IMAGE PROCESSING OF NANOSTRUCTURES

by

KAVITHA ANANDAN

(Under the Direction of Hamid R. Arabnia)

ABSTRACT

One among the non-exhaustive list of applications where image processing is used is Nanotechnology. To visualize and characterize the nanostructures, high resolution Scanning Electron Microscope (SEM) and Atomic Force Microscopy (AFM) are ideally used. One of the crucial steps in image processing the nanostructures is image segmentation. There are various ways to attain segmentation. One way is the statistical approach. In this approach, we fit the foreground and background histogram data to the Gaussian curves and the parameters are found. The optimal threshold is found by solving a set of equations using the six parameters of the mixture of two Gaussian distribution. We have used Expectation-Maximization (EM), kmeans, combination of EM and kmeans, minimum distance method to determine the optimal threshold using the statistical approach. The comparisons of all these techniques are thoroughly studied for the simulated data as well as the real nanostructure images. We have also proposed a faster, preparation insensitive objective method for measuring the length, orientation and density of the single walled carbon nanotubes and the diameter of the nanoparticles using image processing techniques.

INDEX WORDS: Image processing, SEM, AFM, EM, Nanotubes, Nanostructures, Nanowires

IMAGE PROCESSING OF NANOSTRUCTURES

by

Kavitha Anandan

BE, Anna University, India, 2005

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment

of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2009

IMAGE PROCESSING OF NANOSTRUCTURES

by

KAVITHA ANANDAN

Major Professor:	Hamid R. Arabnia
Committee:	Daniel M. Everett Shelby H. Funk

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
May 2009

DEDICATION

I would like to dedicate this thesis to my parents Anandan and Suguna, siblings, in-laws and my valuable friends for their love and support in completing my thesis successfully.

ACKNOWLEDGEMENTS

I would like to thank my major advisor Dr. Hamid Arabnia and my committee members Dr. Daniel Everett, Dr. Shelby Funk for their valuable guidance, constant support and encouragement. I would also like to thank Dr. Marcus Lay, Dept. of Chemistry, Dr. Zhengwei Pan, Dept. of Physics and Dr. Guigen Zhang, Dept of BioEngineering for their help in providing the nanostructure images for my thesis.

I would like to thank my friends in UGA, Sridhar, Nag and Uma anna for their valuable time in discussing lot of concepts and also for their continuous support and encouragement.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION

The task of image analyzing becomes monumental when one tries to understand, characterize and extract relevant information from an image. The non-exhaustive list of applications where image processing is needed are medical and biological purposes, security and surveillance, multimedia, photo and video, robotic, biometrics and remote sensing [1-4] .

Recently image analysis is being used in the field of nanotechnology [1]. Nanotechnology is a field of applied science, which aims in the production and use of materials and structures engineered close to the atomic or molecular scale. These structures are referred as nanostructures, which possess one dimension at least less than 100 nanometers. In order to visualize and characterize this kind of nanostructure materials, high resolution Scanning Electron Microscope (SEM) and Atomic Force Microscopy (AFM) are ideally used [5]. The images captured are extensively analyzed and processed for the qualitative and quantitative information on many physical properties including size, length, density, morphology, surface texture and roughness. Although the images obtained by SEM and AFM are good enough to analyze the above mentioned properties, they cannot be analyzed in the raw image format as the images are abstract. Hence, the raw image has to undergo a number of image pre-processing techniques to make it easy for the researcher and also to obtain the correct information [2-4]. The pre-processing steps aims at contrast enhancement, restoration, compression, background subtraction.

Once the image has been pre-processed, the next crucial step is image segmentation, which is a process of associating similar image features together based on intensity, color and texture [6, 7]. The segmentation techniques are classified as intensity based segmentation thresholding [8], edge based segmentation [9] and region based segmentation. Among these, thresholding is the one which provides an easy and convenient way to perform segmentation on the basis of the different intensities in the foreground and background regions of an image [10].

Many image thresholding algorithms have been developed in the past several years [11] and still much research continues due to the existence of needs [12, 13]. Although there are so many methods present, none of them perform well under all conditions. Each and every method is based on certain conditions and works only under some assumptions. These methods fail when they are applied to SEM or AFM images of nanostructures because the images are fuzzy and abstract [1]. One example of such issues is that carbon nanotube images captured by AFM always produce a unimodal distribution, making it difficult to separate the foreground and background regions. This is one of the special cases of thresholding methods [10]. Another example is that nanorod images captured by SEM have a poor vision in separating the top view from the stem region [14]. This hinders the researchers from further processing of their nanostructure images forcing them to do such processes like measuring the length, orientation and density of the carbon nanotubes manually. Therefore it is very clear that a good thresholding method is required.

