QUALITY GRADING OF PEA USING ARTIFICIAL INTELLIGENCE

A Dissertation Submitted in Partial Fulfilment of the Requirements for the Award of Degree

of

Master of Engineering

in

Electronics Instrumentation and Control

Submitted by

Amrindra Pal

Regd. No 801151004

Under the Guidance of:

Mr. Nirbhow Jap Singh
Assistant Professor
EIED
Thapar University, Patiala

July 2013

Department of Electrical and Instrumentation Engineering
Thapar University
(Established under the section 3 of UGC act, 1956)
Patiala, 147004, Punjab, India

DECLARATION

I hereby certify that the work which is being presented in the dissertation entitled, **"Quality Grading of Pea Using Artificial Intelligence"** in partial fulfilment of the requirements for the award of degree of masters of engineering in Electronics Instrumentation and Control Engineering submitted in Electronics Instrumentation and Control Engineering Department, Thapar university, Patiala is an authentic record of my own work carried out under the supervision of **Mr. Nirbhow Jap Singh**, Assistant Professor, Department of Electrical and Instrumentation Engineering, Thapar University, Patiala, Punjab.

Date: 12/07/2013

Amrindra

Amrindra Pal

Regd. No. 801151004

I certify the above statement made by the candidate is correct and true to best of my knowledge and belief.

Date:

Nirbhow

Mr. Nirbhow Jap Singh
Assistant Professor
Department of Electrical and
Instrumentation Engineering
Thapar University, Patiala
Punjab

Countersigned by:

S. Ghosh

Dr. Smarajit Ghosh
Head of Department
Department of Electrical and
Instrumentation Engineering
Thapar University, Patiala
Punjab

SKM

Dr. S.K. Mohapatra
Dean of Academic Affairs
Thapar University, Patiala
Punjab

Acknowledgement

I express my deep sense of gratitude and respect to my esteemed and worthy guide **Mr. Nirbhow Jap Singh**, Assistant Professor, Department of Electrical and Instrumentation Engineering, Thapar University, Patiala for their valuable guidance in carrying out this work under their effective supervision, encouragement, enlightenment and cooperation Most of novel ideas and solutions found in the thesis are the result of our numerous stimulating discussions. Their feedback and editorial comments were also invaluable for writing of the thesis. I owe my thanks to **Dr. Smarajit Ghosh,** Head, Department of Electrical and Instrumentation Engineering for his kind support. I owe my special thanks to Smt. Nirja for her help and valuable suggestions throughout this work.

A special thank is to my family and friends for their love and support.

Date:
Place: Thapar University, Patiala

Amrindra
Amrindra Pal
801151004

Abstract

In the recent years, computer vision has emerged as a prospective field, related to the recognition of the object by a computer or machine. The most prominent application fields of computer vision are medical image processing, machine vision, military application and optical sorting. In the presented work, the application of computer vision to extract the features of a pea is explored. Feature related to pea are shape, texture, and color. The present work analyse the object, on the basis of surface areas of the pea, computed from different angle. The quality assigning system based on artificial intelligence is developed. The back propagation neural network is chosen as a quality assigning classifier because of its ability to generate complex decision boundaries. The input to back propagation neural network (BPNN) is range data, consisting of surface areas from different views of object. Surface based analysis technique has advantage that the recognition of object becomes simpler and faster. BPNN uses mean square error as a performance index. The number of hidden layers and number of neurons in a hidden layer are selected on trail basis. The selected network models are simulated with available test data, to evaluate the performance. The result shows the effectiveness of the proposed approach to classify the pea on basis of quality.

Table of Contents

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

Computer vision is a field related to identification of object by computer or robots automatically and analyzes them. Earlier work in the direction of recognition is based on visual ability by processing intensity or range image. Intensity based images are usually degraded by noise or shadow of the image so range image is used in the 3-D recognition model. The three dimensional (3-D) object recognition using computer vision identifies objects in an input image using the structured modelbase. Range image is composed of a finite number of elements, called pixel having particular location and values. It directly gives information about the 3-D object and is not affected by the illumination and reflection.

Computer vision mainly consists of two components, first object recognition and second classification. By obtaining the structure, the range image are classified according to similar patterns, attributes, features and other characteristics, this process is known as classification [32]. Modelbase approach recognizes an object using the geometric features like edge, normal vector, surface patch, relationship of surface patches and so on. The geometric features are extracted from the 3-D object [1].

An object and a model can be represented in terms of graphs, containing nodes and links. The surface of an object is described by segmenting it into surface patches. The surface patches with no shape distortion are selected [1]. Such description can be viewed as an attributed graph, a node denotes features of patches such as area, curvature, texture, moment and so on and link shows relation and distance between two nearby surfaces. The segmentation and description of surface is based on measure of curvature of an object [2]. The conventional method for object recognition is based on kernel surface which is the largest area selected from surface patch and a relationship with the other surface is considered [30]. T. W. Chen employed a different approach to recognize by volume [3]. Recognition of an object by shape has the advantage that no needs of learning and easier to describe an object. Pattern recognition can be defined as a process of identifying structure in data by comparison to known structure; the known structure is developed through methods of classification [32]. But this technique has the disadvantage if orientation (distance between neighboring) of the surface of an object is changed then it may lead wrong recognition.

In object recognition another main part is classification. Classifier or model base can make by the previous information of the object. Some artificial techniques are implemented for classifier making. Selection of perfect or appropriate model is difficult. Some old neural network models are perceptron model, mean field annealing neural network (MFA NN), hopefield model and Backpropagation neural network (BPNN). MFA NN is biased on the hopefield model. Some modern classifications are support vector machine (SVM), particle swarm optimization and some hybrid classification techniques.

The recognition method proposed is detailed as follows. A 3-D object is considered here a pea. Firstly an object is divided into number of segments. Each segment is similar to surface patch. Feathers are extracted from the surface patches and neighbor of patches. Each patch is a kernel surface. If properties of modelbase are matched with the input range data then no occlusion in the image. If image is proper matched with the range data then energy level determined by energy function is converges to the stable state [5]. At optimal matching condition energy has its minimum value. The main problem with MFA, if image is occluded then convergence time may increase. So here another neural network model known as Backpropagation neural network (BPNN) is selected. BPNN minimizes mean square error between target and output. As mean square error goes to zero output tends to target vector. Backpropagation NN is fast as compare to other neural network model.

The prasents work is organized in the five chapters.First chapter is an introduction of computer vision system in recognition field. Recognition is based on visual ability by processing intensity or range image. Computer vision identifies objects in an input image using the structured modelbase. Chapter 2 consists of literature review, presents some classical technique for feature extraction. It also provides guideline to select suitable neural network. Backpropagation neural is best for this application. Chapter 3 summarized the discussion of solution methodology. Application of object recognition in different field is presented. Image of pea is acquired by good quality high resolution camera. Image acquired by camera, which is RGB iamge is coverted into gary scale image to make analysis easier. Edge detection technique is also dicussed. Guideline for modelbase contruction is also summarized.In chapter 4 result are presented. Trained neural network is tested with 50 peas so that the success rate of the network can calculated. After 50 trails success rate is 92%. In chapter 5 concludes the presented work and feature scope is summarized.

CHAPTER 2
LITERATURE REVIEW

Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding images. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image [29]. Computer vision system has two main parts, i) feature extraction, ii) classification. Few of feature extraction methods reported in the literature are summarized below.

2.1 Oshima and Shirai’s method

Oshima and Shirai [6], [7] developed a model based recognition system for planer and curved objects. This system works in two phases. First phase that is learning phase a scene containing a single object is shown at a time. The range data of the scene are obtained by a range finder. The description of each scene is built in terms of properties of regions and relations between them. Object model stores this description. In this model object is represented by augmented graph. A node denotes planer or curved surface and link connecting the node give information about the neighbor surface patch. Matching can be achieved by a combination of data driven and model driven searching process. At first a kernel node, which have largest surface area with no occlusion and distortion is pointed then similar surface in model base is considered as matching candidates. The author proposes a system to recognize stacked objects using range data.

2.2 Grimson and Lozano-Perez

Grimson and Lozano – Perez [27] proposed how local measurement of 3-D positions and surface normal can be used to identify and locate objects from among a set of unknown objects. Models consist of polyhedral objects represented by their planar faces. The information about these faces (such as their equations) and the relations between faces (such as distance) are also computed. Sparse range or tactile data of 3-D objects are used as scene feature. The matching process contains two steps: in the first step, a set of feasible interpretations of the sensory data is constructed. In the second step, the feasible interpretations are verified by a transformation test. Only polyhedral objects or objects with a sufficient number of planer surfaces can be recognized by this system.

2.3 Horn *et al.*'s method

Hon *et al.* employed a multi view extended gaussian image (EGI) to recognize 3-D object [22], [23], [24]. Each model object is represented by its mapping on the Gaussian sphere and each scene object is also represented by an EGI. To constrain the search space, the two EGI's are first aligned along the direction of minimum EGI mass criteria, then a match measure is specified by the comparing the similarities in their mass distribution. The model that maximizes this measure is chosen as the matched model. This method has the advantage that EGI can be computed directly; no complicated description stage is needed. The main disadvantage is that EGI is sensitive to occlusion and is unique for convex object only. Furthermore, when multiple objects are present in single scene, it would be necessary to segment the EGI into regions corresponding to separate object, and it is not clear how to achieve this, except for simple-shaped object.

2.4 David W. Tank and John J. Hopfield

David W. Tank and John J. Hopfield [35] described how several optimization problems can be rapidly solved by highly interconnected networks of simple analog processors. Analog-to-digital (A/D) conversion was considered as a simple optimization problem, and an A/D converter of novel architecture was designed. A/D conversion is a simple example of a more general class of signal-decision problems which we show could also be solved by appropriately constructed networks. Circuits to solve these problems were designed using general principles which result from an understanding of the basic collective computational properties of a specific class of analog-processor networks. We also show that a network which solves linear programming problems can be understood from the same concepts.

2.5 Wei Li and N. M. Nasrabadi

W. Li and N. M. Nasrabadi [35] suggested model-based object recognition technique For each model, distinct features such as curvature points are extracted, and a graph consisting of a number of nodes connected by arcs is constructed. Therefore, each node in the graph represents a feature which has an assigned feature type and a numerical feature value, and an arc between two nodes shows the relationship or compatibility between the features such as distances between feature points. Object recognition is formulated as matching a model graph with an input image graph. A hopfield binary network is implemented to perform a sub-graph isomorphism to obtain the optimal compatible matching features between graphs. The compatibility is defined such that it will tolerate the ambiguity of preprocessed features. The algorithm is also extended to detect one object among several objects which could be touching or overlapping. Some simulation results are shown to evaluate the performance of the system.

2.6 Chong-Huah and Hon-Son Don

Chong-Huah and Hon-Son Don [39] proposed a 3-D moment method has been applied to object identification and positioning. A general theory of deriving 3-D moment invariants is proposed in this paper. The notion of complex moments is introduced. Complex moments are defined as linear combinations of moments with complex coefficients. They are collected into multiplets such that each multiplet transforms irreducibly under 3-D rotations. Using the group-theoretic technique, various invariant scalars are extracted from compounds of complex moments via Clebsch-Gordon expansion. Twelve moment invariants consisting of the second and third order moments are explicitly derived in the paper. They can be used as feature vectors for automatic identification of 3-D objects and CAT images in statistical pattern recognition technique. Vectors which consist of the third order moments can be derived in a similar manner. They can be used to solve the problem of two-way ambiguity in defining the principal axes of a 3-D object, so that the rigid body rotation can be unambiguously determined from the relative orientation of principal axes in two frames. The vector moment forms can also be used in the tensor algorithm for motion estimation.

2.7 Fan *et al.'s* method

Fan *et al.* [2] used segmented surface feature for representing object from different direction. The issues to be addressed relate to the representation of the data and the method used to establish correspondences. This paper presented a system which takes dense range data as input and automatically produces a symbolic description of the objects in the scene in terms of their visible surface patches. The feature extracted from the segmented visible test image, identification is performed in three modules, the scanner, the graph matcher and the analyzer respectively. The scanner collects most identical candidates view for the test object. The graph matcher, perform the comparison between the test object and candidate object for detecting the closeness between them. In the end, the analyzer tries to correct errors occurred in matching process of recognition process.

