SILICON DEFECT RECOGNITION

by

ADITI S GODBOLE

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2012

Acknowledgements

I would like to take this opportunity to thank Dr. Manry for inspiring me to do this thesis. This thesis would have not been possible without his guidance and motivation. I would also like to thank Prof. Davis and Prof. Gibbs for dedicating their time for reviewing my thesis. I would also like to extend my special thanks to Mr. Rohit Rawat and Mr. Kunal Vora for their continuous support throughout the process.

My special gratitude to Mr. Jignesh Patel for sharing his academic work that played a vital role in my thesis. Also, I would like to thank my roommates and friends in UTA for their encouragement and support throughout my thesis.

Last but not the least, I would especially like to thank my parents and my sister for their unconditional love and thoughts of wisdom that play a major role in my accomplishment of this thesis.

December 17, 2012

Abstract

SILICON DEFECT RECOGNITION

Aditi S Godbole, MS

The University of Texas at Arlington, 2012

Supervising Professor: Michael T. Manry

An algorithm is presented for the recognition of four types of defects present in silicon wafer images. Defect recognition is achieved by following a 3-step process: segmentation, feature extraction and classification.

Multiple image segmentation algorithms are tried for locating and isolating the defects present in the silicon wafer images. The proposed image segmentation technique is based on simple concept of threshold based segmentation and edge detection based segmentation. Combination of four segmentation algorithms based on above mentioned techniques are used such that each segmentation algorithm specializes in segmenting a certain type of defect, thereby ensuring high chances of correct segmentation. Out of these segmented images, the most relevant and distinctive features are extracted and used to train an efficient neural network based classifier. For the standard sized images 2D DFT features are calculated and fed into HWO-MOLF classifier that can determine the type of defect present.

Results are presented for all four types of defects.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Since 1954 [1] the world has made immense progress with the use of silicon semiconductors in integrated circuit technology. The presence of defects in silicon wafers can cause failures in the resulting components, due to degraded material quality. Thus it is imperative that the manufactured silicon wafer should have a minimum number of defects. The need to produce hyper pure silicon has made it necessary to detect and recognize the defects in silicon at all the stages of silicon semiconductor production.

Pattern recognition algorithms follow a three step process: data preprocessing, feature extraction and classification. This sequence is widely used, with some variations, in applications such as handwritten digit recognition [2], license plate number detection and recognition [3], automated fingerprint recognition [4], mail sorting [6], target recognition [8] and iris recognition [5].

This chapter, describes various image segmentation techniques, feature extraction and classification methods. Existing techniques for silicon defect recognition and their limitations are discussed and the objective of this thesis is outlined at the end of the chapter.

1.1 Image segmentation

Image segmentation is an operation for finding possible regions or objects of interest in the image. Once these regions or objects are isolated, they are passed on to other processing steps. Image segmentation is a vital stage in automated vision systems. Choice of image segmentation techniques to be used depends on application type, difficulties presented by data, and desired level of accuracy.

In thresholding [11] or threshold based segmentation [9][10] similarity of intensity is taken into consideration. This approach is suitable for a subclass of images in which

objects are distinct from the background in intensity. The drawbacks of threshold based segmentation are; difficulty in adjusting threshold parameters, lack of sensitivity and specificity needed for accurate classification, difficulty in deciding threshold value in cases of images with multimodal histograms.

Region based segmentation [9], is performed by partitioning an image into regions that satisfy certain conditions. Region-based methods often cause segmentation errors at the boundaries. Another problem in region based segmentation is the choice of initial seed points and stopping criteria.

Edges are pixel locations of abrupt luminance change. Edge detection [9] identifies points in a digital image at which these changes occur. Edge detection based segmentation fails if the images are noisy or the edges are not well defined [17] , in other words if the attributes between regions differ only by a small amount.

Other popular methods of segmentations are graph partitioning methods [14], clustering methods [15], and knowledge based segmentation [16] etc.

1.2 Feature extraction

In order to reduce redundancy and dimensionality of the input to the classifier, the process of feature extraction is used. This operation concentrates the most relevant and distinctive information from the input data into a vector of features. Typically features are calculated by using geometrical and structural properties of shapes, the distribution of points in a given image or by using a subset of transform coefficients. The discrete Fourier transform [11], Karhunen- Loeve transform [18] and Haar transform are widely used transforms for the purpose of feature extraction [19]. Successful recognition or classification requires that the feature vector from the same class have similar feature values, even when the images to be recognized are scaled rotated shifted or otherwise distorted. Therefore the chosen feature sets may need to be deformation invariant [30].

1.3 Classification

In a recognition system classifiers make the final decision, in other words, classification addresses the problem of assigning an object into a predefined group or class, based on various characteristics of that object.

Conventional classifiers take a statistical approach towards classification. The nearest neighbor classifier and Bays Gaussian classifier are two widely used examples of conventional classifiers.

Neural network based classifiers are designed such that there is one output for each class and each output is a discriminant for the corresponding class. The multilayer perceptron (MLP) is one of the types of feed forward neural network, which is widely used for approximation and classification.

The most widely used classifiers are support vector machines (SVMs) [31], neural network based classifiers and nearest neighbor classifiers [30]. SVMs can perform linear or nonlinear classification [34].

1.4 Problems in silicon defect recognition

In the problem of silicon defect recognition image segmentation is a crucial step. It is essential to find a right type of image segmentation techniques that give reliable results which will be used for the classification. Existing techniques of image segmentation such as thresholding, edge detection, region growing when used on their own are not suitable for segmenting silicon wafer defect images. Moment features [35] or radius features [30] are not invariant to deformations, such as scale, rotation etc. Thus these feature extraction methods require preprocessing or post processing approaches to introduce invariance to the deformations. SVM classifiers are capable of classifying data into two classes, but its multiclass variation has few advantages.

1.5 Objective of this thesis

The objective of this thesis is to develop a silicon defect recognition system that will segment the defect reliably from the given input images and then classify it correctly. This thesis is organized as follows; Chapter 2 describes silicon defects under consideration and their properties and attributes. Chapter 3 discusses methods that can be used for segmenting the defects and proposes an algorithm for consistent segmentation. In Chapter 4 the classification algorithm used for identification of defects is described. In Chapter 5 results of developed system are represented.

Chapter 2

Defects in Silicon Wafers

Hyper pure silicon doped with boron, gallium, phosphorous or arsenic is used in the production of transistors, solar cells, rectifiers and other solid state devices which are widely used in electronics. The presence of defects in semiconductors disturbs their mechanical, electrical, optical, and magnetic properties. The electrical properties of silicon depend on the impurities and defects present in the silicon. The presence of defects is one of the major reasons for rejection of silicon wafers by silicon manufacturers and IC manufacturers. Thus it is highly desired to identify and to characterize the defects that are most detrimental to silicon devices.

In order to recognize defects through an automated vision system, these defects are required to be captured in a form of an image. The imaging techniques that are used for this are varied. In case of multi-crystalline silicon, imaging is performed by infrared camera which uses free carrier emission [20]. Another example is of imaging via x-ray topography that shows the presence of surface damage on the silicon crystal which makes it easy to differentiate between damage to the entry surface and damage to the exit surface [21].The images used for this thesis are obtained from a scanning electron microscope (SEM). The following section discusses the various defects considered in this thesis.

2.1 Defect types

There are many different types of defects such as extended structures, complexes, native defects or impurities. Type A, type D and crystal originated pits [22] are examples of such defects. Type A-defects are interpreted as clusters of silicon self-interstitials. Type D-defects are formed by clusters of silicon vacancies and crystal-

originated pits are formed through agglomeration of vacancies.[22]. The defects formed by interactions of native defects and impurities are the main concern in this thesis.

2.1.1 Copper ball defect

Copper is an omnipresent contaminant that can be easily introduced into silicon wafers. It is introduced on silicon wafer surface during the process of cleaning or device processing. The presence of copper rich precipitates in Si reduces minority carrier diffusion length [22]. These precipitates can have various morphological shapes. The Cu-defects under consideration in this thesis are the ball shaped copper defects.

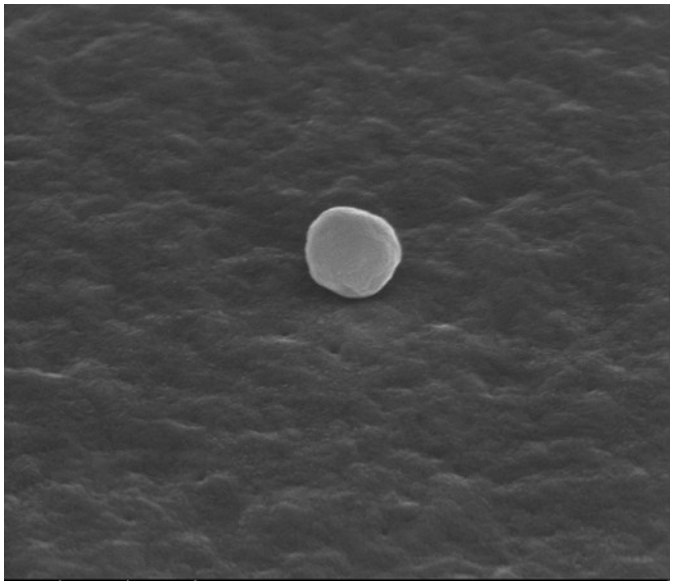


Figure 2-1: Copper ball defect

2.1.2 Plastic defects

Plastic defects in silicon are formed by the adsorption of organic contamination on the silicon wafer surface. It has a damaging effect on the performance and yield of semiconductor devices. When silicon wafers are exposed to the atmosphere in a regular clean room that generally contains polymeric materials, the gaseous organic molecules found in this atmosphere are easily adsorbed onto the wafer surface. Also when the

silicon wafers are stored in plastic boxes; these boxes are made up of polypropylene or polycarbonate materials. The organic gases vaporized from these materials may also be adsorbed onto the surface of a silicon wafer and contribute toward the presence of plastic defects in the silicon wafers [25].