In the literature there are so many global image thresholding methods present that are either based on parametric or nonparametric approaches [13, 15-18]. The parametric approach is one which assumes a statistical model for the foreground and background class distributions. In many cases, Gaussian distribution is considered [17, 19, 20]. Although the methods based on the

Gaussian distribution [16, 17] are proven to be among the best thresholding methods, there are some serious drawbacks. The performance degrades when the initial parameters are random and the prior probabilities are unbalanced. In order to address these drawbacks and to find the best suitable method for thresholding the nanostructure images, in this work, we study various statistical algorithms like Expectation-Maximization (EM), kmeans and a combination of EM and kmeans to segment the image. We also study the novel concept of obtaining the threshold value from minimum distance method which is extensively used in statistics. The comparisons of all these techniques are thoroughly studied for the simulated data and the real nanostructure images in chapter 3. In chapter 2, we discuss the related and the background work.

Nanomaterials are of different types. Some of them are nanorod/nanowire, nanotube, nanobelt, nanocomb, nanosphere, nanodot and ultrathin nanofilm. One of the major focuses of recent nanoscience investigations is single walled nanotubes (SWNTs).

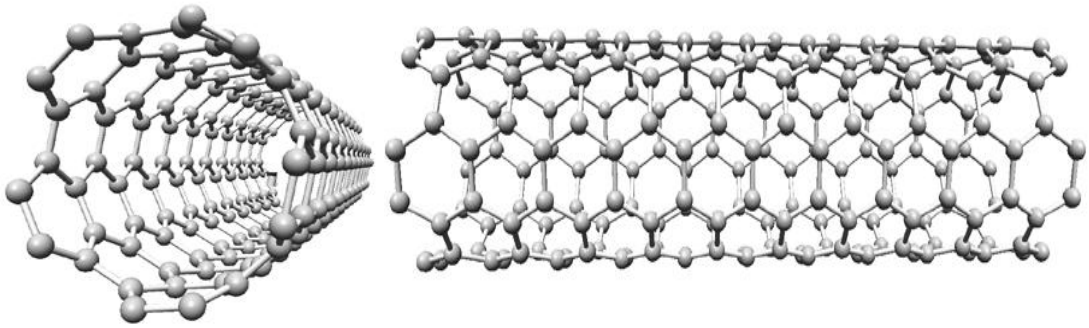

Figure 1.1: Single walled nanotube

SWNTs are unique nanostructures with remarkable mechanical, thermal and electrical properties. They are synthesized by various methods like arc discharge electron cyclotron resonance, carbon sublimation, laser ablation and pyrolysis of organometallic mixture sand chemical vapor deposition (CVD) [21]. They have a wide range of industrial, biological and

scientific applications, which includes reinforcing composite material, nanoscopic tunable field effect transistors and nanoscale quantum wires for electronic devices. The length of the SWNTs is an important parameter in determining its mechanical, thermal and electrical properties, and also it plays an important role in phase transitions and flow properties of rigid-rod dispersions [21]. In addition, determining the orientation and density of SWNTs plays a major role among the researchers in recent times [22].

Usually the length of the SWNTs is estimated by various techniques such as (i) dynamic and static light scattering, (ii) dynamic light scattering in combination with zeta potential measurements, (iii) ultracentrifugation, (iv) gel permeation chromatography, (v) matrix assisted laser ionization, (vi) membrane and vapour pressure osmometry, (vii) intrinsic viscosity measurements by capillary viscometers [21]. Among all the techniques available, dynamic light scattering have shown better results in estimating the length of the SWNTs. However, in this technique, it is difficult to make assumptions on the effect of light absorption by SWNTs. The zeta potential measurements assume that the SWNT length is much smaller than the wavelength of light. In order to overcome these drawbacks, we propose a faster, preparation insensitive objective method for measuring SWNTs length, orientation and density by image processing techniques. The orientation, length and the density determination algorithm of the SWNTs is discussed in chapter 4A.

Another important class of nanomaterials is called nanowires. Nanowires are solid wires with diameter less than 100 nanometers. Alternatively, nanowires are defined as structures that have a lateral size constrained to tens of nanometers or less and an unconstrained longitudinal size. There are different types of nanowires available. For example, metallic nanowires (Ni, Pt, Au), semiconducting (Si, InP, GaN) and insulating nanowires (SiO_2, TiO_2). These nanowires are

not observed spontaneously in nature and must be produced in a laboratory. The different methods to grow nanowire are vapor-liquid-solid (VLS), vapor-solid (VS), electrochemical deposition and solution growth method. Among all, VLS crystal growth mechanism is one of the widely used techniques to grow the wide range of nanowires. In this mechanism, a catalyst nanoparticle is used which directs the nanowire's growth and defines the diameter of the nanowire. For example, germanium is used as an efficient catalyst for high yield growth of very long, extremely straight and quasi-aligned arrays of zinc oxide nanowires [23]. One of the unique features of nanowire is that they are terminated at the growing ends by very large spherical particles with diameters varying from 0.5 to 4 microns. The particle size of the germanium is controlled by varying the growth temperature and the flow rate of the precursor material. During the growth of the nanowire at 500 ºC - 650 ºC, nanowires with the diameter and length in the ranges 50 nm - 400 nm and 50 μm – 200 μm are formed. However, the dimensions of the nanowires formed using this process are greatly depend on the size of the germanium particles. Therefore, measuring the diameter of the germanium particle is very critical in the growth of nanowires. In this study, we propose to find the diameter of this nanoparticle using Hough transform. The diameter determination algorithm of the nanoparticles is discussed in chapter 4B.