2.8 David E. Van Bout and Thomas K. Miller

According to David E. Van Bout and Thomas K. Miller [34] graph partitioning is a class of combinatorial optimization problems which are characterized by their large number of interacting degrees of freedom. Simulated annealing (SA) and neural networks are two recent approaches to the solution of problems in this class. Mean field annealing (MFA) applied to the graph partitioning problem. The MFA algorithm had combined characteristics of the simulated annealing algorithm and the Hopfield neural network. MFA exhibited the rapid convergence of the neural network while preserving the solution quality afforded by simulated annealing. The MFA algorithm is developed herein in the context of the graph partitioning problem. The rate of convergence of MFA on graph partitioning problems is 10 to 100 times that of SA, with nearly equal quality of solutions. A new modification to mean field annealing was also presented which supports partitioning graphs into three or more bins-a problem which had previously shown resistance to solution by neural networks. The temperature behavior of MFA during graph partitioning analyzed and shown to possess a critical temperature at which most of the optimization occurs. This temperature is analogous to the gain of the neurons in a neural network and may be used to tune such networks for better performance.

2.9 Eric Bartlett and Anujit Basu

Eric Bartlett and Anijit Basu [2] proposed an artificial neural network (ANN) training schemes require network architectures to be set before training. However, the learning speed and generalization characteristics of ANNs are dependent on their architectures. Thus, the viability of a specific architecture can be evaluated after training. This work seeks to reduce the dependence of ANN capabilities on the preselection of network architectures. The present work described an ANN dynamic node architecture scheme which determines the appropriate number of nodes for a given network by defining an importance function which assigns an importance to each node in the network. Optimizing the network architecture becomes part of the training objective. The back propagation learning algorithm has been implemented with this new dynamic node architecture scheme.

2.10 Tsu-Wang Chen and Wie-Chung Lin

Tsu-Wang Chen and Wie-Chung Lin [3] described the recognition subsystem of a vision system based on constructive solid geometry (CSG) representation scheme. From the image, a description about the scene - precedence graph - is inferred. Object recognition is formulated as matching a scene precedence graph to a model precedence graph which is executed by a constraint satisfaction network. The proposed system realized based on the principle of recognition-by-component which has strong support from psychological experiments. It is claimed that an articulation of shapes into parts (or components) is useful because one never sees an entire shape in one glance and our visual systems do in fact cut surfaces into parts. The primitives used in CSG representation schemes have simple characteristics and are easily recognized by human vision. The object recognition scheme proposed in this system can be considered as an engineering implementation of the concept.

2.11 Wei-Chung Lin *et al.*

Wei-Chung Lin *et al.* [30] proposed a hierarchical approach for solving the surface and vertex correspondence problems in multiple view based three dimensional object recognition system. This proposed technique provides a more general and easy formulation of the problem and a more appropriate solution for parallel implementation. At the coarse search stage, the surface matching scores between the input image and each object model in the database are computed through a hopfield network and are used to select the candidates for further consideration. At the fine search stage, the object models selected from the previous stage are fed into another hopfield network for vertex matching. The object model that has the best surface and vertex correspondences with the input image is finally singled out as the best matched model. The weights on these connections are constrained to be symmetrical.

2.12 A. P. Davignon

A. P. Davignon [40] proposed a segmentation process of range images. Based upon an analogy between step edges and orientation discontinuities, detect surface orientation discontinuities by extrema of second directional derivatives or by zero crossing of third directional derivatives up to the fourth order. Two methods to extract surface orientation discontinuities in range images presented. These methods are based upon the analogy between a step edge and a roof edge.

2.13 Adam *et al.*'s method.

Adam *et al.*'s. [37] suggested an algorithm for image segmentation. Image segmentation algorithm involved 1) a common set of 40 laser range finder images and 40 structured light scanner images that have manually specified ground truth and 2) a set of defined performance metrics for instances of correctly segmented, missed, and noise regions, over- and under segmentation, and accuracy of the recovered geometry. A tool is used to objectively-compare machine generated segmentation against the specified ground truth. Four research groups have contributed to evaluate their own algorithm for segmenting a range image into planar patches.

2.14 Martin T. Hagan and Mohammad B. Menhaj

Martin T. Hagan and Mohammad B. Menhaj [33] proposed marquardt algorithm for nonlinear least squares and incorporated into the Backpropagation algorithm for training feed forward neural networks. The algorithm compared with a conjugate gradient algorithm and a variable learning rate algorithm. It is found that the marquardt algorithm is much more efficient than either of the other techniques when the network contains no more than a few hundred weights.

2.15 Steven Gold and Anand Rangarajan

Steven Gold and Anand Rangarajan [36] proposed a graduated assignment algorithm for graph matching, which is fast and accurate even in the presence of high noise. By combining graduated no convexity, two-way (assignment) constraints, and sparsity, large improvements in accuracy and speed are achieved. Its low order computational complexity and robustness in the presence of noise offer advantages over traditional combinatorial approaches. The algorithm, not restricted to any special class of graph, is applied to sub graph isomorphism, weighted graph matching, and attributed relational graph matching.

2.16 Kil-Moo Lee, Peter Meer and Rae-Hong Park

Kil-Moo Lee *et al.* [38] proposed a novel image segmentation technique using the robust, adaptive least *k*th order squares estimator which minimizes the *k*th order statistics of the squared of residuals. The optimal value of k is determined from the data, and the procedure detects the homogeneous surface patch representing the relative majority of the pixels.

2.17 David G. Lowe

David G. Lowe [30] developed a recognition system that uses a new class of local image features. The features are invariant to image scaling, translation, and rotation, and partially invariant to illumination changes and affine or 3D projection. These features share similar properties with neurons in inferior temporal cortex that are used for object recognition in primate vision. Features are efficiently detected through a staged filtering approach that identifies stable points in scale space.

2.18 P. N. Suganthan, Eam Khwang Teoh and Dinesh P. Mital

P. N. Saganthan *et al.* [10] suggested an energy formulation for homomorphic graph matching by the hopfield network and a Lyapunov indirect method-based learning approach to adaptively learn the constraint parameter in the energy function. The adaptation scheme eliminates the need to specify the constraint parameter empirically and generates valid and better quality mappings than the analog The proposed Hopfield network with constraint parameter adaptation is applied to match silhouette images of keys. The Hopfield network was able to adapt the constraint parameter in accordance with the characteristics of the scene and model pair. This scheme not only eliminated the need for specifying the model and scene dependent parameter, but also performed much better than the model with fixed constraint parameter value.

2.19 Mahamed G. H. Omran, Ayed Salman and Andries P. Engelbrecht

Mahamed G. H. Omran *et al.* [7] discussed a new dynamic clustering approach, based on particle swarm optimization. This approach is applied to image segmentation. The proposed approach automatically determines the ‘‘optimum’’ number of clusters and simultaneously clusters the data set with minimal user interference. The algorithm started by partitioning the data set into a relatively large number of clusters to reduce the effects of initial conditions. Using binary particle swarm optimization the ‘‘best’’ number of clusters is selected. The centers of the chosen clusters are then refined via the K-means clustering algorithm. The proposed approach was applied on both synthetic and natural images.

2.20 Thomas Serre, Lior Wolf and Tomaso Poggio

Thomas Serre, Lior Wolf and Tomaso Poggio [31] introduced a set of features for robust object recognition. Each element of this set is a complex feature obtained by combining position- and scale-tolerant edgedetectors over neighboring positions and multiple orientations. This approach exhibited excellent recognition performance and outperforms several state of the art systems on a variety of image datasets including many different object categories.

2.21 Ming-Kuei Hu

Ming-Kuei Hu [14] suggested a theory of two-dimensional moment invariants for planar geometric figures. A fundamental theorem is established to relate such moment invariants to the well-known algebraic invariants. Complete systems of moment invariants under translation, similitude and orthogonal transformations are derived. The recognition of geometrical patterns and alphabetical characters are independent of position, size and orientation.

2.22 Yong Li *et al.*

Youg Li *et al.* [1] proposed to improve convergence performance for back propagation neural network. In traditional BP neural network algorithm, the learning rate selection is depended on experience and trial. This technique is based on Taylor’s formula. The functional relationship between the total quadratic training error change and connection weights and biases changes is obtained, and combined with weights and biases changes in BP learning algorithm. The formula for self-adaptive learning rate is given. Unlike existing algorithm, the self-adaptive learning rate depends on only neural network topology, training samples, average quadratic error and error curve surface gradient but not artificial selection.

2.23 Omaima N. Ahmad Al-Allaf

Omaima N. Ahmad Al-Allaf [31] discussed an artificial neural networks (ANNs) especially backpropagation neural network (BPNN) used largely in image processing. BPNN can be used for image compression/decompression. The BPNN required long time to train the neural network with small error A three layered BPNN was designed for building image compression system. The Fast Backpropagation neural network algorithm (FBP) was used for training the designed BPNN to reduce the training time (convergence time) This is done by using different architecture of BPNN by changing the number of input layer neurons and number of hidden layer neurons.

2.24 Horaud and Bolles *et al.'s* method

Horaud and Bolles [25] and Bolles *et al.* [26] developed the 3DPO system for recognizing and locating 3-D parts in range data. The model consists of two parts: an augmented CAD model and a feature classification network. The CAD model describes edges, surfaces vertices, and their relations. The feature classification network classifies observable feature by type and size. The system recognizes unknown objects by searching for features that match features for some model. This system has been shown to recognize objects in highly complex scene, but with very few models. Since the models are represented in 3-D, it is very difficult to compute them automatically, therefore a CAD model is required that usually needs help from a user, it also needs a very complex network. Furthermore this system relies heavily on detecting circular arcs and straight dihedral edges, so the shape of the object it can recognize is restricted

CHAPTER 3
SOLUTION METHODOLOGY

The ultimate aim in a large number of image processing applications is to extract important feature from image data, from which a description, interpretation, or understanding of the scene can be provided by the machine. Image analysis basically involves the study of feature extraction, segmentation, and classification techniques. Some applications of computer vision system are listed in the Table below.

Table 3.1 Computer vision applications

S. No.	Applications	Problems
1	Mail sorting, label reading, supermarket-product billing, bank-check processing, text reading	Character recognition
2	Tumor detection, measurement of size and shape of internal organs, chromosome analysis, blood cell count	Medical image analysis
3	Parts identification on assembly lines, defect and fault inspection	Industrial automation
4	Recognition and interpretation of objects in a scene, motion control and execution through visual feedback, harvesting of crops	Robotics
5	Map making from photographs, synthesis of weather maps	Cartography
6	Finger-print matching and analysis of automated security systems	Forensics
7	Target detection and identification, guidance of helicopters and aircraft in landing, guidance of remotely piloted vehicles (RPV) missiles and satellites from visual cues	Radar imaging
8	Multispectral image analysis, weather prediction, classification and monitoring of urban, agriculture, and marine environment from satellite images	Remote sensing

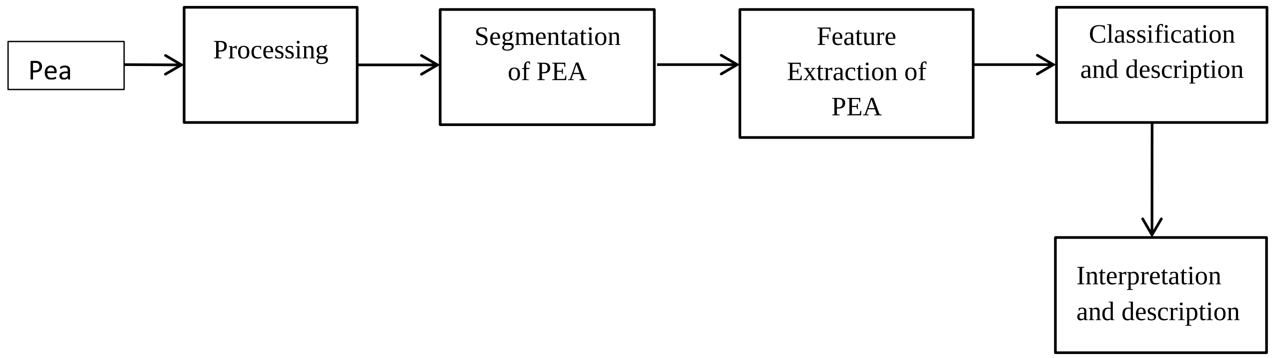


Fig. 3.1 Computer vision system

Computer vision system has two main parts: i) feature extraction and ii) classification. Feature extraction is the process preceding section contains detailed description of feature extraction technique and methods of classification.