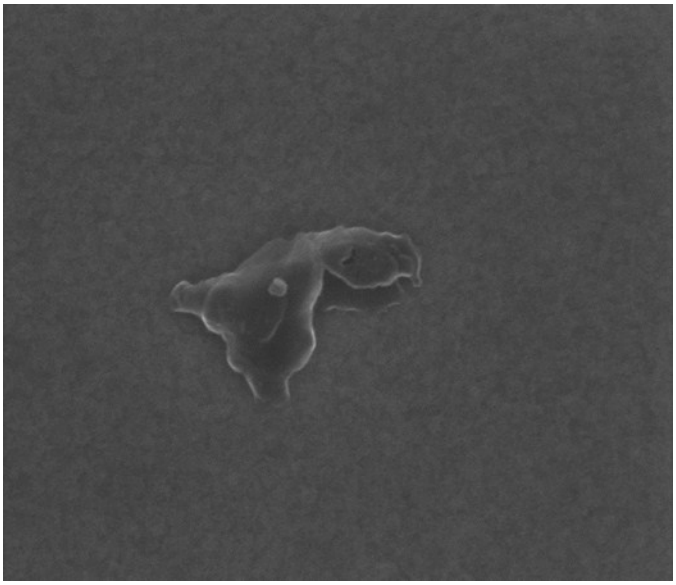


Figure 2-2: Plastic defect

The plastic defects are spread in random shapes over the surface of the silicon wafer. Pixel intensities of these defects on silicon wafer have very little difference from pixel intensities of the background, which makes it a difficult task to decide a definite threshold value or to detect the edges.

2.1.3 Pit defects

COPs: Crystal Originated Pits are a group of vacancies. These are formed during the polishing process or cleaning process of Czochralski-grown silicon wafers. Pits are the reason behind oxide defects. Pits cause gate oxide degradation or increase in leakage current [26][27]. In this thesis rounded pit defects are considered.

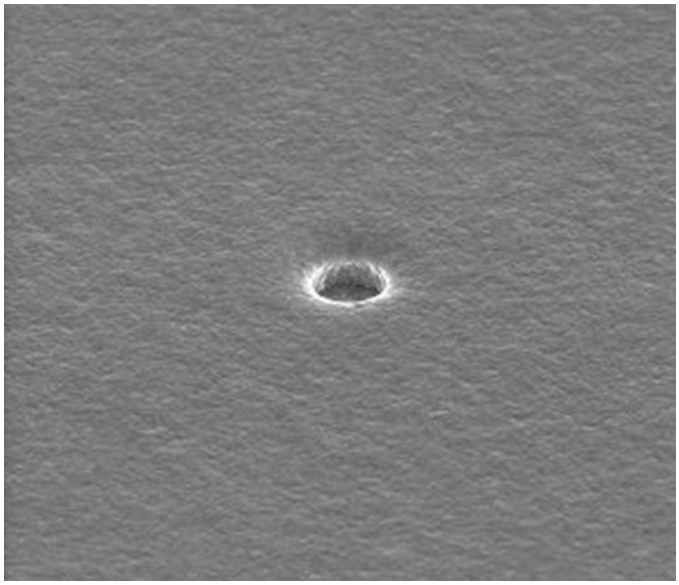


Figure 2-3: Pit defect

2.1.4 Protrusion defects

Protrusion defects are found protruding from the surface of the semiconductor. They deter fabrication processes, by preventing proper mask positioning which results in a loss of resolution. These defects are seen on the surface of the silicon wafers [28]. Also these defects can be formed due to expansion of a thin oxide film on the silicon surface [29]. In this thesis dome shaped or bubble shaped protrusions are considered.

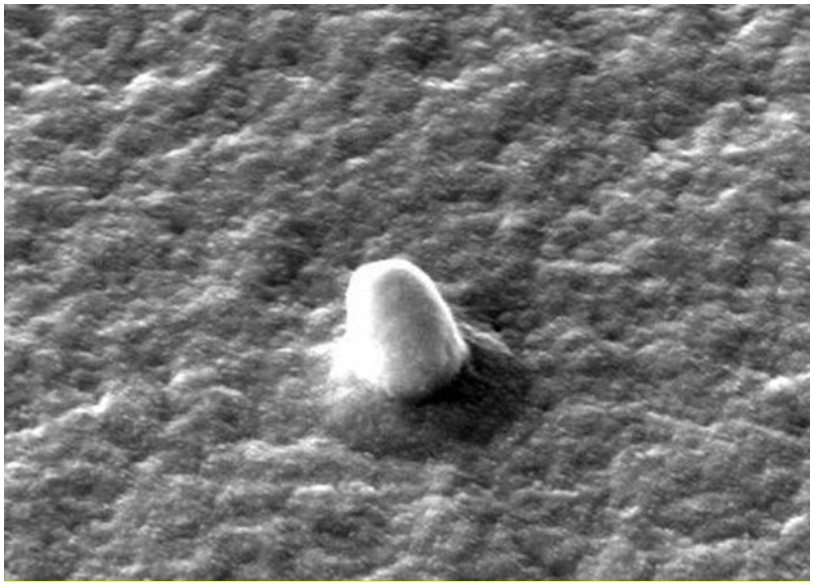


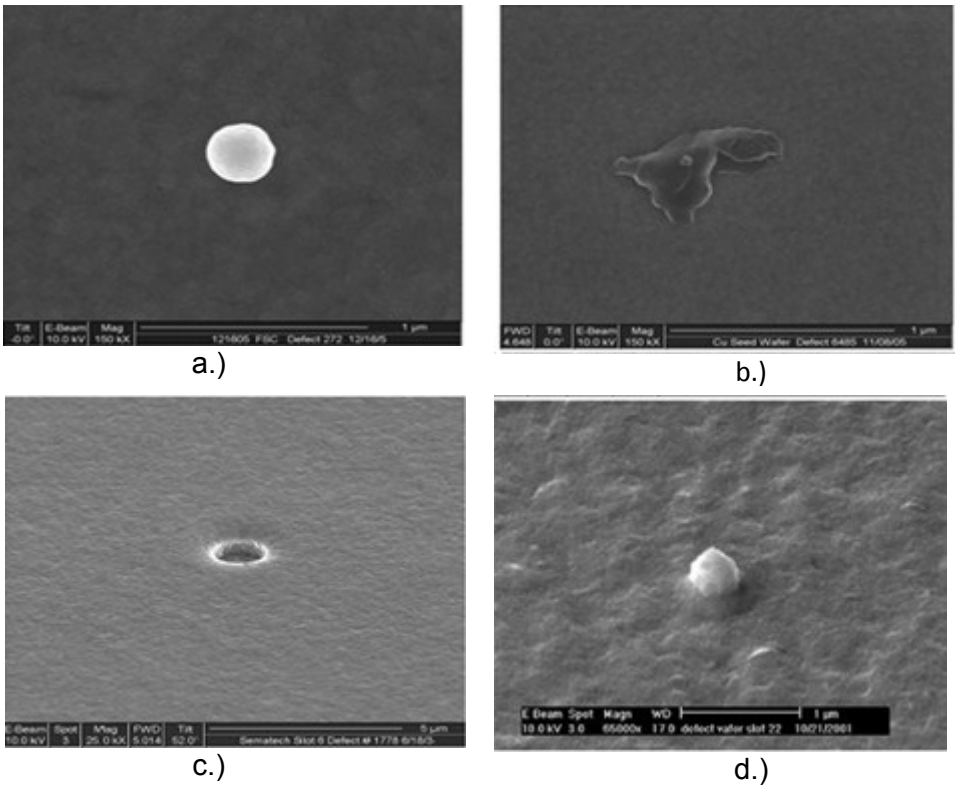
Figure 2-4: Protrusion defect

Chapter 3

Image Segmentation for Silicon Wafer Defects

In this chapter an image preprocessing method is developed, that helps towards better segmentation. Then the segmentation methods that are tried for segmenting defects in silicon wafer images are described. Also In the end, multiple algorithms for defect segmentation based on previously explained methods are presented, that ensure successful segmentation.

The operations that are performed for this thesis are performed on images whose gray levels vary on a scale from 0 to 255.



a.) b.) c.) d.)

Figure 3-1: Examples of four types of defects in silicon wafers a.) Copper ball defect b.) Plastic defect c.) Pit defect d.) Protrusion defect

3.1 Image preprocessing

From Figure 3-1 it can be seen that the background for the silicon wafer images vary in texture and are uneven in intensity. This causes problems while segmenting the defect. In order to reduce the influence of the background on image segmentation, the image negative is simply subtracted from the original, and the negative results are truncated to zero. The resulting image contains gray levels from 0 to 255. Equations (3-1) and (3-2) describe this image preprocessing step.