CHAPTER 2

RELATED WORK AND BACKGROUND

Maximum Likelihood Estimation:

It is one of the most popular statistical methods used for fitting a mathematical model to some data. The estimation by Maximum Likelihood Estimation (MLE) offers a way to tune the free parameters of the model to provide a good fit. It has widespread applications in various fields like psychometrics, econometrics, computational phylogenetics, nuclear and particle physics, communication systems, etc.

When a fixed set of data and underlying probability model are given, MLE chooses the values of the model parameters that make the data more likely than any other values of the parameters would make them. MLE gives an easy way to determine the solution in the case of a normal distribution and also works fine for other distributions as well. MLE begins with writing a mathematical expression known as likelihood function. Likelihood of a set of data is defined as the probability of obtaining that particular set of data given the chosen probability distribution model. The expression of the likelihood function contains some unknown model parameters. MLE maximizes this likelihood function and the values of the parameters that maximize the likelihood function are called as maximum likelihood estimates.

Computation of MLE [24]:

If x is a continuous random variable with pdf:

$$f(x;\theta_1,\theta_2.....\theta_k)$$

where $\theta_1, \theta_2, \ldots, \theta_k$ are k unknown constant parameters which need to be estimated. If N independent observations, $x_1, x_2, \ldots x_N$ are obtained, then the likelihood function is given by,

$$L(x_1, x_2, \ldots x_N | \theta_1, \theta_2, \ldots \theta_k) = L = \prod_{i=1}^{N} f(x_i; \theta_1, \theta_2 \ldots \theta_k)$$

where $i=1,2,\ldots N$

The logarithmic likelihood function is given by:

$$\Lambda = \ln L = \sum_{i=1}^{N} \ln f(x_i; \theta_1, \theta_2 \ldots \theta_k)$$

The maximum likelihood estimators of $\theta_1, \theta_2, \ldots, \theta_k$ are obtained by maximizing L or Λ.

By maximizing Λ, the maximum likelihood estimators of $\theta_1, \theta_2, \ldots, \theta_k$ are the simultaneous solutions of k equations such that:

$$\frac{\partial(\Lambda)}{\partial \theta_j} = 0, \; j = 1,2 \ldots k$$

MLE Method Using the Normal Distribution**:**

To obtain the MLE estimates for the mean, T' and standard deviation, σ_T for the normal distribution, start with the probability density function of the normal distribution which is given by:

$$f(T) = \frac{1}{\sigma_T \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{T-T'}{\sigma_T}\right)}$$

If $T_1, T_2 \ldots T_N$ are known times-to-failure, then the likelihood function is given by:

$$L(T_1, T_2 ..... T_N | T', \sigma_T) = L = \prod_{i=1}^{N} \left[\frac{1}{\sigma_T \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{T_I - T'}{\sigma_T}\right)^2} \right]$$

$$L = \frac{1}{(\sigma_T \sqrt{2\pi})^N} e^{-\frac{1}{2}\sum_{i=1}^{N}\left(\frac{T_I - T'}{\sigma_T}\right)^2}$$

Then:

$$\Lambda = \ln L = -\frac{N}{2}\ln(2\pi) - N \ln \sigma_T - \frac{1}{2}\sum_{i=1}^{N}\left(\frac{T_i - T'}{\sigma_T}\right)^2$$

Then taking the partial derivatives of Λ with respect to each one of the parameters and setting it equal to zero yields:

$$\frac{\partial(\Lambda)}{\partial T} = \frac{1}{\sigma_T^2}\sum_{i=1}^{N}(T_i - T') = 0$$

$$\frac{\partial(\Lambda)}{\partial \sigma_T} = -\frac{N}{\sigma_T} + \frac{1}{\sigma_T^3}\sum_{i=1}^{N}(T_i - T')^2 = 0$$

Solving these two equations gives

$$T' = \frac{1}{N}\sum_{i=1}^{N} T_i$$

$$\sigma_T^2 = \frac{1}{N}\sum_{i=1}^{N}(T_i - T')^2$$

$$\sigma_T = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(T_i - T')^2}$$

These solutions are only valid for data with no suspensions, i.e. all units are tested to failure. In cases in which suspensions are present, the methodology changes and the problem becomes much more complicated.

The parameters obtained from maximizing the likelihood function are estimators of the true value. It is clear that the sample size determines the accuracy of an estimator. If the sample size equals the whole population, then the estimator is the true value. Estimators have properties such as unbiasedness, sufficiency, consistency and efficiency.

Related work in EM:

There has been lot of researches done in the EM using MLE for image segmentation. In one such research, a novel parametric and global image histogram thresholding method is presented based on the estimation of the statistical parameters of object and background classes by the EM algorithm, under the assumption that these two classes follow a Generalized Gaussian (GG) distribution. The adoption of such a statistical model is because of its attractive capability to approximate a broad variety of statistical behaviors with a small number of parameters [12].