Feature extraction process is an algorithmic approach to obtain properties of the image. The steps are summarized in the flow chart shown in Fig. 3.3. Detailed description of the whole process is given below.

Image is acquired by digital camera of sony with 18.2 mega pixel. This camera uses CMOS sensor with extra high sensitivity. Image acquired by camera is RGB image. An RGB color image is an $M \times N \times 3$ array of color pixels, where each color pixel is a triplet corresponding to the red, green, and blue components of an RGB image at a specific location [19].

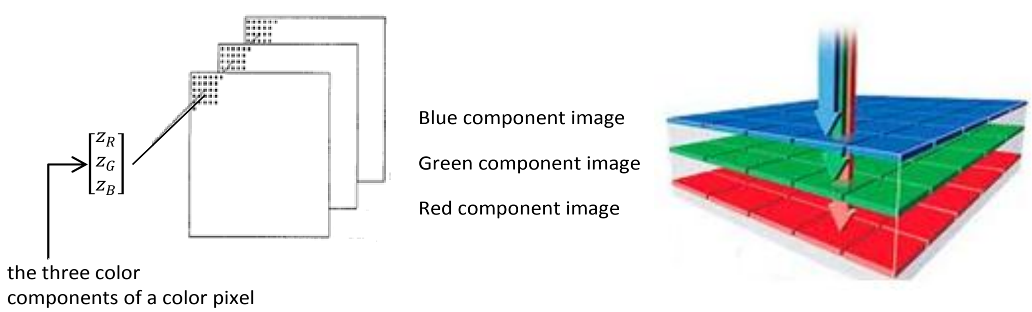


Fig. 3.2 RGB pixel orintation

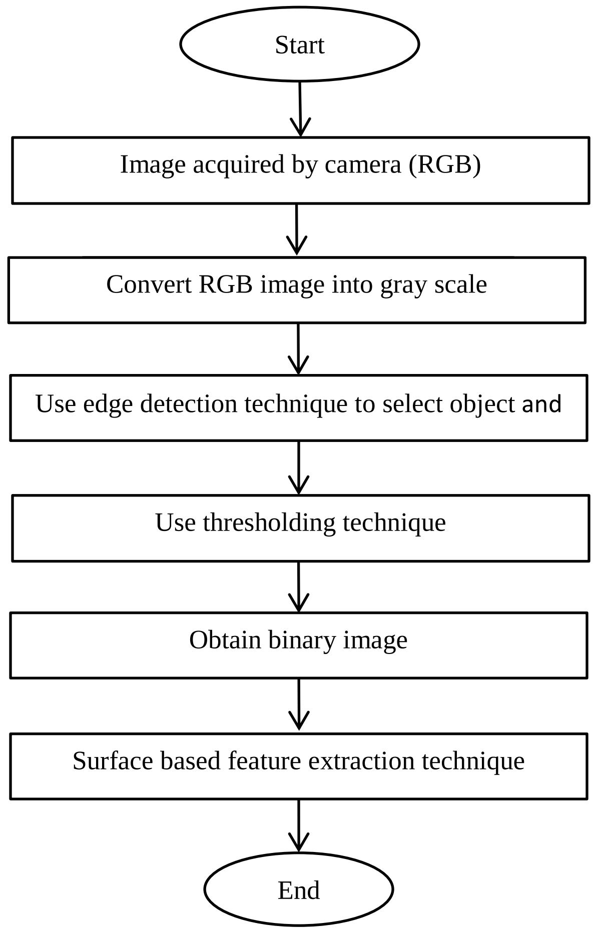


Fig. 3.3 Flow chart for feature extraction

Fig. 3.4 RGB image of pea

The analysis of the RGB image is very difficult because RGB color pixel is 3 dimensional. To resolve this color image can be converted into gray scale image. The gray scale can be represented in X-Y plane.

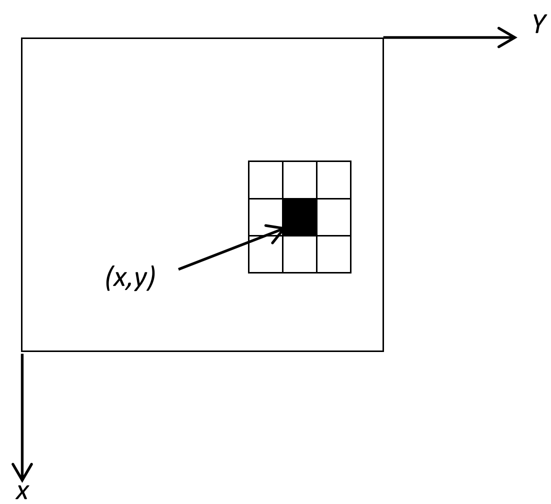
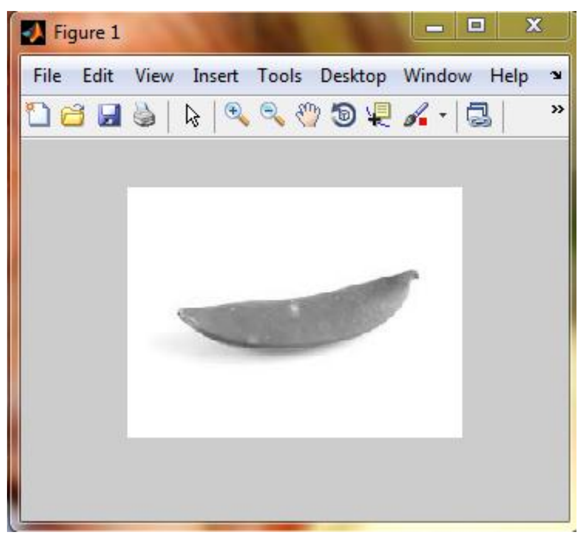


Fig 3.5 Gray pixel centered at point (x, y)



3.6 gray scale image of pea

3.1 Image Segmentation

Segmentation is a process to subdivide an image into its constituents regions or objects. The level of detail to which the subdivision is carried depends on the problem being solved. That is, segmentation should stop when the objects or regions of interest in an application have been detected. For example, in the automated inspection of electronic assemblies, interest lies in analyzing images of products with the objective of determining the presence or absence of specific anomalies such as missing components or broken connection paths. There is no point in carrying segmentation past the level of detail required to identify those element [19].

Segmentation of nontrivial images is one of the most difficult tasks in image processing. Segmentation accuracy determines the eventual success or failure of computerized analysis procedure [19]. The segmentation algorithm is based on one of two basic properties of intensity values: discontinuity and similarity. In the first category, the approach is to partition an image based on abrupt changes in intensity, such as edges. The principal approaches in the second category are based on partitioning an image into regions that are similar according to set of predefined criteria. Thresholding, region growing, and region splitting and merging are examples of methods in this category.

3.1.1 Point, line and edge detection

The focus of this section is on segmentation methods that are detecting sharp, local changes in intensity. The three types of image features in which we are interested are isolated points, lines, and edges. Edge pixels are pixels at which the intensity of an image function changes abruptly, and edges are sets of connected edge pixels. Edge detectors are local image processing methods designed to detect edge pixels. A line may be viewed as an edge segment in which the intensity of the background on either much higher or much lower than the intensity of the line pixels. An isolated point may be viewed as a line whose length and width are equal to one pixel.

a Edge detection

In simple terms, edge detection makes use of differential operators to detect changes in the gradients of the grey levels. It is divided into two main categories: (i) first order edge detection; (ii) second order edge detection. As the name suggests, first order edge detection is based on the use of a first order derivative whereas second order edge detection is based on the use of a second order derivative, in particular, the Laplacian ∇^2.

To detect edges in an image we aim to highlight or emphasize changes in the value of the pixels. Mathematical derivative operations are ideally suited for this purpose. The first derivative, $\partial/\partial x$, shows extremes at an edge and the second derivative, $\partial^2/\partial x^2$, crosses the zero axis where the edge has its steepest gradient. Different types of edges detectors are sobel, prewitt, canny ans roberts edge detector. Robertsedge detector is presented below.

b Roberts edge detector

The masks discussed so far provide a first approximation to computing the gradient of an image in the *x*-and *y*-directions which results in an emphasis on horizontal and vertical lines. Consequently, they are not good at detecting the edges at 45^{o}, for example. The Roberts gradient is based on approximating first order gradients using cross-differences, where the gradient magnitude is given by

$$G_{ij} = \sqrt{\left(f_{ij} - f_{(i+1)(j+1)}\right)^2 + \left(f_{(i+1)j} - f_{i(j+1)}\right)^2} \tag{3.1}$$

which is based on application of the masks

$$D_z = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{3.2}$$

and

$$D_y = \frac{1}{2}\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tag{3.3}$$

However, these masks operate on a relatively small array of pixels and are consequently relatively sensitive to noise. Practical applications of digital gradients such as the Roberts operator usually require pre-processing to reduce the level of noise inherent in an image. Another approach is to consider digital gradients that operate on a larger pixel array [27].

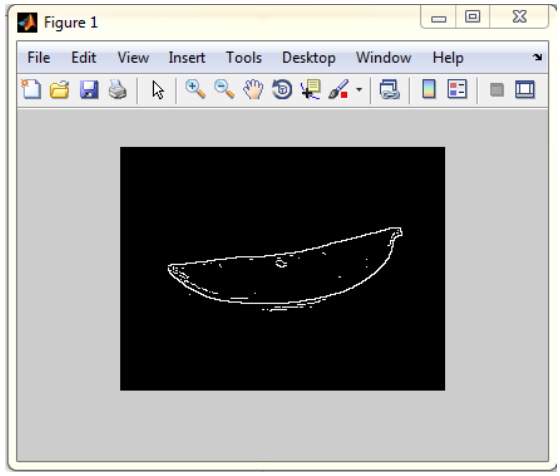


Fig.3.7 Image of pea after edge detection (Roberts)

3.2 Thresholding

Thresholding is a relatively simple approach to segmenting an image into regions of similarity. The basic principle is to group pixels within a common range of grey levels into a predetermined set. The simplest method is the single-band fixed threshold. This is based on first normalizing the image so the pixels have values which lie between [0, 1], i.e.

$$0 \le v_{ij} \le 1 \tag{3.4}$$

where v is the value of the pixel at (i, j) , i and j denotes the location of the pixel in the image. Single-band fixed thresholding converts an image of this type into binary form consisting of just 0's and 1's by application of the following processes:

$$v_{ij}^{out} = \begin{cases} 1; if\ v_{ij}^{in} > threshold \\ 0; if\ v_{ij}^{in} < threshold \end{cases} \tag{3.5}$$

where 0 < threshold < 1. This process can be used to isolate features in an image with a large intensity by applying a threshold which is close to 1. In some cases, it is preferable to retain the grey level variations that occur above the threshold. This is known as semi-thresholding and is achieved by applying the following process:

$$v_{ij}^{out} = \begin{cases} v_{ij}^{in}; if\ v_{ij}^{in} > threshold \\ 0; if\ v_{ij}^{in} < threshold \end{cases} \tag{3.6}$$

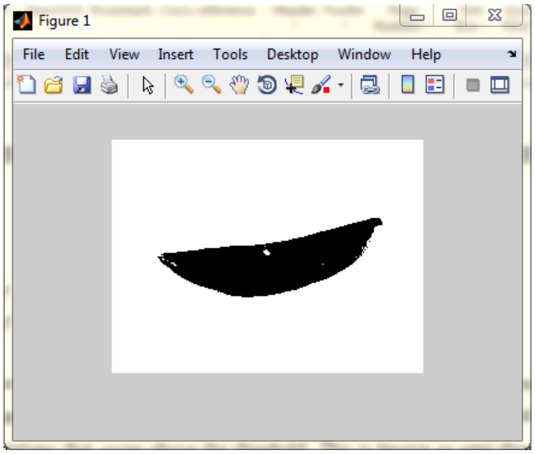


Fig.3.8 Thresholding of pea after edge detection

Figure 3.8 shows an image of pea after thresholding which is a binary image. The advantage of binary image it that pixel in image of two types: i) darker and ii) brighter pixels.