Consider the image x(m, n) of size M by N. For m varying from 1 to M and n varying from 1 to N the negative of image is given by equation 3-1

$$x'(m, n) = 255 - x(m, n) \tag{3-1}$$

The final steps are

$$y(m, n) = [x(m, n) - x'(m, n)] \cdot [u(x(m, n) - x'(m, n))] \tag{3-2}$$

Figure 3-2 and 3-3 respectively show the raw copper ball defect image and it's preprocessed image obtained using equation (3-2).

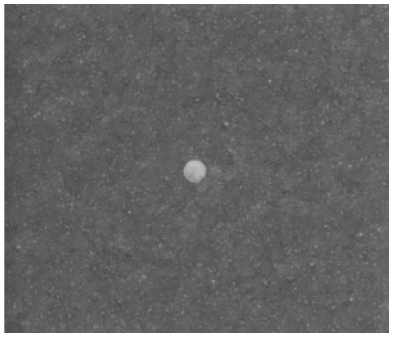


Figure 3-2: Original image

Figure 3-3: Preprocessed image

From Figure 3-3 it can be observed that the brightness of the background is significantly reduced and the defect is highlighted.

3.2 Block mean based segmentation

Figure 3-1 shows that in most of the cases defects or objects have higher luminance than the background. An intuitive method for segmenting these images would be threshold based segmentation [9], but as shown in Figure 3-4, some of the silicon wafer images have multimodal histograms. Thus deciding a threshold value [9] for efficient segmentation becomes a difficult task.

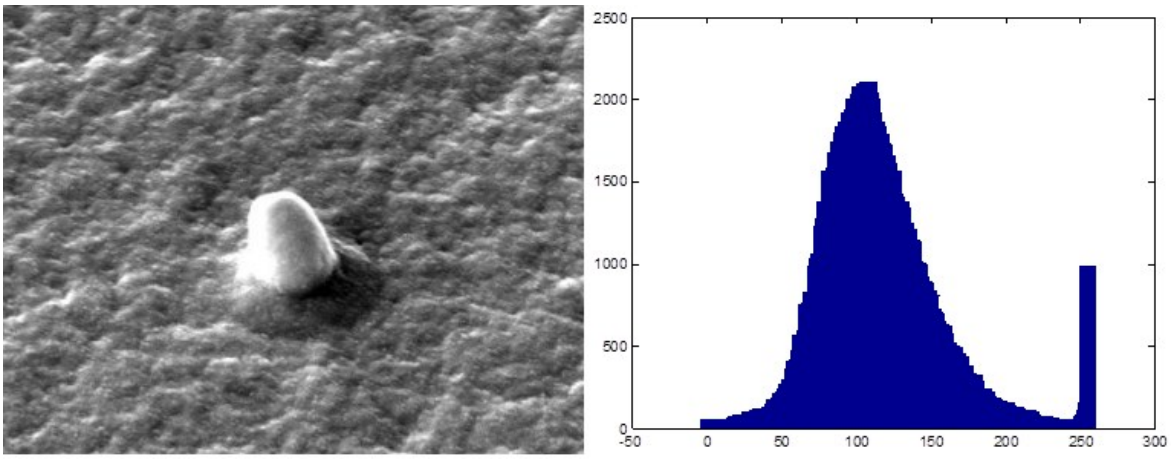


Figure 3-4: Protrusion defect and its histogram.

In this proposed segmentation method, segmentation is performed as follows:

1). Decompose the image into blocks

2). Compare mean pixel values for each block with the background mean value, and find the block that contains the defect or part of the defect.

3). Link together the blocks that are found to be containing defects or parts of defects to find the area where the defect lies.

The detailed algorithm for block mean based segmentation is given below:

Step 1: Decompose the preprocessed image into P·Q blocks, each of size U by V, where P is the number of blocks along rows and Q indicates the number of blocks along columns.

Step 2: For every block, calculate the mean using equation (3-3) and store in a matrix of size P by Q, such that each element of matrix represents mean of its corresponding block.

$$y(p,q) = \frac{1}{U \cdot V} \sum_{u=1}^{U} \sum_{v=1}^{V} x(u,v) \tag{3-3}$$

Step 3: Calculate the background mean value from the corner blocks, assuming they represent the background of the image.

Step 4: Calculate the error between the mean of each block and the background mean and store it in an error matrix E_m. Here each element of the matrix corresponds to a block in the image.

Step 5: Calculate the error threshold T_e using equation (3.4)

$$T_e = \frac{1}{P \cdot Q} \sum_{p=1}^{P} \sum_{q=1}^{Q} E_m(p,q) \tag{3-4}$$

Step 6: Convert the error matrix E_m to a logical error matrix E_{lm} using the error threshold T_e using equation (3-5).

$$\begin{aligned} E_{lm}(p,q)=1;\ E_m(p,q) \geq T_e \\ =0;\ E_m(p,q) \leq T_e \end{aligned} \tag{3-5}$$

Step 7: Remove the blocks that do not satisfy 4-connected neighborhood connectivity, in other words remove those blocks that have value 1 in matrix E_{lm}, but their immediate neighbors along horizontal and vertical directions do not have value 1.

Step 8: Connect all the blocks in the image that correspond to elements of the logical error matrix E_{lm} that have value 1, in order to get the area where defect lies. If all the elements in logical error matrix are zero, then the result has ‘no defects found’.

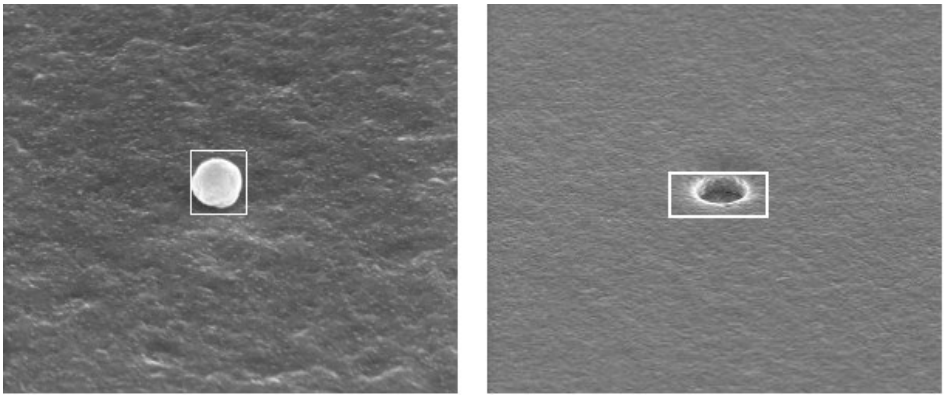


Figure 3-5: Segmentation result for block mean based segmentation.

Figure 3-5, 3-6, and 3-7 show the results of the block mean segmentation method. In Figure 3-5 the defects are successfully segmented. In Figure 3-6 the defect is not detected successfully. In Figure 3-7 the defect could not be found. The block mean based segmentation method fails to segments plastic and protrusion defects. Thus another method is proposed that is explained in the next section.

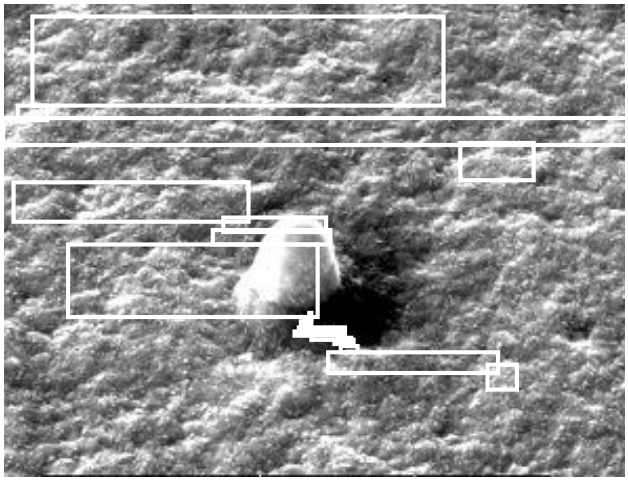


Figure 3-6: Result of block mean based segmentation, showing over segmentation

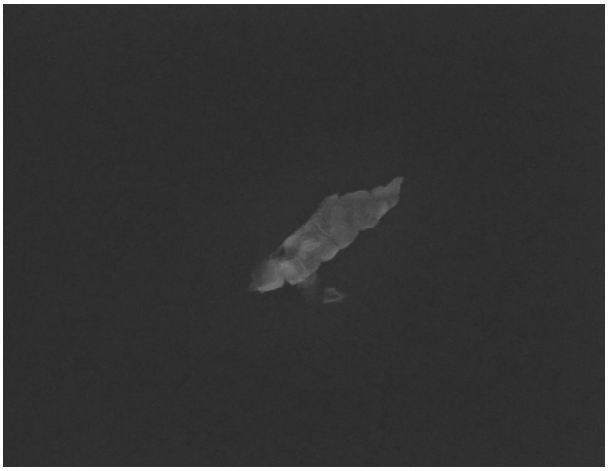


Figure 3-7: Result of block mean segmentation, with failed segmentation

3.3 Edge detection based segmentation

This method takes into consideration the presence of a distinctive boundary between the defect and the background. It uses more than one edge detection filter. The Sobel operator [9], the Prewitt operator [9] and the finite difference operator are used to detect horizontal, vertical, and diagonal edges. When one operator fails to find a region in the image, that contains the defect, another operator is used. The operators are used in order: finite difference, Prewitt and Sobel. When the image is too noisy, it is smoothed by a low pass mean filter.