Related work in thresholding methods:

Many techniques have been proposed for global image thresholding which are based on parametric or nonparametric approaches. Among them, one of the earliest thresholding techniques available in the literature is the iterative selection method [17, 20]. In this technique, the threshold value is first initialized with the mean of the entire image histogram. The image is segmented using this initial threshold to produce two groups G1 and G2. Using this, T1 and T2 are obtained which are the average pixel values of G1 and G2. Again a new threshold is

calculated as the average of the T1 and T2. Using this as the initial threshold, the whole process is repeated until convergence is reached.

Another famous method is called Otsu method, which automatically performs the histogram shape based image thresholding. This technique assumes that the image to be thresholded contains two classes of pixels namely foreground and background, and then calculates the optimum threshold separating those two classes so that their within-class variance is minimal. The between-class variance is usually obtained by subtracting the within-class variance from the total variance of the combined distribution.

$$\sigma^2_{Between}(T) = \sigma^2 - \sigma^2_{Within}(T)$$

μ and σ are not dependent on threshold, thus minimizing the within-class which is same as maximizing the between-class variance [16].

Moment preserving thresholding is a method in which the threshold values are selected deterministically in such a way that the moments of an input image are preserved in the output image. This method is applicable for both global and local thresholding [25].

Some gray level picture thresholding using the entropy of the histogram is also proposed, in which the threshold value is selected by maximizing the sum of the entropies of the foreground and background classes of the image. Also, some methods have been proposed to minimize the cross-entropy between the input gray level image and the output in order to preserve the information [26].

One of the other entropy based method determines the optimal threshold by the maximization of the posterior entropy subject to inequality constraints derived from measures of the uniformity and shape of the regions in the image [27].

One other interesting entropy based method identifies the 2-D entropies from a bi-dimensional histogram that is constructed using the gray level values and the local average gray values [28].

Minimum cross entropy thresholding is one of the other entropy based techniques which selects the optimum threshold by minimizing the cross entropy between the original image and the segmented version [15]. This cross entropy is formulated on a pixel-to-pixel basis between the images. It is also based on the Kullback's minimum cross entropy principle.

Maximum entropy segmentation is based on the autocorrelation function of the image histogram [29]. It is one of the other entropy techniques which maximize the sum of the entropies computed from the autocorrelation function of the thresholded image histogram.

Minimum error thresholding is one of the thresholding techniques which optimizes a criterion function that is based on the Bayes classification for minimum error and Gaussian distribution assumption for foreground and background classes [19].

Maximum-likelihood thresholding is based on population mixture-model and discriminant criterion [30]. In this, minimizing the mean square errors between the original image and the output binary image is equivalent to the maximization of the likelihood of the conditional distribution in the mixture model, under the assumption of the Gaussian distribution with a common variance.

There are some thresholding techniques which are based on the fuzzy logic in which the threshold value is determined by minimizing the measure of the fuzziness of the original image [31, 32]. Based on this measurement, fuzzy range is detected to find the adequate threshold value within this range.

Image thresholding based on the EM algorithm and the Generalized Gaussian distribution [12], estimates the statistical parameters of the foreground and the background distribution using the expectation maximization algorithm.

Unimodal thresholding is an algorithm which is capable of performing bilevel thresholding for a unimodal distribution images [10]. It has various applications towards edges, difference images, optic flow, texture difference images, polygonal approximation of the curves, and image segmentation.

Related work in nanostructure image processing:

There has been a lot of research done in analyzing the nanostructure images using image processing. Image processing techniques are used for the analysis of micro and nano spaces in carbon materials [33]. In this research, the authors developed a quantitative analysis method for texture and structure of carbon materials containing the micro and nano spaces by using electron microscopy combined with 2D fast fourier transform image processing technique.

Image analysis of multiphased ceramics are done for the recognition and characterization of the different phases present in a (Zr, Sn)TiO_4 ceramic (with some amounts of La_2O_3 and NiO as sintering aids). In this work, the authors used various segmentation techniques like simple thresholding algorithm, Fisher's thresholding algorithm and region based segmentation to extract different phases from a scanning electronic microscope image of a polished section of the ceramic [34].

Automated sizing of DNA fragments in AFM images are studied using image processing techniques. The authors carried out various steps like segmentation, edge smoothing, thinning, removal of corner pixels, removal of invalid fragments, returning the end pixels removed by

thinning, length calculation using various image processing techniques like 3 x 3 averaging filter, automatic thresholding, fast parallel thinning algorithm, etc., to find the length of the DNA fragments [4].

Using the image processing techniques, a method for measuring electrospun nanofiber diameter is developed. In this, they used techniques like pruning, skeletonization, etc., to find the diameter of the nanofiber [35].

CHAPTER 3

COMPARITIVE STUDY OF DIFFERENT STATISTICAL BASED THRESHOLDING METHODS

In some cases, like in SEM and AFM images, when an image with a number of objects is considered, some are easily separable from the rest and some remain together. In that case, it's a challenging task to discover their statistical distributions. Each object is considered as an independent identity and each characterizes a statistical distribution model. This scenario is referred as mixture model [36].