3.3 Feature Extraction

The feature extraction process is a necessary step for modelbase construction. Model base is classifier which trained with the features of the pea. Feature is basically properties of surface or relationship between the neighboring surfaces. This section elaborates the feature extraction process of the pea. Some extraction processes are listed below:

3.3.1 Calculating area by neighboring pixel in image

The area of a gray image can be obtained by counting on pixels in an image by summing the areas of each pixel in the image. The area of an individual pixel is determined by looking at its neighborhood. There are six different patterns, each representing different area. This technique is used for the surface area calculation of the image. In this area computation process brighter pixels are counted [4].

Patterns with zero on pixel (area=0)

Patterns with one on pixel (area=1/4)

Patterns with two adjacent on pixels (area=1/2)

Patterns with two adjacent on pixels (area=3/4)

Patterns with three on pixels (area=7/8)

Patterns with all four on pixels (area=1)

This technique is simple and easy to obtain the area, and not affected by noise.

3.4 Neural networks

The artificial neural networks (ANNs) consist of many connected neurons simulating a brain at work. A basic feature which distinguishes an ANN from an algorithmic program is the ability to generalize the knowledge of new data which was not presented during the learning process. Expert systems need togather actual knowledge of its designated area. However, ANNs only need one training and show tolerance for discontinuity, accidental disturbances or even defects in the training data set. This allows for usage of ANNs in solving problems which cannot be solved by other means effectively [3], [35]. Because of these features and advantages, area of neural network’s application is very wide.

3.4.1 Biological neural network

The human brain consists of a larger number (approximately 10^{11}) of highly connected elements (approximately 10^4 connections per elements) called neurons. The neurons have three principal components: the dendrites the cell body and axon. The dendrites are tree-like receptive networks of nerve fibers that carry electrical signals into cell body. The cell body effectively sums and thresholds these incoming signals. The axon is a single long fiber that carries the signal from the cell body out to other neurons. The point of contact between an axon of one cell and the dendrite of another cell is called synapse [33]

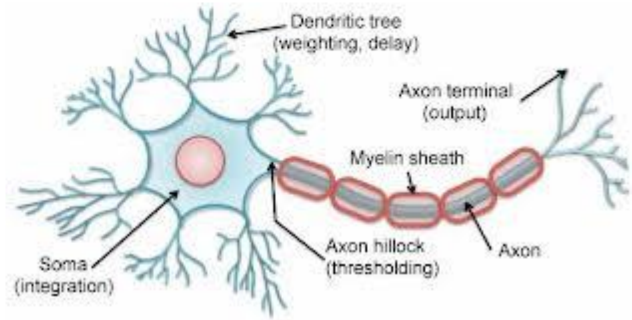


Fig. 3.9 Schematic drawing of biological neurons

Some of the neural structure is defined at birth. Other parts are developed through learning, as new connections are made and others waste away. Artificial neural network do not approach the complexity of the brain. There are, however, two key similarities between biological and artificial neural network. First, the building blocks of both networks are simple computational devices that are highly interconnected. Second, the connections between neurons determine the function of the network.

It is worth nothing that even though biological neurons are very slow when compared to electrical circuits (10^{-3}s compared to 10^{-9}s), the brain is able to perform many tasks much faster than any conventional computer. This is in part because of the massively parallel structure of biological neural networks. Artificial neurons can be interconnected to form a variety of network architecture. A basic neuron model is explained in the preceding section.

3.5 Neuron model

A single input neuron is shown in Fig. 3.10. The scalar input *p* is multiplied by the scalar weight *w* to form *wp*, one of the terms that is sent to the summer. The other input 1, is multiplied by a bias *b* and then passed to the summing point. The corresponding output *n*, often referred to as the net input, goes into a transfer function *f*, which produces the scalar neuron output *a*.

I this neuron model is correlated to the biological neuron model, the weight *w* corresponds to the strength of a synapse, the cell body is analogous to the summation point and the transfer function. The neuron output *a* represents the signal on the axon.

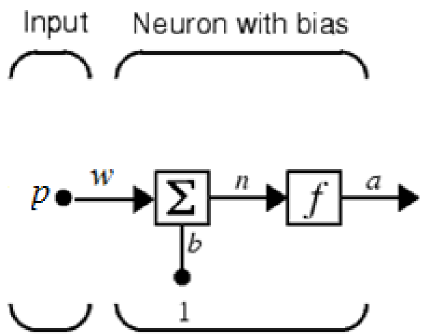


Fig. 3.10 Single input neuron model

The neuron output *a* is computed as

$$a = f(wp + b) \tag{3.7}$$

The actual output depends on the particular transfer function that is chosen. The bias is much like a weight, except that it has constant input of 1. However, if somebody do not want to have a bias in a particular neuron, it can be omitted.

Note that *w* and *b* are both adjustable scalar parameter of the neuron. Typically the transfer is chosen by the designer and then the parameters *w* and *b* will be adjusted by some learning rule so that the neuron input/output relationship meets some specific goal.

The transfer function in Fig. 3.11 may be a linear or a nonlinear function of *n*. A particular transfer function is chosen to satisfy output specification of the problem that the neurons are attempting to solve. Three most commonly used functions are discussed below.

Figure 3.11 (a) shows a hard limit function. This function sets the output of the neuron to 0 if the function argument is less than 0, or 1 if its argument is greater than or equal to 0.

$$a = \begin{cases} 1; if\ n \geq 0 \\ 0; if\ n < 0 \end{cases} \tag{3.8}$$

This function is used to create neurons that classify inputs into two distinct categories. Figure 3.11 (b) shows a linear transfer function. The output of a linear function is equal to its input

$$a = n \tag{3.9}$$

as illustrated in Fig. 3.11 (b). Neurons with this transfer function are used in the ADALINE networks. The log-sigmoid transfer function is shown Fig. 3.11 (c). This transfer function takes the input (which may have any value between plus and minus infinity) and squashes the output into the range 0 to 1, according to expression:

$$a = \frac{1}{1 + e^{-n}} \tag{3.10}$$

The log-sigmoid transfer function is commonly used in multilayer networks that are trained using the Backpropagation algorithm, in part because this function is differentiable.

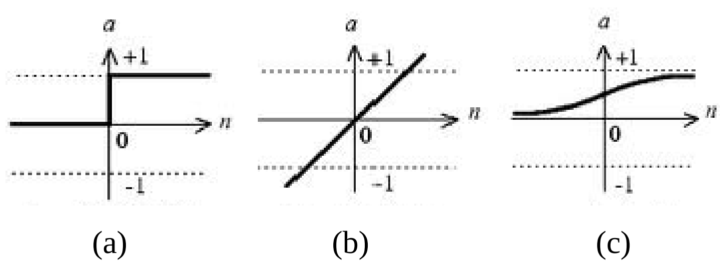


Fig. 3.11 Some useful transfer function

3.6 Perceptron Model

The first accepted neural network model is perceptron. Figure 3.12 illustrates a single-layer perceptron with a hard limit transfer function.

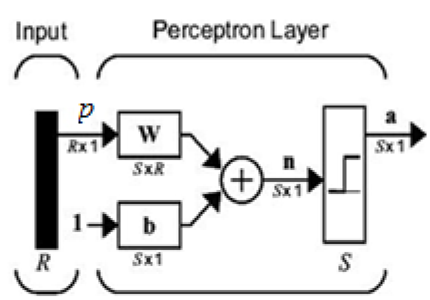


Fig. 3.12 Single-layer perceptron model

where *R* is the number of elements in input and *S* is the number of neurons in the layer.

Perceptron model is able to solve recognition problem. Two input single neuron perceptron (R = 2), which can be easily analyzed graphically. The two input perceptron model is shown in Fig.3.13.

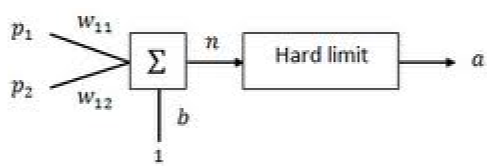


Fig. 3.13 Two input single layer perceptron

Single-neuron perceptron has ability to classify input vectors into two categories. The decision boundary between the categories is determined by the equation:

$$wp + b = 0 \tag{3.11}$$

Notice that this decision boundary will always be orthogonal to the weight matrix and the position of the boundary can be shifted by changing b. There will be one boundary for each row of *w*.

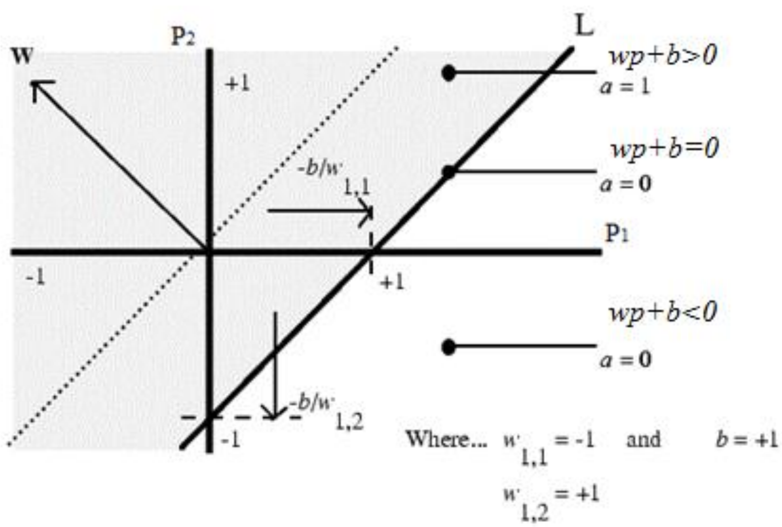


Fig. 3.14 Perceptron decision boundary

Because the decision boundary must be linear, the single-layer perceptron can only be used to recognize patterns that are linearly separable (can be separated by a linear boundary).

Neurons in the multilayer ANNs are grouped into three different types: input layer, output layer, and hidden layer (Fig. 3.15). There can be one or more hidden layers in the network but only one

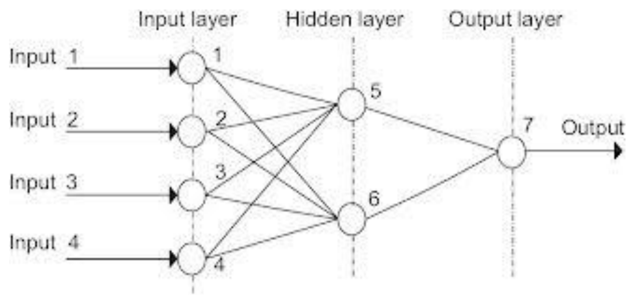


Fig. 3.15 Multi-layer feed-forward neural network

output and one input layer. The number of neurons in the input layer is specified by the type and amount of data which will be provided to the input layer. The number of output neurons corresponds to the type of expected output of the network. The number of hidden layers and their neurons is more difficult to determine. There is no fix rule for the number of neurons in hidden layer.