In a 3 X 3 region of image y(m,n) where the y's are gray-level values as shown in Figure 3-8. Finite difference, Prewitt and Sobel operators are applied to calculate the gradient at point labeled y_5 , and gradients for detecting horizontal, vertical and diagonal edges for each of the above mentioned operators are given in Table 3-1.

y_1	y_2	y_3
y_4	y_5	y_6
y_7	y_8	y_9

Figure 3-8: 3 ×3 region of image

Table 3-1: Table of equations for gradients of finite difference, Sobel, Prewitt operator

	Gradient for :		
Operator	Horizontal edges	Vertical edges	Diagonal edges
Finite difference operator	$(y_8 - y_2)$	$(y_6 - y_4)$	$(y_9 - y_1)$ and $(y_1 - y_9)$
Prewitt operator	$(y_7+y_8+y_9)-(y_1+y_2+y_3)$	$(y_3+y_6+y_9)-(y_1+y_4+y_7)$	$(y_2+y_3+y_6)-(y_4+y_7+y_8)$ and $(z_6+y_8+y_9)-(y_1+y_2+y_4)$
Sobel operator	$(y_7+2y_8+y_9)-(y_1+2y_2+y_3)$	$(y_3+2y_6+y_9)-(y_1+2y_4+y_7)$	$(y_2+2y_3+y_6)-(y_4+2y_7+y_8)$ and $(y_6+y_8+2y_9)-(2y_1+y_2+y_4)$

The algorithm for edge detection based segmentation is based on the following steps.

Step 1: Pass the image through filters, starting with the finite difference filter.

Step 2: After filtering a region which is predicted to be the defect is found. If there is more than one region found then this noisy image is passed through low pass mean filter for image smoothing.

Step 3: If in step 2 no regions are found then the filter is replaced by next filter and step 2 is repeated under suitable conditions. The filters follow the order- finite difference, Prewitt, Sobel

Step 4: If all the filters fail to find the region then the result is “No defect found”.

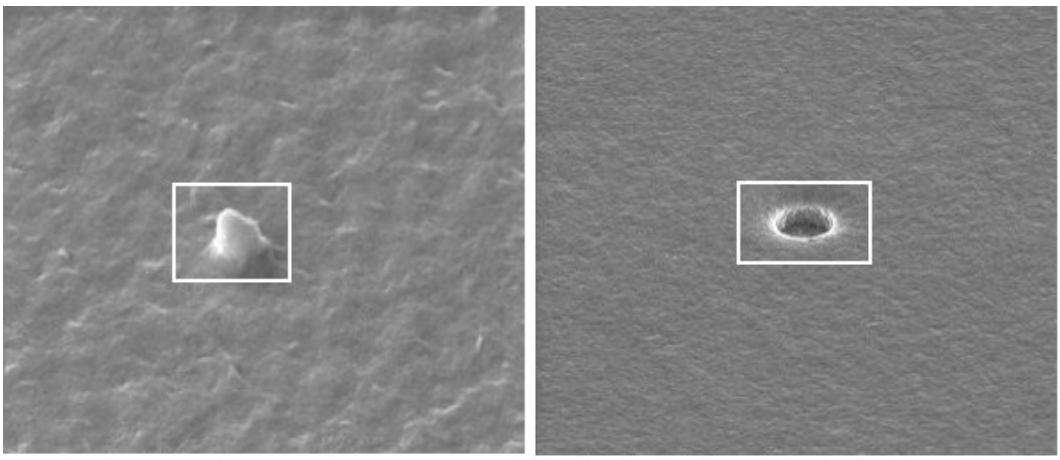


Figure 3-9: Result of edge based segmentation

Figure 3-9 shows results of edge detection based segmentation, where the algorithm successfully detected the area covered by the defect. From Figure 3-10 it can be clearly seen that this segmentation method failed since instead of detecting the entire object only partial object was detected in each case.

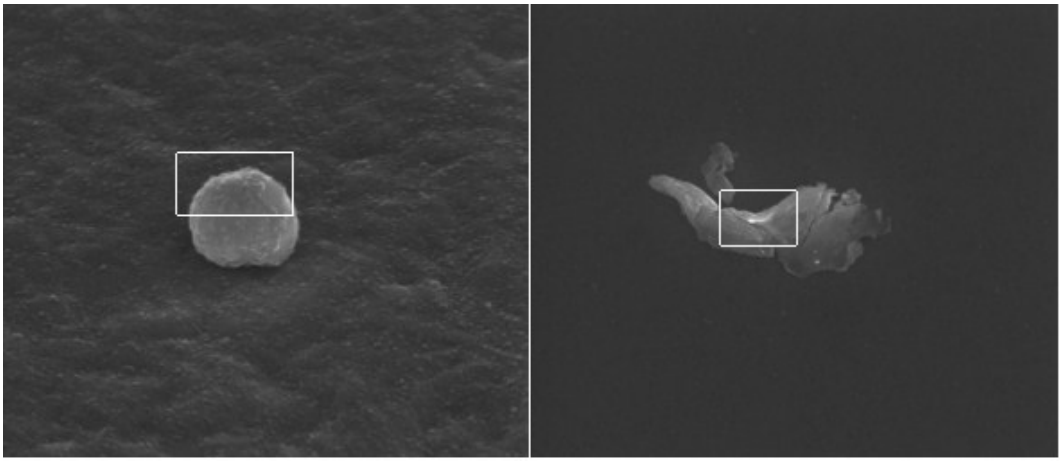


Figure 3-10: Result of edge based segmentation showing partial segmentation

3.4 Multiple algorithms based segmentation.

Even though the block mean based segmentation method and edge detection based segmentation method are capable of detecting defects in silicon wafer images, they are not efficient and reliable. Thus a method that consists of multiple algorithms is developed. The main issue, while selecting a segmentation method for silicon wafer defects, is that each defect differs from the background. In the proposed method four different segmentation techniques; that are based on block mean based segmentation and edge detection based segmentation are combined together. Each segmentation algorithm is designed such that it possesses highest efficiency and accuracy in segmenting a particular type of defect.

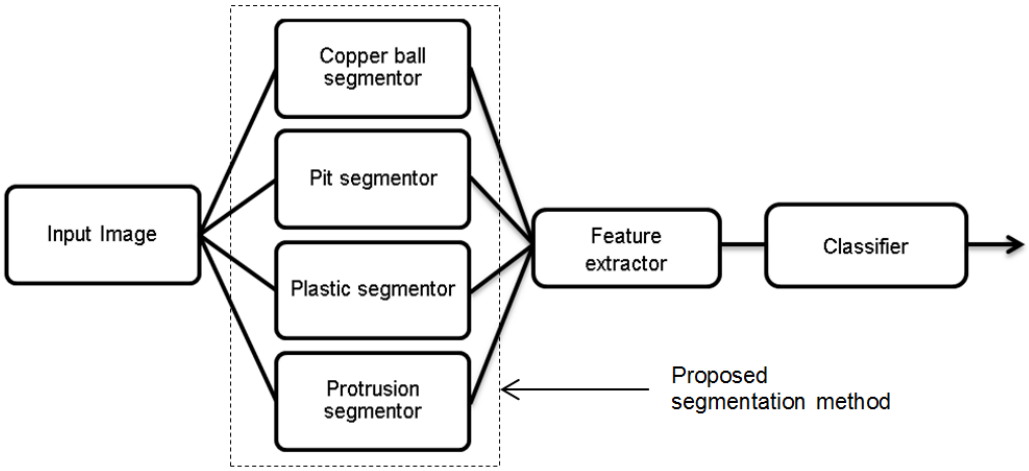


Figure 3-11: Proposed segmentation method within recognition system

For copper ball defects, the block size in the block mean based segmentation is uses P= 36, Q= 40.ase the values selected for the number of blocks. For these block values, the block mean based algorithm is followed.

For pit defects; P= 36, Q= 40. The criterion for selecting T_e depends on the median value of mean of every column in error matrix E_m.

Figure 3-12 and 3-13 show the segmentation of copper ball and pit defects by their respective segmentors.

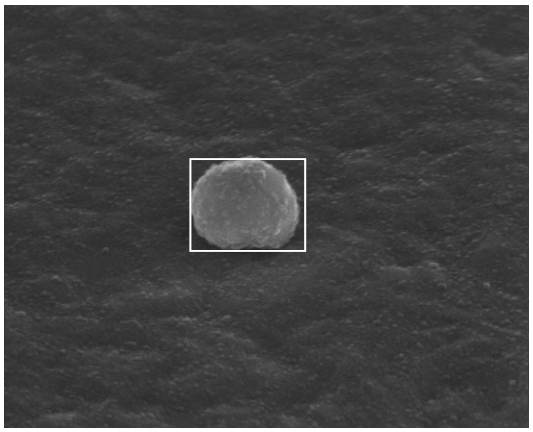


Figure 3-12: Result of the copper ball segmentor on copper ball defect

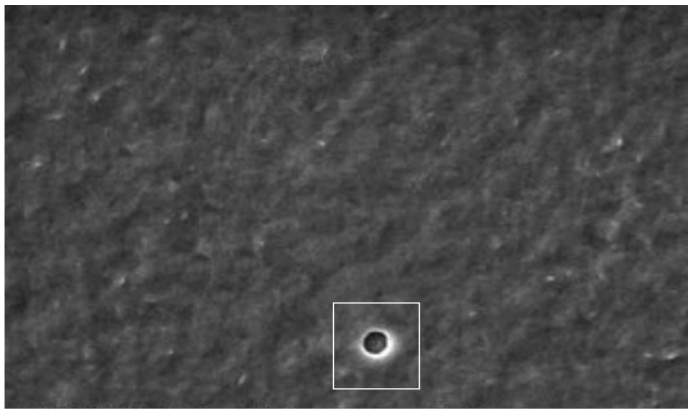


Figure 3-13: Result of the pit segmentor for pit defects.

