 1 pp. 809-815.2018.

[17] **Freedman, P. P** and **Adhikar,T."**The root mean square"2000[Online]. Available**:http://www.analytictech.com/mb313/rootmean.htm** [Accessed December. 16.2018].

[18] **Brown, S**. **"**Measure of shape skewness and kurtosis", 2018 [Online]. [Available] https://brownmath.com/stat/shape.htm. **[Accessed December. 25 .2018].**

[19] **Kenton, W. "**Kurtosis" 2019 [Online]. [Available]: https://www.investopedia.com/terms/k/kurtosis.asp [Accessed December. 17.2018].

[20] **Zarco-Tejada , P.J. Hornero , A. Hernández-Clemente . R.** and **Beck ,P.S.A.** "Understanding the temporal dimension of the red-edge spectral region for forest decline detection using high-resolution hyperspectral and Sentinel-2a imagery" in ISPRS Journal of Photogrammetry and Remote Sensing,2018.

[21] **Gonzalez, R.C. Woods, R.E.** and **Eddins ,S.L.** Digital image processing using MATLAB*,* New Jersey, Prentice Hall, 2003, Chapter 11[Online]
[Available]:https://www.mathworks.com/help/images/ref/entropy.html**[Accessed November. 17.2018].**

[22] **Intensity: Get standard deviation** [Available] http://www.fon.hum.uva.nl/praat/manual/Intensity__Get_standard_deviation___.ht m l

[23] **Vijayarekha, K.** "Features extraction" School of Electrical and Electronics Engineering SASTRA University, Electronics & Communication Engineering.

[24]   **Kennedy, J., and Eberhart, R.** "Particle Swarm Optimization" In Proceedings of   ICNN'95 - International Conference on Neural Networks 27 Nov.-1 Dec. 1995, Australia, Australia.

[25]   **Esmaeil Atashpaz** and **Gargari, Caro Lucas** "Imperialist competitive algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition 2007.

[26]   **Arti,N. Rathod, B. T .and atsal ,S.** "Image processing techniques for detection of leaf disease" Vol 3, Issue 11, November 2013.

[27]   **Sghair, M.E. Jovanovic, R.** and **Tuba, M.** "An Algorithm for Plant Diseases

Detection Based on Color Features" International Journal of Agricultural Science Vol 2, 2017.

[28]   **Padmavathi, K.** and **Thangadurai, K.** "Implementation of RGB and Grayscale Images in Plant Leaves Disease Detection – Comparative Study" Vol 9(6), DOI: 10.17485/ijst/2016/v9i6/77739, February 2016.

[29]   **Monishanker, H. Ananya ,S.** and   **Habibullah ,B.** "plant disease detection by image processing" Department of Computer Science and Engineering, Jashore University of Science and Technology, Jashore-7408. FEB-2019

[30]   **Prajakta ,M . Priyanka, P .Mira, N .riyanka ,B. Padwal ,M.** and **Priti ,N**

."Leaf Disease Detection and Prevention Using Image Processing using Matlab" Volume 02, Issue 02; February– 2016

RESUME

<div align="center">

LAMYAE GOUBRAIM
23 years' old
AMAL 14 N° 48 HAY EL KHEIR CYM RABAT.
0653032880.
05513857147
Lamyae.goubraim@gmail.com

</div>

Education

- 2016 -2019: Master degree in Electronic and Electrical engineering "Istanbul Aydin University "IAU Istanbul TURKEY"
- 2015 -2016:3rd year in the Higher institute of Applied Engineering "IGA RABATAGDAL"
- 2014-2015 :2nd year in the Higher Institute of Applied Engineering "IGA RABATAGDAL"
- 2013-2014 : 1st year in the Higher Institute of Applied Engineering "IGA RABAT-AGDAL"
- 2012-2013: Baccalaureate diploma in science of life and earth.

Trainings

- F**rom 22 June to 22 July 2015:** Training at DISTRAL MOROCCO COMPANY.
- **From 1 august to 1 September 2016:** Training at POSTE MOROCCO.
- **From 1 august to 1 September 2017:** Training at NOKIA MOROCCO COMPANY.
- **From 1 august to 23 September 2018:** Training at AGAOGLU COMPANY ISTANBUL**.**

Technical Skills

- Word,
- Excel.
- Access.
- Power point.
- C/C++ programming
- MATLAB software
- VB .net, HTML, PHP
- Database**:** ACCESS, MYSQL, SQL SEVER, ORACLE

Languages

- Berber   : Mother tongue.
- Arabic   : Mother tongue
- French   : Fluent
- English   : Speaking, Reading, Writing
- Turkish   : Speaking

Qualities

- Dynamic, rigorous, good communication skills, punctual.

Hobbies

- Reading books, travelling, Music