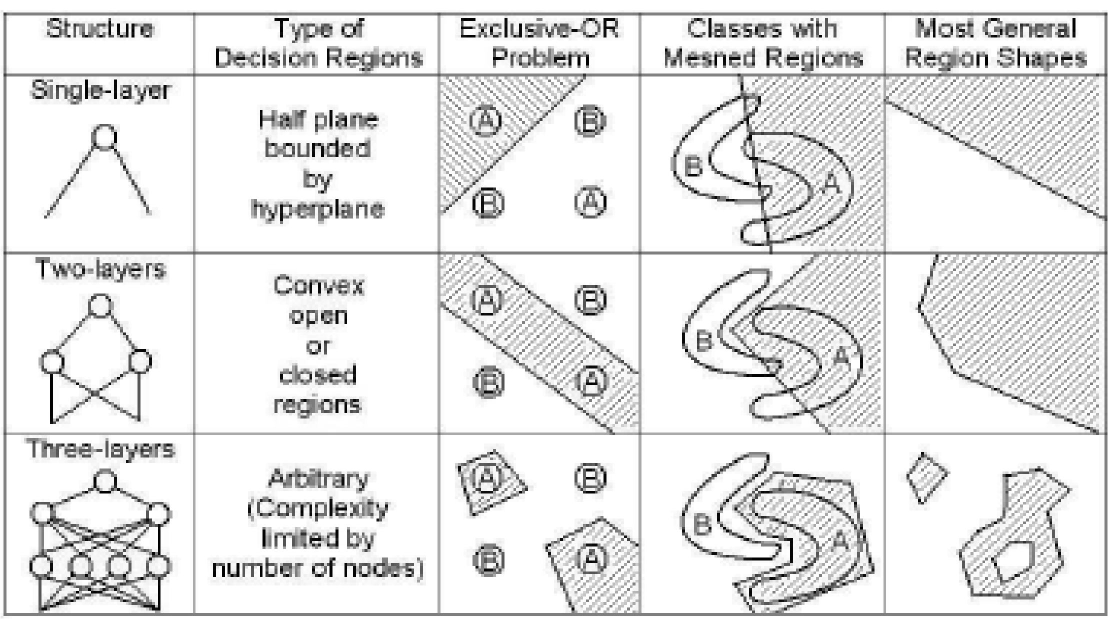


Fig. 3.16 Selection of the number of layers for solving different problems

Two types of a multilayer ANNs can be distinguished with regards to the architecture: feed-forward and feedback networks. In the feed-forward networks signal can move in one direction only and cannot move between neurons in the same layer. Such networks can be used in the pattern recognition. Feedback networks are more complicated, because a signal can be sent back to the input of the same layer with a changed value. Signals can move in these loops until the proper state is achieved. These networks are also called interactive or recurrent [21], [20].

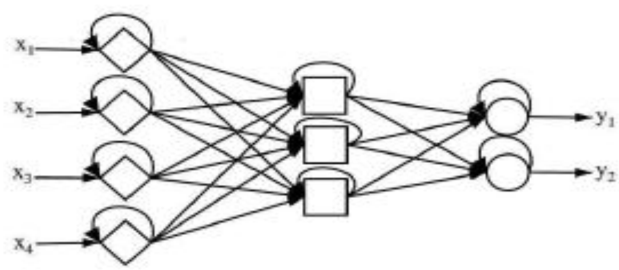


Fig. 3.17 Multilayer feedback neural network

Perceptron model is unable to solve nonlinear problem. So other advanced models of neural network are proposed. One of the well accepted paradigm to solve nonlinear problem is backpropagation neural network.

3.7 Backpropagation Neural Network (BPNN)

The multilayered networks are capable of computing a wider range of functions than networks with a single layer of computing units. However the computational effort needed for finding the correct combination of weights increases substantially when more parameters and more complicated topologies are considered. A popular learning method capable of handling such large learning problem is the backpropagation algorithm. It has been one of the most studied and used algorithms for neural networks learning ever since. This method is not only more general than the usual analytical derivations, which handle only the case of special network topologies, but also much easier to follow. It also shows how the algorithm can be efficiently implemented in computing systems in which only local information can be transported through the network

Backpropagation neural network having supervised learning and the performance index is mean square error [33]. The error expresses in a single layer linear network is a linear function of the weight of the network. For multilayer neural network, the transfer function may be nonlinear, so calculation of error of the network becomes more complex. The Backpropagation neural network (BPNN) algorithm used for training the BPNN for image compression/decompression [31].

3.7.1 Backpropagation algorithm

In multilayer network, the output of the previous layer becomes the input to the following layers: it can be expressed as follows mathematically

$$a^{m+1} = f^{m+1}(w^{m+1}a^{m+1} + b^{m+1}); \ \ (\forall\, m \ \in 0,1,2,\ldots,M-1) \tag{3.12}$$

where M is the number of layers in the network, a neuron's output, f transfer function, w weight matrix and b bias.

The initial layer accepts input as a neuron externally. The first external input is

$$a^0 = P \tag{3.13}$$

provide the starting point for the eq. (3.21). The network output, which is the output of the neurons in the last layer, is written as follows:

$$a = a^M \tag{3.14}$$

3.7.2 Performance Index

Backpropagation use, mean square error criteria as a performance index. This algorithm accepts an input format which is given in eq. (3.24) for proper network behavior

$$\{P_1,t_1\},\{P_2,t_2\},\{P_3,t_3\},\ldots\{P_Q,t_Q\} \tag{3.15}$$

where P_Q is an input to the network and t_Q is the corresponding target output. The output of the network a is compared with target t as input P is applied to the network. The algorithm should adjust the network parameters in order to minimize the mean square error: mathematically it is written as follows

$$F(x) = E[e^2] = E[(t-a)^2] \tag{3.16}$$

where x is the vector of network weights and biases: mathematically represented as

$$x = \begin{bmatrix} {}_1w \\ b \end{bmatrix} \tag{3.17}$$

If the network has multiple outputs this generalizes to

$$F(x) = E[e^T e] = E[(t-a)^T (t-a)] \tag{3.18}$$

In a more general way, the mean square error can be written as below

$$\hat{F}(x) = \big(t(k) - a(k)\big)^T \big(t(k) - a(k)\big) \tag{3.19}$$

$$= e(k)^T e(k) \tag{3.20}$$

where the expression squared error has been replaced by the squared error at iteration k.

The steepest descent algorithm for the approximate mean square error is

$$w_{i,j}^m(k+1) = w_{i,j}^m(k) - \alpha \frac{\partial \hat{F}}{\partial w_{i,j}^m} \tag{3.21}$$

$$b_i^m(k+1) = b_i^m(k) - \alpha \frac{\partial \hat{F}}{\partial b_i^m} \tag{3.22}$$

where α is learning rate.

The stepwise procedure for implementation of BPNN is shown in flowchart in Fig. 3.18

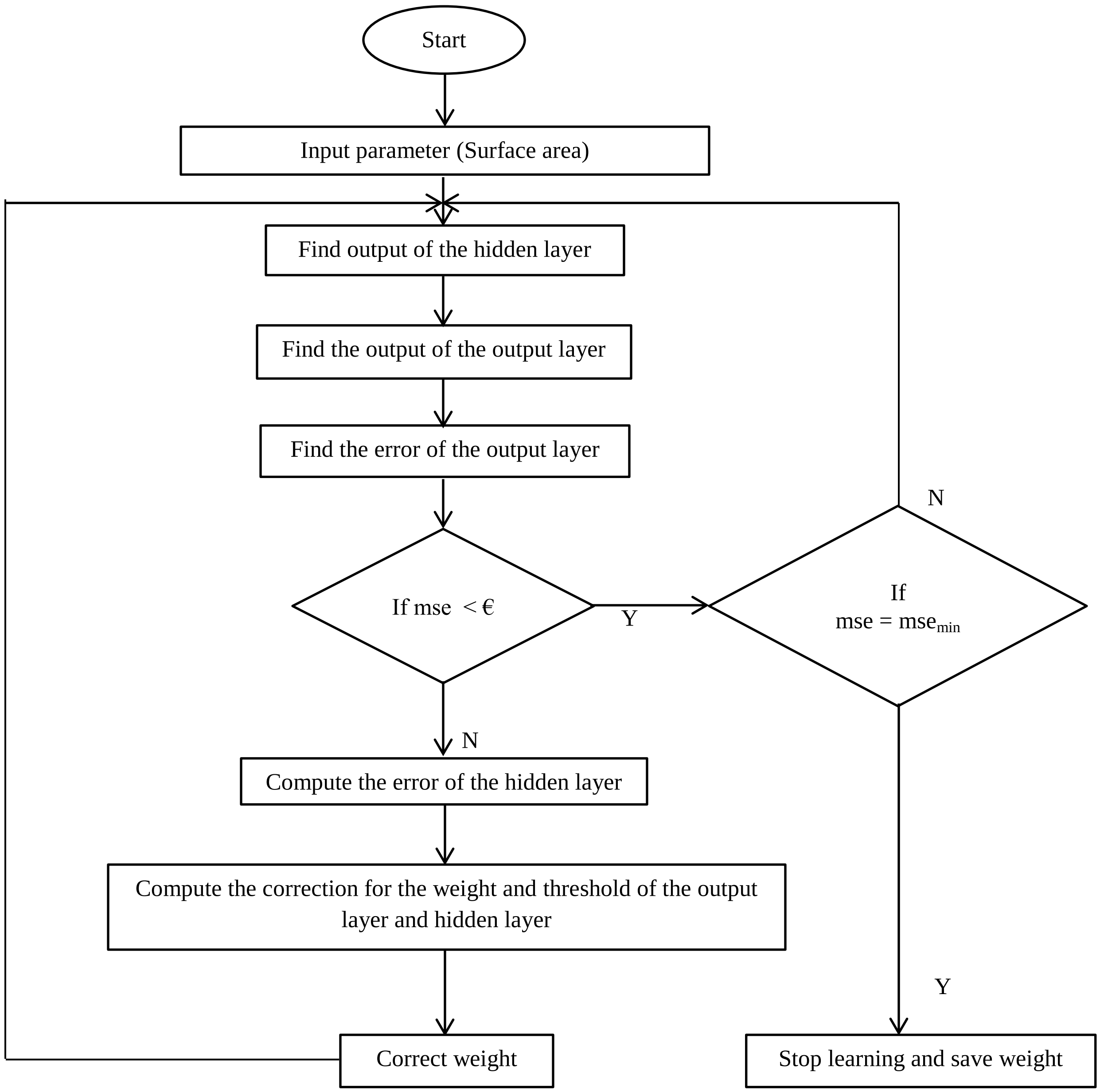


Fig. 3.18 Flow chart for backpropagation neural network algorithm

where € is a small number. MSE is mean square error.

3.8 Model construction

The techniques covered in the literature review for the model construction is based on (1) the analysis kernel surface (2) or an object view in multiple directions. If the selected kernel is properly matched with of the object surfaces, the object recognition process becomes simple and fast. On the other hand selected surface is incorrect or distorted either due to occlusion the matching process becomes difficult.