There are different ways to estimate the mixture model. One way is to assume that each pixel is a member of one of the distribution model. Then the suitable parameters for the model are estimated. Once the parameters are obtained, then using suitable distribution formula, the pixels are classified. There are many statistical approaches to fit the mixture model to observed data. They are EM, kmeans, minimum distance, etc. We propose to compare and study the different methods discussed above.

Kmeans:

Kmeans clustering is an algorithm which classifies the given dataset into k clusters based on certain specific distance measurements. Kmeans is the most widely used and studied among all the clustering algorithms. This is one of the common techniques used for statistical data analysis, machine learning, data mining, pattern recognition, image analysis, neural networks, classification analysis and bioinformatics.

Kmeans is one of the fastest clustering algorithms and can be easily used in image segmentation. The algorithm is defined as follows:

Algorithm:

1. Assume the number of clusters is k.
2. Randomly guess k cluster center locations.
3. Determine which center is closest to each datapoint.
4. Find the centroid of each cluster and move the center to this location.
5. Repeat until convergence.

In our case, the clusters are going to be foreground and background. Hence, the value of k is two. The clusters center is given randomly. We choose to give the center values from the two peaks of the histogram. The input dataset is the image pixels. Then, each pixel finds the center to which it is closest to. For these new clusters, again a center is found. This is repeated until there are no changes in the cluster. The final clusters form the segmented image.

Once the mean is determined by the algorithm, the variance and proportions are calculated separately by their respective formulas.

$$mean = \frac{sum(a * h(a))}{sum(h(a))}$$

$$\text{var}iance = \frac{sum(a^2 * h(a)) - (sum(h(a)) * mean^2)}{sum(h(a)) - 1}$$

$$proportion = \frac{sum(h(a))}{n}$$

where a is the pixel value, $h(a)$ is the number of pixels in an image with grayscale value a, n is the total number of pixels in an image.

Some of the basic observations to be considered for kmeans algorithm are:

1. Initial grouping will determine the algorithm significantly. So, initial grouping should be done based on some reasonable assumptions.
2. Number of clusters k must be determined initially.
3. The initial conditions should be appropriate because the algorithm is very sensitive to the initial conditions. Different initial conditions may give different results ultimately affecting the algorithm to a great extent.

EM:

A general scenario like the observed data a and the set of unknown parameters θ are considered. Let us assume some parameters initially as $p(\theta)$. Then a posteriori probability function is given by $p(\theta/a)$. This is proportional to $p(a/\theta)p(\theta)$. If there is access to some other data b, then it is easy to maximize $p(a,b/\theta)p(\theta)$, where $p(a,b/\theta)$ is related to $p(a/\theta)$ through marginalization [37].

$$p(a/\theta) = \int p(b,a/\theta)db$$

EM is an iterative procedure between the expectation and maximization step which converges to a local maximum of the marginal a posteriori probability function

p(θ/a)=p(a/θ)p(θ) without calculating the marginal likelihood *p(θ/a)*. Therefore, EM estimates the probability that the data is present in certain cluster.

EM is generally based on Bayesian theory. EM is defined as [36]

- We assume the algorithm estimates *M* clusters which can also be referred as classes C_j where *j=1,2,3,…….M*.
- If there are *N* input vectors x_k, *k=1, 2…..N*, then the probability $P(C_j | x_k)$ to belong to a certain cluster is calculated by the algorithm.
- The highest probability will point to that particular vector's cluster.

Computation of EM:

The image pixels show a different behavior at various parts of the image, so there is a mixture distribution of the pixels. In our case, the mixture distribution produces a probabilistic model composed of two classes namely foreground and background. The data is assumed to be distributed as Gaussian and the general equation for bimodal Gaussian is

$$g(t) = p_1\phi_1(t) + p_2\phi_2(t) \text{ ----------(1)}$$

where $p_1 + p_2 = 1$

p_1 is the probability that the data is chosen from class 1

p_2 is the probability that the data is chosen from class 2

$\phi_1(t)$ and $\phi_2(t)$ denotes the probability density of the two Gaussian distribution

$$\phi_i(t) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{\frac{-(t-\mu_i)^2}{2\sigma_i^2}} \qquad i = 1, 2$$

where μ_l and σ_l^2 are the mean and variance of pixel values in category *l*.

t is the input pixel.

let $\theta = (p_1, p_2, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2)$, $p_2 = 1 - p_1$

then $p(i \mid \theta) = p_i$

$$p(t \mid i, \theta) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{\frac{-(t-\mu_i)^2}{2\sigma_i^2}}$$

$$p(t \mid \underline{\theta}) = \sum_i p(t, i \mid \theta)$$

$$p(t \mid \underline{\theta}) = \sum_i p(t \mid i, \theta) p(i \mid \theta)$$

We want to find the parameters $\underline{\theta}$ that maximize the likelihood of N datapoints under the above generative model

$$L = \sum_{n=1}^{N} \ln\ p(t_n \mid \underline{\theta})$$

Determining the sum of the solution for *L* is difficult. So, we need to use EM.