For plastic defects, the problem is that they lack distinguishable edges or do not show much difference in luminance between defect and background. By making changes in the preprocessing method, the error threshold and block size for block mean based segmentation method, a highly reliable segmentor for silicon plastic defect is developed. The steps employed for this method are shown below.

Step 1: Decompose the preprocessed image into P.Q blocks P= 54, Q= 60. After trying different combinations of P and Q these values were found most effective.

Step 2: For every block, calculate the mean using equation (3-3) and store in a matrix of size P by Q, such that each element of the matrix represents a mean of its corresponding block.

Step 3: Calculate the background mean value from four blocks from each corner

Step 4: Calculate the error between the mean of each block and the background mean and store it in an error matrix E_m. Here each element of the matrix corresponds to a block in the image.

Step 5: Calculate the error threshold T_e using equation (3-6)

$$T_e = \min(E_m) \tag{3-6}$$

Step 6: Convert the error matrix E_m to a logical error matrix E_{lm} using error threshold T_e, using equation (3.5).

Step 7: Remove the blocks that do not satisfy 4-connected neighborhood connectivity.

Step 8: Connect all the blocks in image that correspond to elements of logical matrix that have value 1, in order to get the area where a defect lies. If all the elements in logical error matrix are zero, then the result has no defects.

Figure 3-14 shows the result of the plastic segmentor for plastic defects.

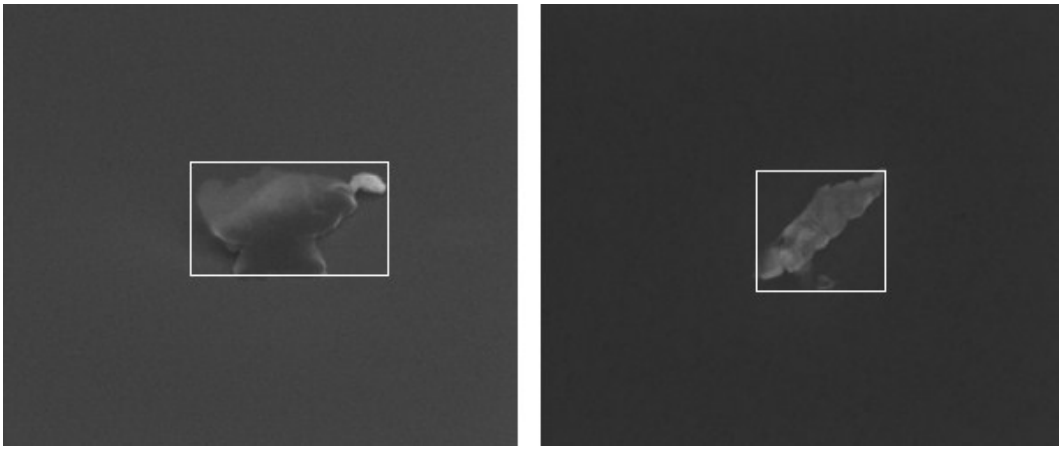


Figure 3-14: Result of the plastic segmentor on plastic defects

For protrusion defects, the wafer images have a rough background that affects defect detection by block mean based segmentation. This leads to over segmentation thereby failing to segment the protrusion defects. The edge based segmentation algorithm was tried for protrusion defects found to work effectively for protrusion defects. Figure 3-15 shows the segmentation of protrusion defects using the protrusion segmentation method.

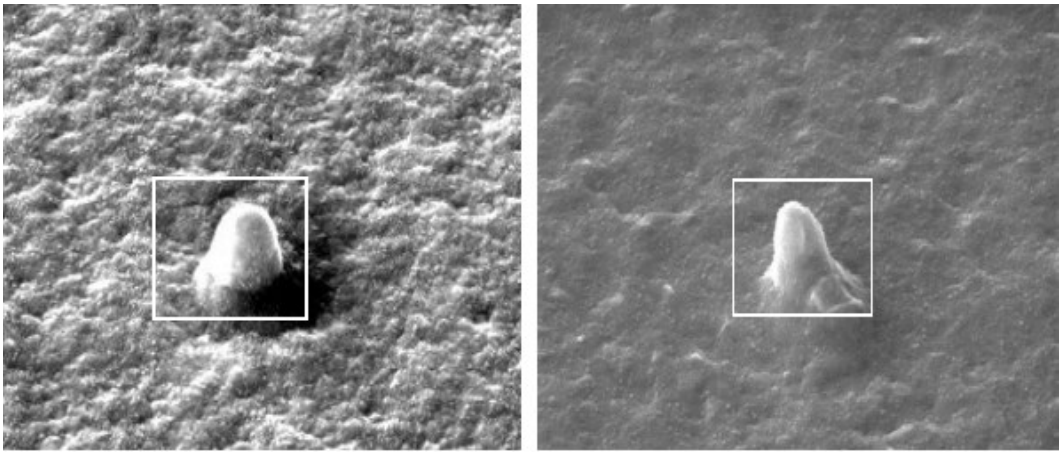


Figure 3-15: Result of the protrusion segmentor on protrusion defects

The block diagram given in Figure 3-11 shows how the proposed segmentors are used .Results of these segmentors are the isolated defects. These results are fed into a classifier through a feature extractor, and by applying majority rule to the outputs of the classifier; the correct class is determined.

Chapter 4

Feature Extraction and Classification

The segmentation algorithms presented in Chapter 3 locate and isolate the silicon wafer image defects. The resultant segmentation images are used to determine the type of defect or to assign the segmented defects to a certain class on the basis of their attributes. This operation is called classification.

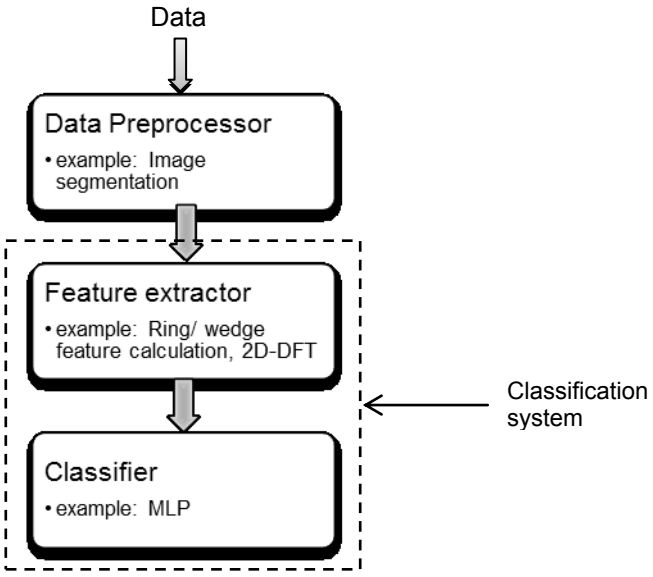


Figure 4-1: Pattern recognition system

In this chapter, the classification system shown in Figure 4-1 is described. The first part of this chapter discusses techniques for feature extraction such as the Fourier transform. In the next part, feed forward neural network based MLP classifiers are explained in detail.

4.1 Feature extraction

The feature extractor which precedes the classifier in Figure 4-1, calculates vectors of numbers called features that can effectively represent the object to be

recognized. These feature vectors are similar to each other within a same class but are different for different classes. An efficient feature extraction method must be able to generate distinctive features that are invariant to distortion and can effectively maintain class separability. Feature sets are not designed to minimize classification error [30].

The feature extraction method used in this thesis uses a transformation approach [30]. Entire image is transformed into a vector or series, and significant compression of data is achieved. The 2-D discrete Fourier transform (DFT) of the image is calculated and mapped to a one dimensional array in order to obtain 2-D DFT features.