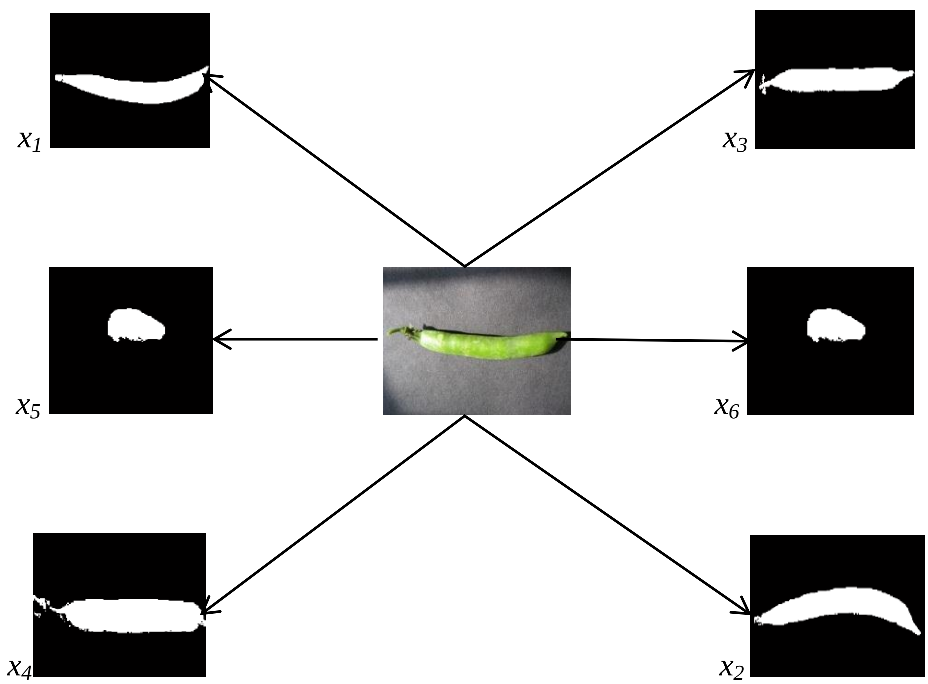


Fig.3.19 Model construction by taking images from different angles

As the recognition process is to be independent of viewing direction. The features of all surfaces are obtained from multiple images from different angles and the information obtained from features is combined to from a model. So that the recognition should be independent of specific viewing direction. Let x_i represent a particular view of subject. So x_i; $\forall\, i \in [1, 6]$ are the six different view of object, taken from different angles. Then 'X' can be represented as follows:

$$X = \begin{cases} x_1 & top\ view \\ x_2 & bottom\ view \\ x_3 & front\ view \\ x_4 & rear\ view \\ x_5 & side\ view \\ x_6 & other\ side\ view \end{cases} \tag{3.23}$$

$$T = [t1, t2, \dots ti];\ \forall\, i = 1,2 \dots 15 \tag{3.24}$$

where T is target vector.

Since more than one subjects are to be investigated. Let X_j represent a particular subject such that

$$X_j = \{x_{1j}, x_{2j}, \dots, x_{ij}\} \; \forall \, i \in [1, 15] \tag{3.25}$$

where *i* is the view and *j* is the subject.

Backpropagation neural network uses performance index mean square error. Number of hidden layers is selected on the trail basis as in following section discussed. Flow chart shown in Fig. 3.20 presents the whole process of feature extraction and classification.

Table 3.2 Network training input vector and target vector

Object no.(*j*)	x_{j1}	x_{j2}	x_{j3}	x_{j4}	x_{j5}	x_{j6}	Target (T)	Output (a)
1	0.80	0.63	0.90	1.0	0.87	0.83	1.	0.96
2	0.98	0.83	0.81	0.53	0.91	0.60	1	0.97
3	0.72	0.18	0.40	0.37	1.0	0.61	1	0.97
4	0.85	0.83	0.55	0.24	0.12	0.33	1	0.95
5	0.86	0.84	0.24	0.21	0.19	0.58	1	0.87
6	1.0	0.98	0.26	0.24	0.05	0.22	0	0.91
7	0.97	1.0	0.85	0.22	0.07	0.23	0	0.97
8	0.21	0.93	1.0	0.29	0.40	0.64	1	0.95
9	0.81	0.77	0.25	0.28	0.10	0.52	1	0.97
10	0.23	0.69	0.28	0.37	0.09	0.34	0	0.93
11	0.91	0.84	0.32	0.27	0.11	0.45	1	0.94
12	0.76	0.85	0.28	0.10	0.01	1.0	0	0.93
13	0.91	0.78	0.31	0.32	0.10	0.51	1	0.94
14	0.86	0.79	0.22	0.22	0.12	0.35	1	0.94
15	0.18	0.76	0.10	0.10	0.05	0.73	0	0.96

Table 3.2 shows the set of data used for fairing of BPNN. The data shown is a collection of different surface area, viewed from six angle.

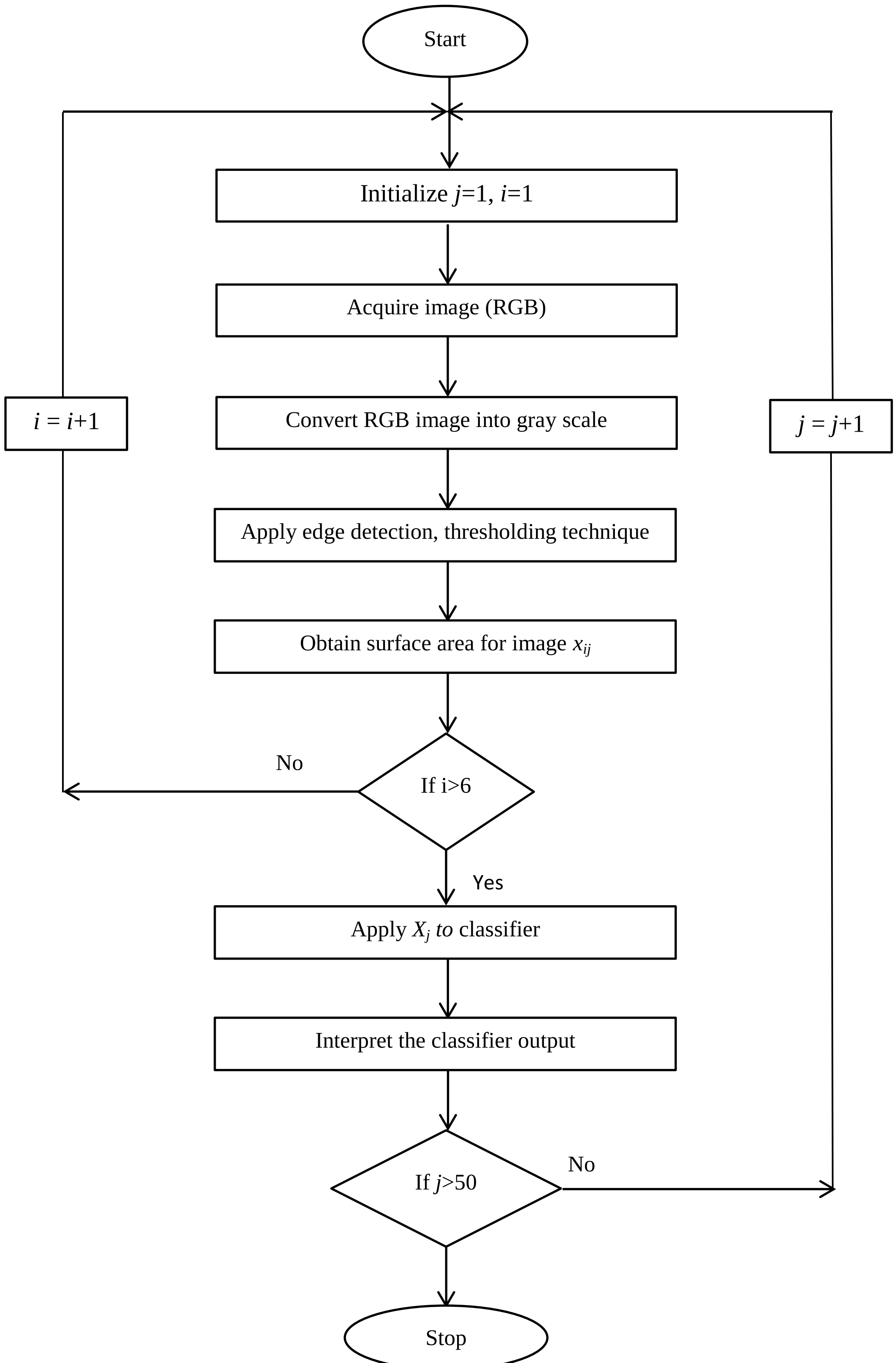


Fig.3.20 Flow chart for feature extraction and classification

Different combination of no. of neuron and no. of hidden layer is shown in Table 3.3. Each case shown in Table 3.3 is trained with input vector and corresponding target; by keeping permissible error between expected target and output of the network less then 0.01.

Table 3.3 Cases to select number of neurons and hidden layers

Case	Number of neuron in hidden layer	Number of hidden layer
1	2	2
2	3	2
3	4	2
4	5	2
5	6	2
6	2	3
7	3	3
8	4	3
9	5	3
10	6	3
11	2	4
12	3	4
13	4	4
14	5	4
15	6	4

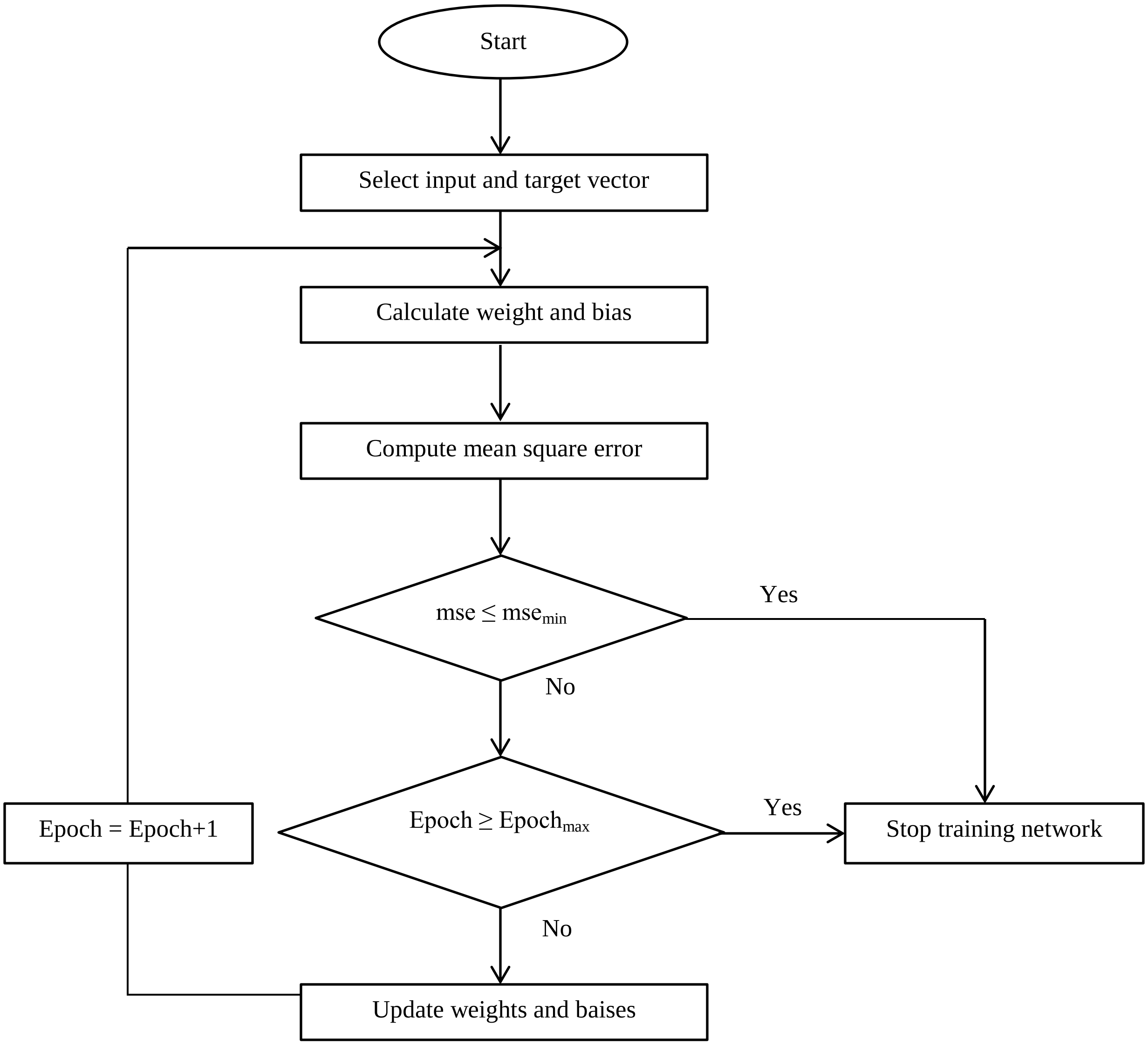


Fig. 3.21 Flow chart for network training

Flow chart shown in Fig. 3.21 presents weight and bias updation during training of the network.