E-step:

Compute $p(i \mid t_n, \theta^{old}) = \dfrac{p(t_n \mid i, \theta^{old}) p(i \mid \theta^{old})}{\sum_{i'=1}^{2} p(t_n \mid i', \theta^{old}) p(i' \mid \theta^{old})}$

M-step:

$$\frac{\partial}{\partial \theta_i} Q(\theta) = \frac{\partial}{\partial \theta_i} \sum_n \sum_i p(i \mid t_n, \theta^{old})(\ln(\, p(t_n \mid i, \theta)) + \ln(\, p(i \mid \theta))$$

$$\ln(\, p(t_n \mid i, \theta) = -\frac{1}{2}\ln(2\pi) - \frac{1}{2}\ln(\sigma_i^2) - \frac{1}{2}\frac{(t - \mu_i^2)}{\sigma_i^2}$$

$$\ln(\, p(i \mid \theta)) = \ln(\, p_i)$$

$$Q(\theta) = \sum_n \sum_{i'} p(i' \mid t_n, \theta^{old}) \left(-\frac{1}{2}\ln(2\pi) - \frac{1}{2}\ln(\sigma_{i'}^2) - \frac{1}{2}\frac{(t_n - \mu_{i'}^2)}{\sigma_{i'}^2} + \ln(\, p_{i'}) \right)$$

$$\frac{\partial Q(\theta)}{\partial \mu_i} = \sum_n \sum_{i'} p(i' \mid t_n, \theta^{old}) \left(-\frac{1}{2\sigma_{i'}^2} \frac{\partial}{\partial \mu_i} (t_n - \mu_{i'})^2 \right)$$

$$= \sum_n \sum_{i'} p(i' \mid t_n, \theta^{old}) \left(-\frac{1}{2\sigma_{i'}^2} \right) (-1) 2 (t_n - \mu_i)$$

$$= \sum_n p(i \mid t_n, \theta^{old}) \left(\frac{1}{\sigma_i^2} \right) (t_n - \mu_i)$$

Equating to 0,

$$\mu_i = \frac{\sum_n p(i \mid t_n, \theta^{old}) t_n}{\sum_n p(i \mid t_n, \theta^{old})}$$

Similarly by taking the derivatives with respect to σ_i and equating to 0, we get

$$\sigma_i = \frac{\sum_n p(i \mid t_n, \theta^{old})(t_n - \mu_i)^2}{\sum_n p(i \mid t_n, \theta^{old})}$$

Similarly for p_i with the constraint $p_1 + p_2 = 1$, we can show that

$$p_i = \frac{1}{N} \sum_n p(i \mid t_n, \theta^{old})$$

Hence the EM algorithm for the mixture of two Gaussian [45, 46],

Step 1: Randomly initialize parameters

$$\theta^{old} = (p_1, p_2, \mu_1, \mu_2, \sigma_1, \sigma_2)$$

Step 2: E-step

Using the parameters from step 1, compute the posterior probability function.

$$p(i \mid t_n, \theta^{old}) = \frac{(2\pi\sigma^{2}{}_{i}^{old})^{-\frac{1}{2}} e^{\left(-\frac{1}{2\sigma^{2}{}_{i}^{old}}(t_n-\mu_i^{old})^2\right)} p_i}{\sum_{i'=1}^{2} (2\pi\sigma^{2}{}_{i'}^{old})^{-\frac{1}{2}} e^{\left(-\frac{1}{2\sigma^{2}{}_{i'}^{old}}(t_n-\mu_{i'}^{old})^2\right)} p_{i'}}$$

Step 3: M-step

Compute the new set of parameters using the result from step 2

$$p_i^{new} = \frac{1}{N}\sum_n p(i \mid t_n, \theta^{old})$$

$$\mu_i^{new} = \frac{\sum_n p(i \mid t_n, \theta^{old}) t_i}{\sum_n p(i \mid t_n, \theta^{old})}$$

$$\sigma_i^{new} = \frac{\sum_i p(i \mid t_n, \theta^{old})(t_n - \mu_i^{new})^2}{\sum_i p(i \mid t_n, \theta^{old})}$$

Step 4: $\theta^{old} = \theta^{new}$

Step 5: Continue until convergence

The input parameters are randomly initialized to the algorithm. EM is more sensitive to the input parameters. Hence, the initial set of parameters has to be more reasonable. We have used the peak values from the histogram as the initial mean and the variance depending on the image and the histogram. The proportions have been set as 0.5.