4.1.1 2D DFT features

While applying Fourier transform to images, 2-D Fourier transform calculations is necessary. The pixels indexed by two dimensional spatial coordinates correspond to horizontal and vertical frequencies in frequency domain [11]

For given N by N image f(x, y), which represents the output of image segmentation, 2D DFT F(u, v) is given by

$$F(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) e^{-j\left(\frac{2\pi}{N}\right)(ux+vy)} \tag{4-1}$$

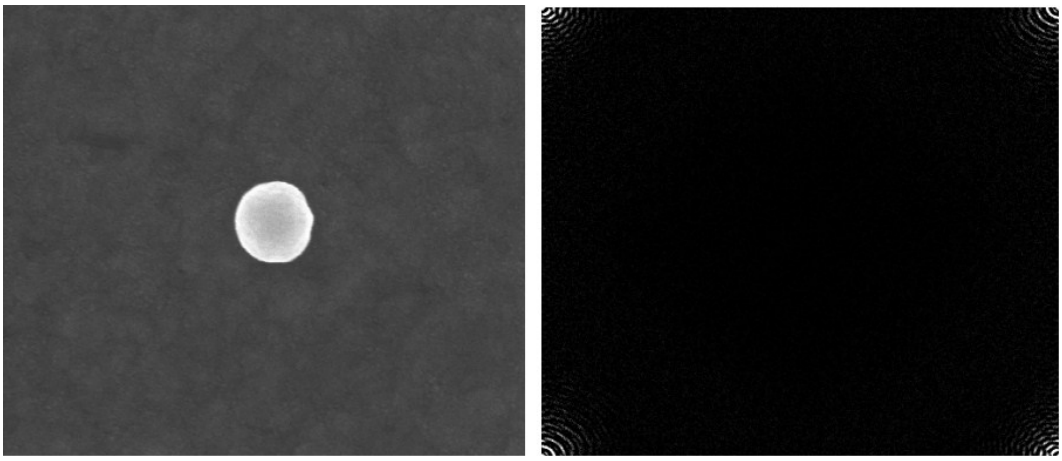
The real and imaginary parts of F (u, v) are respectively R(u, v) and I(u, v).

$$R(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left(\left(\frac{2\pi}{N}\right)(ux+vy)\right) \tag{4-2}$$

$$I(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\sin\left(\left(\frac{2\pi}{N}\right)(ux+vy)\right) \tag{4-3}$$

$$\mathrm{g} = \begin{pmatrix} R \\ I \end{pmatrix} \tag{4-4}$$

When a 2-D DFT is calculated for an image, the corners of the 2-D DFT contain low frequency components while high frequency components are near the center. Figure 4-2 shows, image of silicon wafer and its 2-D DFT.



a.) b.)

Figure 4-2: a.)Silicon wafer image and b.) its' 2 D DFT

From Figure 4-2; 2D DFT of segmented copper ball defect, it can be observed that most of the information in the image is low frequency information.

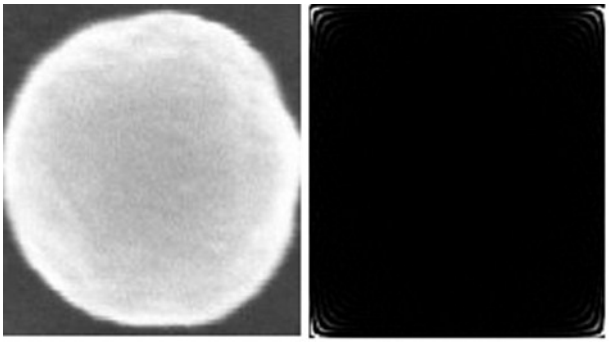


Figure 4-3: Segmented copper ball defect and its 2D DFT

For this thesis, low frequency, nonredundant features are used for a positive integer M, for u and v ($-M \le u \le M$, $0 < v \le M$) and ($1 \le u \le M$ for $v=0$).The features calculated by

equations (4-2) and (4-3) will be mapped onto a one dimensional array; as shown in equation (4.4) in order to get feature vector **g**

4.2 Classifier

The features calculated by feature extractor are fed into the MLP discriminant, since it does not need a mathematical model for classification [36].

Neural network classifiers approximate the optimal Bayesian classifier [37] and are trained using the mean squared error (MSE) objective function. Although the expectation value of the classifier error rate is considered to be the ideal objective function, training algorithms that are based on the minimization of the expected squared error criteria are often easier to mechanize and better understood. The mean square error training criterion is one of the reasons for the good performance of MLP over other classifiers.

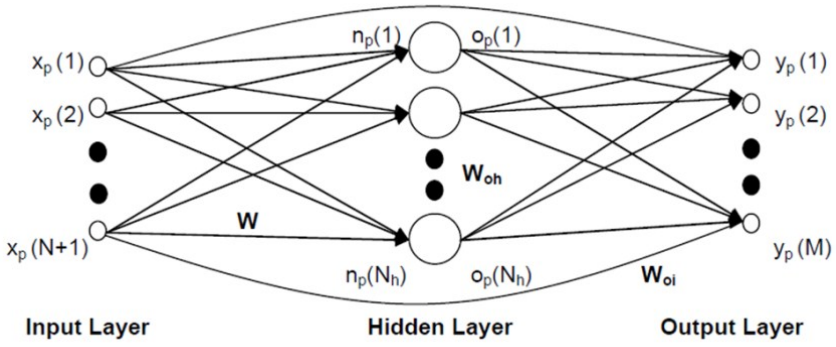


Figure 4-4: Fully connected MLP network

Figure 4-4 shows a fully connected MLP network. It consists of an input layer, one or more hidden layers and an output layer. All the layers are connected by weights and the signal travels from the input through the hidden layers, to the output layer.

Input weights w(k, n) connect the n^{th} input $x_p(n)$ to the k^{th} hidden unit with net function $n_p(k)$. Output weights $w_{oh}(m, k)$ connect the kth hidden unit having activation

$o_p(k)$ to the mth output $y_p(m)$, which has a linear activation. The bypass weight $w_{oi}(m,n)$ connects the nth input to the mth output. The training data, described by the set $(\mathbf{x_p},\mathbf{t_p})$ consists of N-dimensional input vectors $\mathbf{x_p}$ and M-dimensional desired output vectors, $\mathbf{t_p}$. The pattern number p varies from 1 to N_v where N_v denotes the number of training vectors present in the data set.

The input vectors are augmented by an extra element $x_p(N+1)$ where, $x_p(N+1) = 1$, so $\mathbf{x}_p = [x_p(1), x_p(2),\ldots., x_p(N+1)]^T$, in order to handle the thresholds in the hidden and output layers. Let N_h denote the number of hidden units. The dimensions of the weight matrices $\mathbf{W}$, $\mathbf{W_{oh}}$ and $\mathbf{W_{oi}}$ are respectively N_h by (N+1), M by N_h and M by (N+1). The k^{th} element of net vector $\mathbf{n}_p$ can be written as,

$$n_p(k) = \sum_{n=1}^{N+1} w(k,n)x_p(n) \tag{4-5}$$

In matrix notation,

$$\mathbf{n_p} = \mathbf{W} \cdot \mathbf{x_p} \tag{4-6}$$

where the k^{th} element of the hidden unit activation vector $\mathbf{o_p}$ is calculated as $o_p(k) = f(n_p(k))$ and f(.) denotes the hidden layer activation function. For the sigmoid activation case, the output of the k^{th} hidden unit is given by

$$f(n_p(k)) = \frac{1}{1+e^{-n_p(k)}} \tag{4-7}$$

The actual output of the network, $\mathbf{y_p}$ can be written as,

$$y_p(m) = \sum_{n=1}^{N+1} w_{oi}(m,n)x_p(n) + \sum_{k=1}^{N_h} w_{oh}(m,k)o_p(k) \tag{4-8}$$

In matrix notation,

$$\mathbf{y_p} = \mathbf{W_{oi}} \cdot \mathbf{x_p} + \mathbf{W_{oh}} \cdot \mathbf{o_p} \tag{4-9}$$

The outputs of MLP are computed as a weighted sum of the inputs and the hidden unit outputs. The weights form the unknowns, which are typically found by minimizing the mean squared error between the actual and desired outputs,

$$E = \sum_{i=1}^{M} E(i) = \frac{1}{N_v} \sum_{i=1}^{M} \sum_{p=1}^{N_v} \left[t_p(i) - y_p(i) \right]^2 \tag{4-10}$$

The MLP is designed by minimizing the standard training error as in equation (4.10), where M is the number of classes, Nv denotes the total number of training patterns, E(i) is mean-squared error for the i^{th} class. $t_p(i)$ denotes the ith desired output and $y_p(i)$ denotes the i^{th} observed output. The i^{th} desired output for the p^{th} pattern is defined as

$$t_p(i) = \begin{cases} 1 & \text{for } i = i_c \\ 0 & \text{for } i = i_d \end{cases} \tag{4-11}$$

where i_c denotes the correct class number for the current training pattern and i_d denotes any incorrect class number for that pattern.

Output activations $y_p(i_c)$ are used as class discriminants. The classifier is said to have correctly classified the p^{th} pattern when $y_p(i_c)$ is the largest observed output, otherwise we indicate that the network has misrecognized the p^{th} pattern by incrementing the classification error count. In either case, squared residuals accumulate in the standard training error as we step through the set of N_v training patterns. This network can be trained by output weight optimization – backpropagation algorithm [33] (OWO-BP) and hidden weight optimization [40] with multiple optimal learning factors [41] (HWO-MOLF)

Output weight optimization (OWO) [38] is a technique to solve for weights connected to the actual outputs of the network (this would be the output weights, $\mathbf{W}_{oh}$ and

by-pass weights, $\mathbf{W}_{oi}$). Since the outputs have linear activation, finding the weights connected to the outputs is equivalent to solving a system of linear equations. The back propagation (BP) algorithm [39] is a first order method that uses gradient information to update the weights in the network. In full batch mode, the (BP) algorithm updates the input weights and thresholds.

In HWO [40] , the hidden weights are updated by minimizing separate error functions for each hidden unit. The error functions measure the difference between the desired and the actual net function.