CHAPTER 4

RESULT AND DISCUSSION

In the presented work a network is trained to classify the pea on the basis of quality. The network is tested with available set of data. Different combination of number of neurons and hidden layer are undertaken for study. The various combination of neurons & hidden layers in network are summarized in Table 3.3. The training stastistics are corresponding outputs are summarized in appendix A. It is clear from Table A.1 that the network specified as case 9 and case 15 are capable of classify them. The effect of number of hidden layer are also influencing the performance. These two networks are selected for performance analysis. The response of networks to available 50 sample test data is evaluated and shown in Table 4.1.

Table 4.1 Validation of network using test data

Object no.(j)	x_{j1}	x_{j2}	x_{j3}	x_{j4}	x_{j5}	x_{j6}	Target	Output (Case 15)	Output (Case9)
1	0.95	0.94	0.78	0.79	0.89	0.88	1	0.999	0.998
2	0.21	0.20	0.82	0.89	0.74	0.79	1	0.999	0.997
3	0.85	0.75	0.21	0.20	0.10	0.01	0	0.002	0.007
4	0.40	0.12	0.92	0.85	0.72	0.78	1	0.999	0.891
5	0.53	0.45	0.71	0.60	0.45	0.51	1	0.990	0.859
6	0.31	0.21	0.11	0.10	0.15	0.19	0	0.001	0.002
7	0.87	0.79	0.11	0.19	0.25	0.29	0	0.001	0.121
8	0.80	0.91	0.69	0.62	0.80	0.76	1	0.992	0.999
9	0.42	0.40	0.87	0.80	0.95	0.83	1	0.999	0.972
10	0.98	0.91	0.89	0.79	0.89	0.92	1	1	0.998
11	0.31	0.26	0.83	0.95	0.96	0.89	1	0.998	0.992
12	1.0	0.86	0.46	0.49	0.90	0.93	1	0.999	0.999
13	0.71	0.74	0.60	0.69	1.0	0.93	1	0.990	0.991
14	0.23	0.25	0.31	0.9	0.87	0.89	1	0.997	0.990
15	0.19	0.21	0.76	0.89	0.87	0.83	1	0.999	0.993
16	0.36	0.31	0.54	058.	0.43	0.52	0	0.009	0.011
17	0.55	0.59	0.71	0.78	0.83	0.88	1	0.824	0.900
18	0.98	0.98	0.83	0.89	0.98	0.95	1	1	1
19	0.48	0.42	0.73	0.75	0.89	0.85	1	0.999	0.999
20	0.20	0.21	0.41	0.43	0.82	0.89	1	0.999	0.991
21	0.32	0.36	0.89	0.86	0.90	0.92	1	0.999	0.998
22	0.89	0.86	0.89	0.93	0.86	0.89	1	0.999	0.891
23	1.0	0.98	0.36	0.36	0.23	0.30	0	0.129	0.009
24	0.45	0.46	056	.52	.49	.51	0	0.345	0.007

25	0.95	0.95	0.89	0.88	0.96	0.96	1	1	0.996
26	0.87	0.80	0.83	089	0.80	0.79	1	0.999	0.999
27	0.32	0.32	0.24	0.25	0.89	0.90	1	0.983	0.993
28	0.13	0.12	0.35	0.36	0.99	0.98	1	0.991	0.995
29	0.36	0.32	0.92	0.96	0.89	0.83	1	0.999	0.901
30	0.78	0.73	0.89	0.85	0.90	0.93	1	0.992	0.993
31	0.10	0.12	0.25	0.23	0.12	0.12	0	0.001	0.009
32	0.89	0.83	0.75	0.79	0.80	0.83	1	0.999	0.984
33	0.93	0.90	0.13	0.15	0.20	0.30	0	.009	0.135
34	0.63	0.64	0.59	0.61	0.82	0.83	1	0.786	0.856
35	0.37	0.38	0.65	0.67	0.86	0.82	1	0.999	0.966
36	0.76	0.69	0.78	0.78	0.84	0.81	1	0.993	0.864
37	0.53	0.51	0.65	0.68	0.75	0.73	1	0.999	0.999
38	0.34	0.32	0.68	0.69	0.86	0.89	1	0.999	0.998
39	0.96	0.97	0.82	0.81	0.86	0.89	1	0.999	0.995
40	0.23	0.30	0.64	0.65	0.90	.89	1	0.999	0.864
41	0.80	0.81	0.86	0.89	0.81	0.91	1	0.998	1
42	0.21	0.12	0.31	0.29	0.24	0.21	0	0.012	0.320
43	0.29	0.28	0.65	0.69	1.0	0.98	1	0.999	0.999
44	0.13	0.15	0.31	0.29	0.30	0.31	0	0.045	0.129
45	0.89	0.84	1.0	0.99	0.89	0.85	1	0.999	0.996
46	0.31	0.33	0.46	0.42	0.83	0.84	1	0.989	0993
47	0.51	0.52	0.76	0.81	0.79	0.80	1	0.990	0.993
48	0.21	0.23	0.53	.50	0.74	0.79	1	0.810	0.997
49	0.41	0.42	0.59	0.60	1	0.92	1	0.999	0.999
50	0.31	0.32	0.43	0.44	0.84	0.84	1	0.841	0.912

From the Table 4.1 it is evident that the success rate for case 9 network is 0.72 and for case 15 is 0.92.

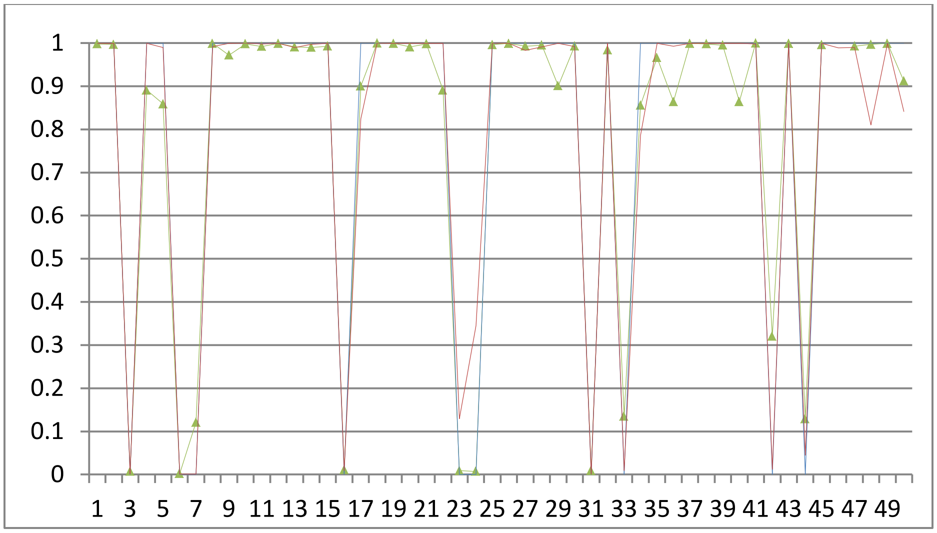


Fig. 4.1 Plot of taget vector ,case 15 network's output and case 9 network's output

The plotted graph between the objects and corresponding target is shown in Fig 4.1.The blue line in the graph shows traget vector, the red line represents case 15 network's output and the green line shows case 9 network's output

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

Quality grading of objects is an essential requirement for decision making purposes. ANN is one of the efficient tool for quality grading and pattern recognition in the presented work, BPNN is deployed for classification of pea on the basis of feature extracted using computer vision. The BPNN is chosen as a quality assigning classifier because of its ability to generate complex decision boundaries. Surface areas based vector is applied to the BPNN as input. The various combinations of hidden layer and number of neurons in hidden layer are experimented, to select an appropriate network for classification for pea. The analysis shows that case 9 (6 neurons and 3 hidden layers) and case 15(6 neurons and 4 hidden layers) will able to classify the subject. Then two networks were simulated with available test data. The result shown that % success rate for case 9 is 72 and case 15 is 92. Hence it is concluded that BPNN with four hidden layers and six neurons are suitable for pea classification purpose.

5.2 Future scope

Recognizing objects is one of the most tasks in vision. While this is an ability that comes naturally in humans. The scope of this presented work is in field of harvesting of crops. Ripe crops are identified on the basis of texture. Corps are recognised by their shape so that the best quality of the crop can be selected. Its another application is in industries for selection of the specified object in many objects.

BIBLIOGRAPHY

Timothy J.Ross, "Fuzzy logic with engineering applications", 2, pp 363,John Wiley (2004).

Yang-Lyual Lee, R.-H.P., "A surface based approach to 3-D object recognition using a mean field annealing network", Elsevier Pattern Recognition, 35(2002), pp 299-316 (2002).

T.J.Fan, G.M.R.N., "Recognizing 3-D objects using surface discriptions", IEEE Trans., pp 1140-1157 (1989).

W.C.Lin, F.-Y.L.C.K.T.T.L., "A hierarchical multiple-view approach to three-dimensional object recognition", IEEE Trans., Neural Networks NN-2, pp 84-92 (1991).

T.-W.Chen, W.-C.L., "A neural network approach to CSG based 3D object recognition", IEEE Trans., PAMI-11(1991), pp 719-726 (1991).

D.E.Van den Bout, T.K.M., "Graph partitioning using annealed neural network", IEEE Trans.Neural Network, 1, pp 192-203 (1990).

Milan Sonka, V.H.R.B., "Image Processing Analysis, and Machine Vision",Thomson (2008).

M.Oshima, Y.S., "A scene description method using three dimensional information", IEEE Trans., Pattern Recognition 11, pp 9-17 (1979).

M.Oshima, Y.S., "Object recognition using three dimensional information", IEEE Trans., PAMI-5, pp 353-361 (1983).

W.E.L.Grimson and Lozano-Prez, "Model-based recognition and locating of 3-D objects", Int.J.RoboticsRes., 3, pp 3-35 (1984).

B.K.P.Horn, "Extended Gaussian Images", IEEE Trans., 72, pp 1656-1678 (1984).

K.Ikeuchi, "Recognition of 3-D objects using the extended Gaussian image", 7th Int.Joint Conf., pp 595-600 (1981).

B.K.P.Horn and K.Ikeuchi, "The mechanical manipulation of randomly oriented parts", Science America, 251, pp 100-111 (1984).

Wei Li, N.M.N., "Object Recognition Based on Graph Matching Implemented by a Hopfield-Style", IEEE Trans., CAS-33, pp 533-541 (1986).

Chong-Huah Lo, H.-S.D., "3-D Moments Forms: Their Construction and Application to Object Identification and Positioning", IEEE Trans., pp 1053-1064 (1989).

David E.Van Den Bout, T.K.M., "Graph Partitioning Using Annealed Neural Networks", IEEE Trans., 1 No. 2, pp 192-203 (1990).

Eric Bartlett, A.B., "A Dynamic Node Architecture Scheme for Backpropagation Neural Networks", Artificial Neural Networks, pp 559-564 (1991).

A.P.Davignon, "Detecting Orientation Discontinuties in Range Images by Use of Dicertional Derivatives", IEEE Trans., pp 455-458 (1992).

Adam Hoover et al., "An Experimental Comparison of Range Image Segmentation Algorithms", IEEE Trans., PAMI-18, pp 673-689 (1996).

M.T.Hagan, H.B.D.M.B., "Neural network design", pp 11.1-11.9 (1996).

Steven Gold, A.R., "A Graduated Assisgnment Algorithm for Graph Matching", IEEE Trans., PAMI-18, pp 377-388 (1996).

Kil-Moo Lee, P.M.R.H.P., "Robust Adaptive Segmentation of Range Image", IEEE Trans., PAMI-20, pp 200-205 (1998).

David G.Lowe, "Object recognition from local scale invarient features", IEEE conf., 2, pp 1150-1157 (1999).

P.N.Suganthan, E.K.T.D.P.M., "Hopfield network with constraint parameter adaptation for overlapped shape recognition", IEEE Trans., NN-10, pp 444-449 (1999).

Mahamed G.H.Omran, A.S.A.P.E., "Dynamic clustering using particle swarm optimization with application in image segmentation", Springer-Verlag London, Pattern Anal Applic-8, pp 332-344 (2005).