Binary image generation:

Once the final parameters are obtained, the classification is done by any Bayesian decision criteria. We prefer using the posterior probability of the two classes. So, the optimal label *t* is assigned to each pixel *x* of the image in such a way,

$t = 0$ if $z_1[x] >= z_2[x]$

$t = 255$ otherwise

where $z_i(x) = \dfrac{\dfrac{p_i}{\sqrt{2\pi\sigma_i^2}} e^{\frac{-0.5(x-\mu_i)^2}{\sigma_i^2}}}{\dfrac{p_1}{\sqrt{2\pi\sigma_1^2}} e^{\frac{-0.5(x-\mu_1)^2}{\sigma_1^2}} + \dfrac{p_2}{\sqrt{2\pi\sigma_2^2}} e^{\frac{-0.5(x-\mu_2)^2}{\sigma_2^2}}}$

Finding the optimal threshold:

By recalling the equation 1, the best classification criterion where the misclassification of the pixels is very least, allocates pixels with value t to category 1 if [38]

$p_1\phi_1(t) >= p_2\phi_2(t)$

$\dfrac{p_1}{p_2} >= \dfrac{\phi_2(t)}{\phi_1(t)}$

$\dfrac{p_1}{p_2} >= \dfrac{\dfrac{1}{\sqrt{2\pi\sigma_2^2}} e^{\frac{-(t-\mu_2)^2}{2\sigma_2^2}}}{\dfrac{1}{\sqrt{2\pi\sigma_1^2}} e^{\frac{-(t-\mu_1)^2}{2\sigma_1^2}}}$

$$\frac{p_1}{p_2} >= \frac{\sigma_1}{\sigma_2} e^{-\left\{\frac{(t-\mu_2)^2}{2\sigma_2^2}-\frac{(t-\mu_1)^2}{2\sigma_1^2}\right\}}$$

$$\ln\left\{\frac{\sigma_2 p_1}{\sigma_1 p_2}\right\} <= \frac{(t-\mu_2)^2}{2\sigma_2^2} - \frac{(t-\mu_1)^2}{2\sigma_1^2}$$

$$\ln\left\{\frac{\sigma_2 p_1}{\sigma_1 p_2}\right\} <= \left\{\frac{\sigma_1^2(t^2+\mu_2^2-2t\mu_2)-\sigma_2^2(t^2+\mu_1^2-2t\mu_1)}{2\sigma_1^2\sigma_2^2}\right\}$$

$$\ln\left\{\frac{\sigma_2 p_1}{\sigma_1 p_2}\right\}^2 <= \left\{\frac{t^2}{\sigma_2^{\,2}}+\frac{\mu_2^2}{\sigma_2^{\,2}}-\frac{2t\mu_2}{\sigma_2^{\,2}}-\frac{t^2}{\sigma_1^2}-\frac{\mu_1^2}{\sigma_1^2}+\frac{2t\mu_1}{\sigma_1^2}\right\}$$

$$t^2\left\{\frac{1}{\sigma_1^2}-\frac{1}{\sigma_2^2}\right\}-2t\left\{\frac{\mu_1}{\sigma_1^2}-\frac{\mu_2}{\sigma_2^2}\right\}+\left\{\frac{\mu_1^{\,2}}{\sigma_1^2}-\frac{\mu_2^{\,2}}{\sigma_2^2}+\log\frac{\sigma_1^2 p_2^2}{\sigma_2^2 p_1^2}\right\}$$

$$t^2 A - 2tB + C <= 0$$

$$A = \frac{1}{\sigma_1^{\,2}} - \frac{1}{\sigma_2^{\,2}}$$

$$B = \frac{\mu_1}{\sigma_1^2} - \frac{\mu_2}{\sigma_2^2}$$

$$C = \frac{\mu_1^{\,2}}{\sigma_1^2} - \frac{\mu_2^{\,2}}{\sigma_2^2} + \log\frac{\sigma_1^2 p_2^2}{\sigma_2^2 p_1^2}$$

Solving these equations, we get

$$t_1 = \frac{B+\sqrt{B^2-AC}}{A}$$

$$t_2 = \frac{B - \sqrt{B^2 - AC}}{A}$$

Using the above threshold formula, the threshold values are obtained. This is called as the optimal threshold.

EM with initial values from kmeans:

The main drawback of EM is the initial parameters because EM is more sensitive to initial parameters. The initial parameters should always be given a reasonable value. In order to tackle this issue in EM and also to increase its stability and capability, we have used the results of the kmeans as the seed points and performed EM. This is expected to show us some good results. As we expected, this improved EM method showed us some great resultant accuracy.

Minimum distance estimation:

The minimum distance estimators [39] have shown excellent robustness properties compared to the maximum likelihood which has practical deficiencies like lack of resistance to outliers and the general nonrobustness with respect to model misspecification. Some methods like Hellinger distance and L_1 error have shown great advantages since they are dimensionless. Certain approximations are typically encountered for these distance measures. Parametric and nonparametric methods rarely use the same estimation criteria. Parametric methods rely on maximum likelihood while nonparametric methods rely on L_2 (integrated square error criterion).