A learning algorithm called the multiple optimal learning factor (MOLF) algorithms [41] calculates an optimal learning factor for every hidden unit. In MOLF Newton's method is used to calculate a separate optimal learning factor for each hidden unit's input weights in a feed forward network. The Newton's method can be viewed as a second order method to assign a learning rate to every weight in the network.

4.2.1 Classification error

Algorithm for calculating the probability of error and percentage classification error is as follows

(1) Initialize $N_{err} = 0$, $P_e = 0$ and Percentage Classification Error = 0

(2) For $1 \leq p \leq N_v$, read the input vector ($\mathbf{x_p}$) and the correct class (i_c)

(3) Find the output vector ($\mathbf{y_p}$) using the input vector ($\mathbf{x_p}$) and the output weight matrix ($\mathbf{W_o}$)

(4) Find the index i for which $y_p(i)$ is maximum, that index i is the estimated class

(5) If estimated class = i_c; correct class then go to step(7)

(6) Increment N_{err}

(7) If the entire data file is read then go to step(8) else go to step(2)

(8) Classification error, $P_e=N_{err}/N_v$,

Percentage classification error= $(100 \times N_{err})/ N_v$

4.2.2 Output reset method

An MLP based classifier trained using an MSE objective function, has a standard squared error that accumulates squared residuals for each pattern. Each instance of residual error contains an additive bias inherent to each output vector and error component due to an individual output value having the correct sign but a larger magnitude than that of the desired output. The output reset method [32] is an approach for removing these biases. In OR training method, the error, E', more closely models the classification error. The OR algorithm, combined with enhanced MOLF-HWO training, greatly improves classifier performance. Details for the OR algorithm are described in [32]

Chapter 5

Results

In this chapter, results for the algorithms shown in Chapter 3 and 4 are presented. The algorithms are implemented in Matlab. 3-fold validation is used to generate the results.

In order to perform 3-fold validation, the raw images are divided into three disjoint groups such that no two groups contain the same images. For every group a training data set, validation data set, and a test data set is created. For each group the training data set and the validation data sets contain segmented images obtained from the algorithm proposed in Chapter 3, for corresponding raw images data.

In 3- fold validation- training is performed on training data from one group, and validation data from second group is used for pruning [43] of the network. The best sized trained network is used on the remaining group for testing. Testing is performed by passing the test data through multiple algorithm segmentation system and the feature vectors for each of the segmented images are passed on to the network trained in previous step. The accuracy of the results of classifier determines the efficiency of the proposed system.

The following part discusses the results obtained in 3 fold validation and steps taken to improve these results. For each case discussed in following sections there are three subcases A, B and C. These cases are defined by the group used for training, validation and testing.

For the available set of raw silicon wafer images, the images are distributed such that each of the three groups contains roughly the same number of images for each type of defect. Following Table 5-1 describes the number of images for each type of defect in every group

Table 5-1: Table describing the number of images in each group for 3-fold validation

	Group 1	Group 2	Group 3
Copper ball	10	9	9
Pit	3	3	3
Plastic	7	8	7
Protrusion	3	4	4
Total	23	24	23

5.1.1 Case I

For this case, all the features from the feature extractor; explained in chapter 4 are used for training, validation and testing. The total number of features in a feature vector is 24.

Table 5-2: Describing groups used and accuracy percentage for each case for case I

	Group number		
Case Number	I-A	I-B	I-C
Training	1	1	2
Validation	2	3	3
Testing	3	2	1
% of accurately recognized detects for testing data set	69.5652	62.5000	47.8261

Since the accuracy does not represent successful classification for each type of defect, it can be misleading. Also the number of images in the test group for each type of defect is not equal in a group. Thus a confusion matrix is presented. Table 5-3, Table 5-4 and Table 5-5 show confusion matrices for cases I-A, I-B and I-C respectively

Table 5-3 Confusion matrix for case I-A

Actual class	Predicted Class				
	Case I-A	Copper ball	Pit	Plastic	Protrusion
	Copper ball	7	2	0	0
	Pit	0	2	1	0
	Plastic	0	0	5	2
	Protrusion	0	1	1	2

Table 5-4: Confusion matrix for case I-B

Actual class	Predicted Class				
	Case I-B	Copper ball	Pit	Plastic	Protrusion
	Copper ball	8	0	1	0
	Pit	0	3	0	0
	Plastic	4	1	3	0
	Protrusion	1	2	0	1

Table 5-5: Confusion matrix for case I-C

Actual class	Predicted Class				
	Case I-C	Copper ball	Pit	Plastic	Protrusion
	Copper ball	8	0	0	2
	Pit	0	2	1	0
	Plastic	4	1	1	1
	Protrusion	2	0	1	0

From Table 5-3, 5-4 and 5-5, it can be seen that the accuracy for recognition of individual classes is poor as compared to the overall recognition percentage for all the images.

5.1.2 Case II

In order to improve the results, feature selection [42] is tried. In feature selection for each chosen subset, an MLP classifier was designed. The final classifier is that with the minimum validation error.

Table 5-6: Describing groups used and accuracy percentage for each case for case II

	Group number		
Case number	II-A	II-B	II-C
Training	1	1	2
Validation	2	3	3
Testing	3	2	1
% of accurately recognized detects for testing data set	78.2609	70.8333	69.5652

From Table 5-6 and 5-2, it can be seen that using feature selection has significantly improved the percentage of accurately classified defects.Table 5-7Table 5-8Table 5-9 show confusion matrices for cases II-A, II-B and II-C respectively

Table 5-7 Confusion matrix for case II-A

	Predicted Class				
Actual class	Case II-A	Copper ball	Pit	Plastic	Protrusion
	Copper ball	9	0	0	0
	Pit	1	1	1	0
	Plastic	0	0	7	0
	Protrusion	0	0	3	1

Table 5-8: Confusion matrix for case II-B

Actual class	Predicted Class				
	Case II-B	Copper ball	Pit	Plastic	Protrusion
	Copper ball	8	1	0	0
	Pit	1	2	0	0
	Plastic	2	1	4	1
	Protrusion	1	0	0	3

Table 5-9: Confusion matrix for case II-C

Actual class	Predicted Class				
	Case II-C	Copper ball	Pit	Plastic	Protrusion
	Copper ball	10	0	0	0
	Pit	1	1	1	0
	Plastic	1	1	4	1
	Protrusion	1	1	0	1

From Table 5-6, 5-7, 5-8, and 5-9 it can be observed that the overall performance has improved for case II as compared to case I. However, the performance for individual type of defects has been considerably affected. For example, in case II-A all the test images for copper ball and plastic defects are correctly recognized whereas in case I-A, 2 images out of 9 for copper ball and 2 images out of 7 for plastic were incorrectly recognized. On the other hand, 1 image out of 3 for pit and 1 image out of 4 protrusion images are correctly classified for case I-A while, 2 images out of 3 for pit and 2 images out of 4 protrusion images are correctly classified for case II-A. Thus, in case of the defect types where samples in testing data are less in number are adversely affected by feature selection.

5.1.3 Case III

In this case, the training data to be used for training is increased by using training group and validation group together. The network is first pruned [44] using training data and validation data. Best network size determined during pruning is used to train a network for data in training group and validation group together. This newly trained network is used for testing. The results are as follows:

Table 5-10: Describing groups used and accuracy percentage for each case for case III

	Group number		
Case number	III-A	III-B	III-C
Training	1	1	2
Validation	2	3	3
Testing	3	2	1
% of accurately recognized detects for testing data set	78.2609	75.0000	65.2174

Tables 5-11, 5-12, 5-13 show confusion matrices for cases III-A, III-B and III-C respectively.

Table 5-11 Confusion matrix for case III-A

	Predicted Class				
	Case III-A	Copper ball	Pit	Plastic	Protrusion
Actual class	Copper ball	9	0	0	0
	Pit	0	1	2	0
	Plastic	1	0	6	0
	Protrusion	0	1	1	2

Table 5-12: Confusion matrix for case III-B

Actual class	Predicted Class				
	Case III-B	Copper ball	Pit	Plastic	Protrusion
	Copper ball	9	0	0	0
	Pit	0	3	0	0
	Plastic	3	1	4	0
	Protrusion	2	0	0	2

Table 5-13: Confusion matrix for case III-C

Actual class	Predicted Class				
	Case III-C	Copper ball	Pit	Plastic	Protrusion
	Copper ball	10	0	0	0
	Pit	0	3	0	0
	Plastic	5	1	1	0
	Protrusion	2	0	0	1

From Table 5-10, 5-11, 5-12 and 5-13 even though results for case III are improved or have remained equally good in case of test group 3 and 2, in case of test group 1 there is scope of improvement as plastic defect and protrusion defect are poorly classified.

5.1.4 Case IV

In this case instead of using entire feature matrix, only the features having nonzero values are used for training. The training is performed using the same method used in case III. The results are shown in

Table 5-14, 5-15, 5-16 and 5-17 show confusion matrices for cases III-A, III-B and III-C respectively.