Thomas Serre, L.W.a.T.P., "Object recognition with features inspired by visual cortex", IEEE conf., 2, pp 994-1000 (2005).

Ming-Kuei Hu, "Visual Pattern Recognition by Moment Invarients", IRE Trans., pp 179-187 (2009).

Yong Li, Y.F.H.L.S.-W.Z., "The Improved Training Algoritm of Back Propagation Neural Network with Self-Adaptive Learning Rate", Internation Conference on Computational Intelligance and Neural Computing, pp 73-76 (2009).

Omaima N.Ahmad AL-Allaf, "Fast Backpropagation neural network algorithm for reducing convergence time of BPNN image compression", Proc.5th Int.Conf.on IT & Multimedia ay UNITEN Malaysia, (2011).

P.Horaud and R.C.Bolles, "3DPO's stratgy for mtching three dimensional objects in range data", Int.conf.Robotics, pp 78-85 (2013).

R.C Bolles and P Horaud, "3DPO: A three-dimensional part oriantation system", Int.J.RoboticsRes., 5, pp 3-26 (1986).

Rafael C.Gonzalez, R.E.W., "Digital Image Processing", 3 (2011).

Jonathan M.Blackledge, "Digital Image Processing Mathematical and Computational Methods", 1,Horwood Publishing Chichester, West Sussex (2005).

Pratt, W.K., "Digital Image Processing", pp 634,Jhon Wiely & Sons, Inc. (1991).

Christos Stergiou and Dimitrios Siganos, "Neural networks", Surprise, 96 (1996).

S.Agatonovic-Kustrin and R.Beresford, "absic concepts of artificial neural network modeling and its application in pharmaceutical research", Journal of Pharmaceutical and Biomedical Analysis, 22, pp 717-727 (2000).

Appendix A

The details of trails performed to train different cases of BPNN (Table 3.3) with data input are summarized in Table A1. Figure A.1 showes performance of the network.

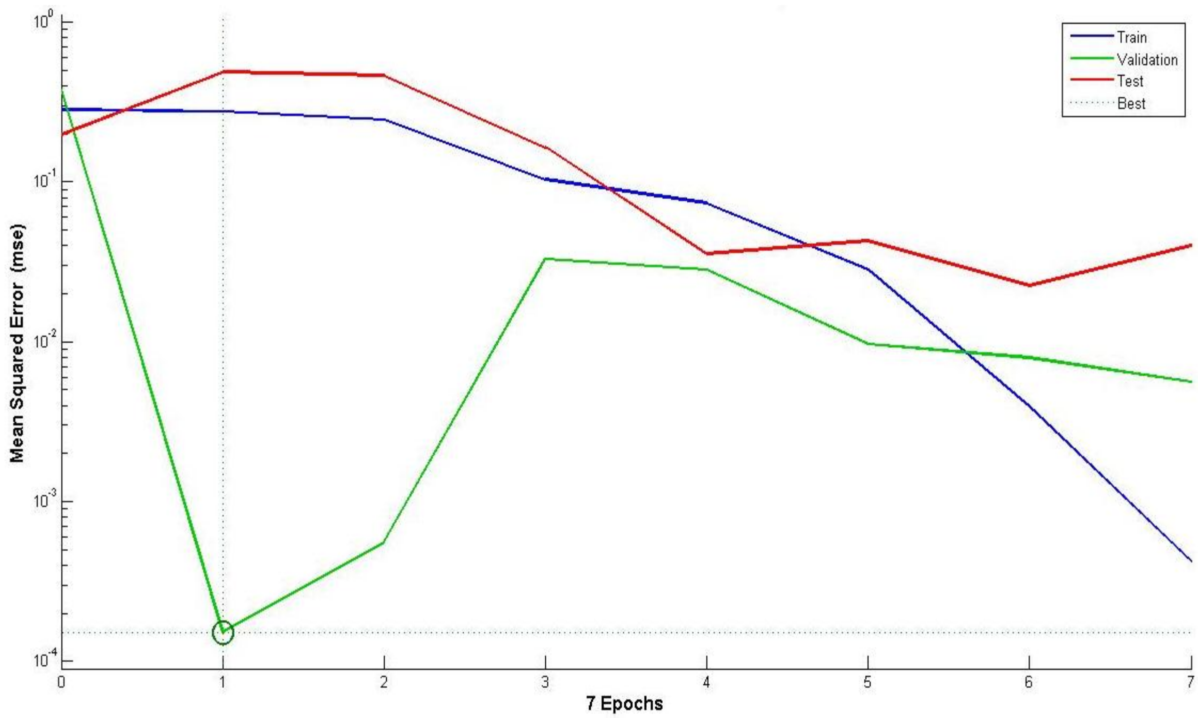


Fig. A.1 Best validation performance

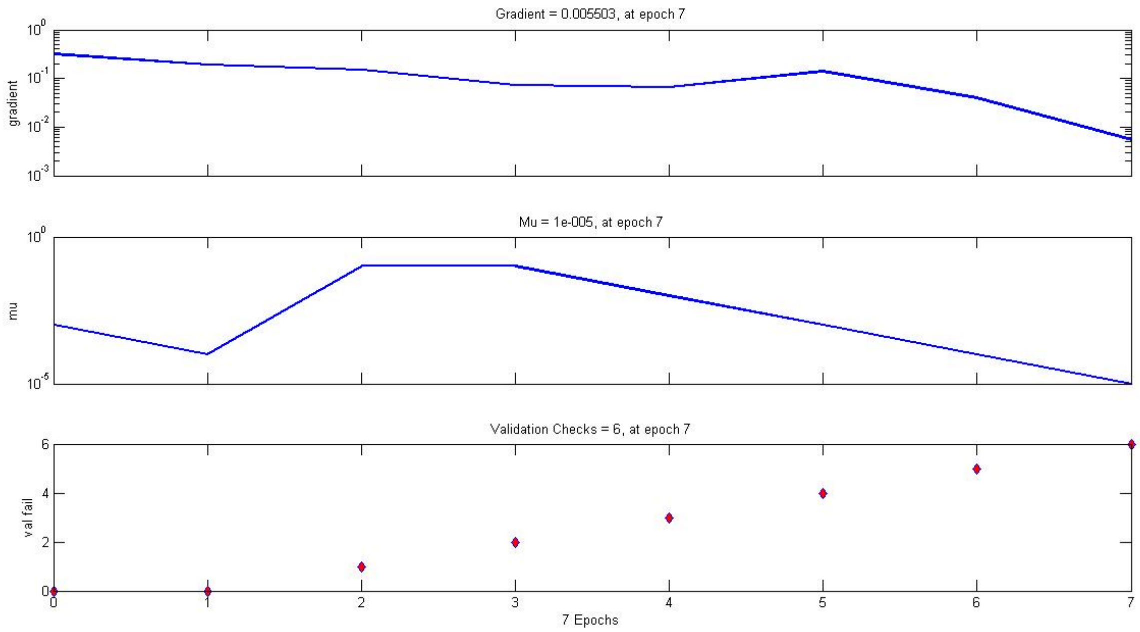


Fig.A.2 Training states of the network

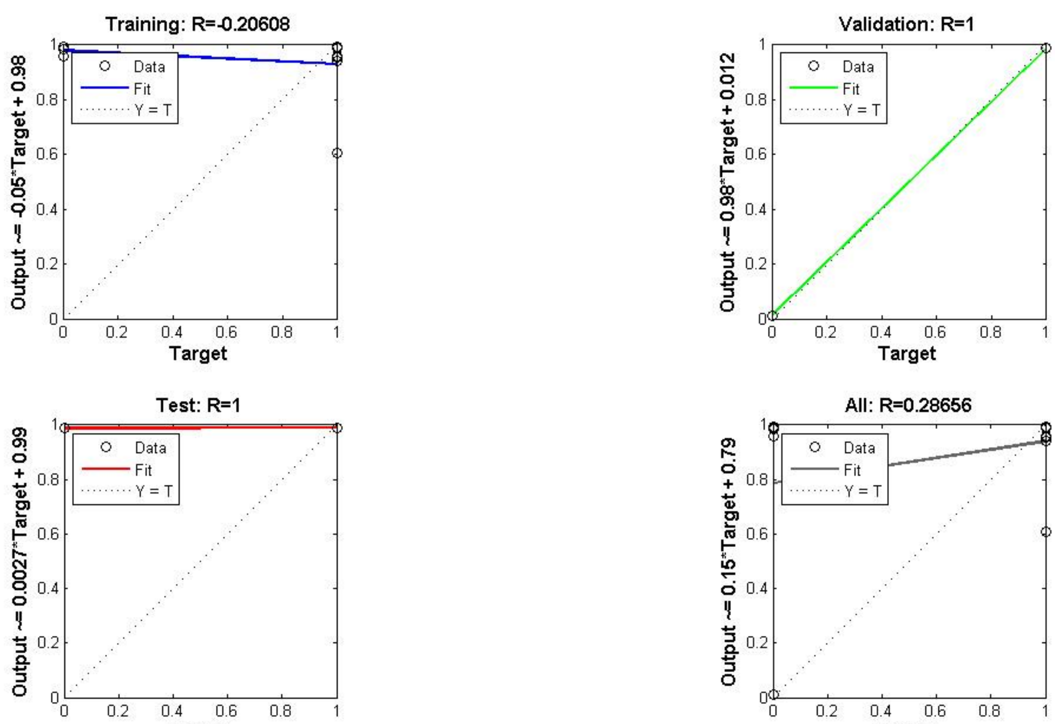


Fig. A.3 Training regration of the network

Table A.1 Comparison of neural network's output

Object	Target	Case1	Case2	Case3	Case4	Case5	Case6	Case7	Case8	Case9	Case10	Case11	Case12	Case13	Case14	Case15
1	1	0.96	1	0.71	0.99	1	0.942	0.957	0.761	1	0.929	1	1	0.9997	0.9838	1
2	1	0.97	1	0.70	0.99	1	0.930	0.967	0.740	1	0.964	1	1	0.9997	0.9726	1
3	1	0.97	1	0.79	0.99	0.99	0.953	0.666	0.617	1	0.7025	1	1	0.8641	0.9729	1
4	1	0.95	1	0.68	0.99	0.99	0.956	0.978	0.495	0.999	0.9391	1	1	0.9997	0.8526	0.9999
5	1	0.87	1	0.29	0.001	0.99	0.956	.973	0.483	0.996	0.9046	1	1	0.9997	0.6036	0.9998
6	0	0.91	1	0.15	0.0008	0.99	0.956	0.980	0.482	0.0008	0.9565	1	1	0.9997	0.0223	0.0007
7	0	0.97	1	0.77	0.99	0.001	0.955	0.979	0.506	0.0011	0.9487	1	1	.09999	0.1771	0.0009
8	1	0.95	1	0.91	0.99	0.99	0.954	0.970	0.746	0.999	0.9318	1	1	0.9992	0.9766	0.9999
9	1	0.97	1	0.27	0.0003	0.99	0956	0.973	0.474	0.999	0.8901	1	1	0.0002	0.8700	0.9999
10	0	0.93	1	0.58	0.99	0.0001	0.956	0.940	0.431	0.0007	0.6052	1	1	0.9997	0.1335	0.0007
11	1	0.94	1	0.31	0.0003	0.99	0956	0.977	0.483	0.999	0.9359	1	1	0.0003	0.5699	09998
12	0	0.93	1	0.60	0.99	0.001	0.956	0.978	0.440	0.0007	0.8130	1	1	0.9997	0.0310	0.0002
13	1	0.94	1	0.27	0.99	0.99	0.956	0.977	0.478	0.9998	0.9250	1	1	0.9997	0.8615	09999
14	1	0.94	1	0.26	0.00021	0.99	0.956	0.972	0.475	0.9998	0.8924	1	1	0.9995	0.7787	0.9999
15	0	0.96	0.0080	0.72	0	0.0005	0.956	0.984	0.147	0.00001	0.3268	1	1	0.9785	0.0001	0.0004
Success Rate(%)		0	73.33	0	46.67	26.67	0	0	0	72	0	66.67	66.67	46.67	6.67	100