Table 5-14: Describing groups used and accuracy percentage for each case for case IV

	Groups number		
Case number	IV-A	IV-B	IV-C
Training	1	1	2
Validation	2	3	3
Testing	3	2	1
% of accurately recognized detects for testing data set	78.2609	75.0000	78.2609

Table 5-15, 5-16 and 5-17 show confusion matrices for cases IV-A, IV-B and IV-C respectively

Table 5-15 Confusion matrix for case IV-A

	Predicted Class				
Actual class	Case IV-A	Copper ball	Pit	Plastic	Protrusion
	Copper ball	9	0	0	0
	Pit	0	2	1	0
	Plastic	1	0	6	0
	Protrusion	1	1	1	1

Table 5-16: Confusion matrix for case IV-B

	Predicted Class				
Actual class	Case IV-B	Copper ball	Pit	Plastic	Protrusion
	Copper ball	8	1	0	0
	Pit	0	3	0	0
	Plastic	1	1	5	0
	Protrusion	2	0	0	2

Table 5-17: Confusion matrix for case IV-C

Actual class	Predicted Class				
	Case IV-C	Copper ball	Pit	Plastic	Protrusion
	Copper ball	10	0	0	0
	Pit	0	3	0	0
	Plastic	3	0	4	0
	Protrusion	2	0	0	1

5.2 Conclusions

Based on the 3- fold validation test cases discussed above, the following observations are made:

- No single segmentation scheme worked for all 4 defects.
- DFT features work acceptably.
- Feature selection did not help all classes.
- After network sizing, it is best to use all training and validation data to train a classifier.
- More data is needed.

References

[1]. Texas Instruments Incorporated. TI introduces first commercial silicon transistor Available: http://www.ti.com/corp/docs/company/history/timeline/semicon/1950/docs/54commercial.htm

[2]. F. T. Shah and K. Yousaf. Handwritten digit recognition using image processing and neural networks. World Congress on Engineering 2007 (Volume 2) 2pp. 648-651. 2007.

[3]. K. Maglad, D. Mohamad and N. A. Abulgasem. Saudian car license plate number detection and recognition using morphological operation and RBF neural network. Australian Journal of Basic & Applied Sciences 5(12), pp. 1780-1786. 2011.

[4]. N. Ratha and R. Bolle. Automatic Fingerprint Recognition Systems 2003.

[5]. Qi-Chuan Tian, Quan Pan, Yong-Mei Cheng and Quan-Xue Gao. Fast algorithm and application of hough transform in iris segmentation. Presented at Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on, 2004, 7, 3977-3980 vol.7

[6]. D. Gaceb, V. Eglin, F. LeBourgeois and H. Emptoz. Physical layout segmentation of mail application dedicated to automatic postal sorting system. Presented at Document Analysis Systems, 2008. DAS '08. the Eighth IAPR International Workshop on. 2008, 408- 414.

[7]. M. Sabourin and A. Mitiche. Optical character recognition by a neural network. Neural Networks 5(5), pp. 843-852. 1992.

[8]. C. F. Olson and D. P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. Image Processing, IEEE Transactions on 6(1), pp. 103-113. 1997.

[9]. R. C. Gonzalez and R. E. Woods. Digital Image Processing 2002

[10]. M. Petrou and C. Petrou. Image Processing: The Fundamentals 2010.

[11]. M. S. Nixon and A. S. Aguado. Feature extraction and image processing. pp. xv, 405 p. 2008.

[12]. E. L. Hines and P. A. Watson. Application of edge detection techniques to detection of the bright band in radar data. Image Vision Comput. 1(4), pp. 221-226. 1983.

[13]. Q. Munib, M. Habeeb, B. Takruri and H. A. Al-Malik. American sign language (ASL) recognition based on hough transform and neural networks. Expert Syst. Appl. 32(1), pp. 24-37. 2007.

[14]. M. Sonka, V. Hlavac and R. Boyle. Image processing, analysis, and machine vision. pp. xxiv, 770 p. 1999.

[15]. F. Z. Kettaf, D. Bi and J. P. Asselin de Beauville. A comparison study of image segmentation by clustering techniques. Presented at Signal Processing, 1996., 3rd International Conference on. 1996, 1280-1283.

[16]. N. R. Pal and S. K. Pal. A review on image segmentation techniques. Pattern Recognition 26(9), pp. 1277-1294. 1993.

[17]. H. Trichili, M. -. Bouhlel, N. Derbel and L. Kamoun. A survey and evaluation of edge detection operators application to medical images. Presented at Systems, Man and Cybernetics, 2002 IEEE International Conference on. 2002

[18]. K. Fukunaga. Introduction to statistical pattern recognition. pp. xiii, 591 p. 1990.

[19]. O. Ersoy and D. Kim. Image recognition with the discrete rectangular-wave transform. JOSA A 5(1), pp. 5-18. 1988.

[20]. T. Trupke, R. A. Bardos, M. C. Schubert and W. Warta. Photoluminescence imaging of silicon wafers. APPLIED PHYSICS LETTERS, 2006.

[21]. D. R. Black and G. G. Long. 960-10 Special Publication on X-ray topography.

[22]. J. Parisi and R. M. Osgood. Into The Nano Era: Moore's Law Beyond Planar Silicon CMOS

[23]. C. Telidevara. Silicon wafer defect segmentation using modified pulse coupled neural network. 2011.

[24]. K. Choi, S. Ghosh, J. Lim and C. Lee. Removal efficiency of organic contaminants on si wafer by dry cleaning using UV and ECR plasma. Appl. Surf. Sci. 206(1), pp. 355-364. 2003.

[25]. K. Saga and T. Hattori. Identification and removal of trace organic contamination on silicon wafers stored in plastic boxes. Journal of the Electrochemical Society 143(10), pp. 3279-3284. 1996.

[26]. G. Lorenzi, K. H. Nguyen, C. Sanna, R. Orizio and G. Borionetti. COPs/particles discrimination using an automated surface inspection tool. pp. 39-49. 2003.

[27]. C. Kupfer, H. Roth and H. Dietrich. Defect requirements for advanced 300 mm DRAM substrates. Materials Science in Semiconductor Processing 5(4–5), pp. 381-386. 2002.

[28]. J. Green. Fast growing protrusions from epitaxial semiconductor surfaces. Metallurgical and Materials Transactions B 1(3), pp. 647-650. 1970.

[29]. Kuang-Peng Lin, Kai-Ming Ching, Kwo-Shu Huang and Shun-Liang Hsu. A study of implant damage induced thin oxide film expansion during photoresist dry etching. Presented at Reliability Physics Symposium, 2000. Proceedings. 38th Annual 2000 IEEE International.

[30]. H. C. Yau. Transform-based shape recognition employing neural networks. 1990.

[31]. F. Y. Shih. Image Processing and Pattern Recognition: Fundamentals and Techniques 2010.

[32]. R. Gore, J. Li, M. T. Manry, L. M. Liu, C. Yu and J. Wei. Iterative design of neural network classifiers through regression. Int. J. Artif. Intell. Tools 14(01n02), pp. 281-301. 2005.

[33]. M. -S. Chen and M. T. Manry. Basis vector analyses of back-propagation neural networks. Presented at Circuits and Systems, 1991., Proceedings of the 34th Midwest Symposium on. 1991.

[34]. K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. J.Mach.Learn.Res. 2pp. 265-292. 2002.

[35]. A. Khotanzad and Y. H. Hong. Invariant image recognition by zernike moments. Pattern Analysis and Machine Intelligence, IEEE Transactions on 12(5), pp. 489-497. 1990.

[36]. J. M. Zurada. Introduction to Artificial Neural Systems 1992.

[37]. D. W. Ruck, S. K. Rogers, M. Kabrisky, M. E. Oxley and B. W. Suter. The multilayer perceptron as an approximation to a bayes optimal discriminant function. Neural Networks, IEEE Transactions on 1(4), pp. 296-298. 1990.

[38]. M. T. Manry, X. Guan, S. J. Apollo, L. S. Allen, W. D. Lyle and W. Gong. Output weight optimization for the multi-layer perceptron. Presented at Signals, Systems and Computers, 1992. 1992 Conference Record of the Twenty-Sixth Asilomar Conference on. 1992

[39].]Mu-Song Chen and M.T. Manry, Basis Vector Analyses of Back Propagation Neural Networks, Presented at Proceedings of the 34th Midwest Symposium on Circuits and Systems 1991.

[40]. C. Yu, M. T. Manry, J. Li and P. Lakshmi Narasimha. An efficient hidden layer training method for the multilayer perceptron. Neurocomputing 70(1–3), pp. 525-535. 2006.

[41]. S. S. Malalur and M. T. Manry. Multiple optimal learning factors for feed-forward networks. Presented at SPIE Defense, Security, and Sensing. 2010

[42]. J. Li, M. T. Manry, P. L. Narasimha and C. Yu. Feature selection using a piecewise linear network. Neural Networks, IEEE Transactions on 17(5), pp. 1101-1115. 2006.

[43]. J. Sietsma and R. J. F. Dow. Neural net pruning-why and how. Presented at Neural Networks, 1988., IEEE International Conference on. 1988.

[44]. P. L. Narasimha, W. H. Delashmit, M. T. Manry, J. Li and F. Maldonado. An integrated growing-pruning method for feedforward network training. Neurocomputing 71(13), pp. 2831-2847. 2008.

Biographical Information

Aditi Godbole was born in 1989. She received her degree in Bachelor of Engineering form University of Mumbai, Maharashtra in Electronics and Telecommunication Engineering in August 2010. She obtained her Master of Science degree in Electrical Engineering from The University of Texas at Arlington in December 2012.