The parametric L_2E originates in the derivation of the nonparametric least squares cross-validation for choosing the bin width, h, of the histogram. Now the integral should be found $\int \hat{g}(x)dx$ and consider minimizing an estimate of ISE with respect to h.

$$\hat{h} = \arg\min_h \int \left[\hat{f}_h(x) - f(x) \right]^2 dx \text{----------(1)}$$

$$\hat{h} = \arg\min_h \left[\int \hat{f}_h(x)^2 dx - 2\int \hat{f}_h(x) f(x) dx + \int f(x)^2 dx \right] \text{----------(2)}$$

$$\hat{h} = \arg\min_h \left[\int \hat{f}_h(x)^2 dx - 2\hat{E}\left[\hat{f}_h(X) \right] \right] \text{----------(3)}$$

From equation (2), the minimizing value of h is not changed by eliminating $\int f(x)^2 dx$, an constant and the first integral can be evaluated exactly any value of h. The average height of the histogram with bin width h can be found from the remaining part of the equation. The unbiased estimate by partitioning the sample into n-1 points for estimation and 1 point for evaluation, $\hat{f}_{h,-i}(x_i)$ and then cycling over all n points and averaging, we get,

$$\hat{h} = \arg\min_h \left[\int \hat{f}_h(x)^2 dx - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{h,-i}(x_i) \right] \text{----------(4)}$$

$$\hat{h} = \arg\min_h \left[\frac{2}{(n-1)h} - \frac{(n+1)}{n^2(n-1)h}\sum_k \nu_k^2 \right]$$

where ν_k is the bin count of $B_k = (x_0 + kh, x_0 + (k+1)h)$ and x_0 is the origin of the bin. In the parametric setting with model $f(x|\theta)$, equation (1) can be written as

$$\hat{\theta} = \arg\min_{\theta} \int [f(x|\theta) - f(x|\theta_0)]^2 \, dx \text{ ----------(5)}$$

where the true parameter θ_0 is unknown. The expected height of the density, $\int f(x|\theta) f(x|\theta_0) dx$, is the key quantity to estimate. Data partitioning is not required in the parametric setting and the entire random sample is available to estimate the average height of $f(x|\theta)$ .Thus, the minimized parametric integrated square error criterion is

$$\hat{\theta}_{L_2E} = \arg\min_{\theta} \left[\int f(x|\theta)^2 dx - \frac{2}{n} \sum_{i=1}^{n} f(x_i|\theta) \right] \text{----------(6)}$$

Correct parametric family is assumed in the equation (6) and this is also used in the situation where the assumed parametric form is known to be incorrect in order to achieve robustness.

Computation of minimum distance:

There are several L_2E functional. The univariate is denoted by $\phi(x|\mu, \sigma^2)$ and the parametric density model for the mixture model is denoted by

$$f(x|\theta) = \sum_{k=1}^{K} w_k \phi(x|\mu_k, \sigma_k^2)$$

This kind of a mixture fitting has always been a difficult task in practice. The L_2E criterion works fine for these tasks and it is particularly easy to apply with the use of the following identity:

$$\int_{-\infty}^{\infty} \phi(x \mid \mu_1, \sigma_1^2)\phi(x \mid \mu_2, \sigma_2^2)dx = \phi(\mu_1 - \mu_2 \mid 0, \sigma_1^2 + \sigma_2^2)$$

In our case, it is univariate bimodal distribution. So, by solving the set of equations for k=2, we get

$$L_2 E(w_1, \mu_1, \mu_2, \sigma_1, \sigma_2) = \frac{w_1^2}{2\sqrt{\pi\sigma_1}} + \frac{(1-w_1)^2}{2\sqrt{\pi\sigma_2}} + 2w_1(1-w_1)\phi(\mu_1 - \mu_2 \mid 0, \sigma_1^2 + \sigma_2^2)$$

$$-\frac{2}{n}\sum_{i=1}^{n}[w_1\phi(x_i \mid \mu_1, \sigma_1^2) + (1-w_1)\phi(x_i \mid \mu_2, \sigma_2^2)]$$

By minimizing the above equation with the initial set of parameters, we get the final value and the final set of parameters. The initial set of parameters is given in the same way as discussed above.

Testing and comparisons:

In order to assess the effectiveness of these approaches, we have used both the simulated and the real images in our experiments. The simulated data is obtained with different statistical characteristics and it is tested for all our approaches namely EM, kmeans, EM with initial set of values from kmeans and minimum distance method. Once the parameters are obtained by all these methods, the classification is done using two methods which we already discussed earlier. They are posterior probability function and the optimal threshold formula. The results are also tested using some real nanostructure images.

Experiments on simulated data:

Test Case 1:

$\mu 1 = 50$, $\mu 2 = 100$

$\sigma 1 = 15$, $\sigma 2 = 15$

P1=0.5, P2=0.5

Optimal threshold=75

Dataset=1000

Table 3.1: Comparisons of all the methods showing all the parameters and the threshold value that is been obtained for test case 1

	Methods	Classification	Mean		Variance		Proportion		Threshold
1	EM	PPF	48.75	100.50	242.65	233.82	0.50	0.49	75
2	Kmeans	-	47.93	101.06	208.23	210.24	0.49	0.50	75
3	EM with kmeans	PPF	48.06	100.92	220.12	218.09	0.49	0.50	75
4	Minimum distance method	PPF	48.92	99.28	219.04	249.64	0.49	0.50	75

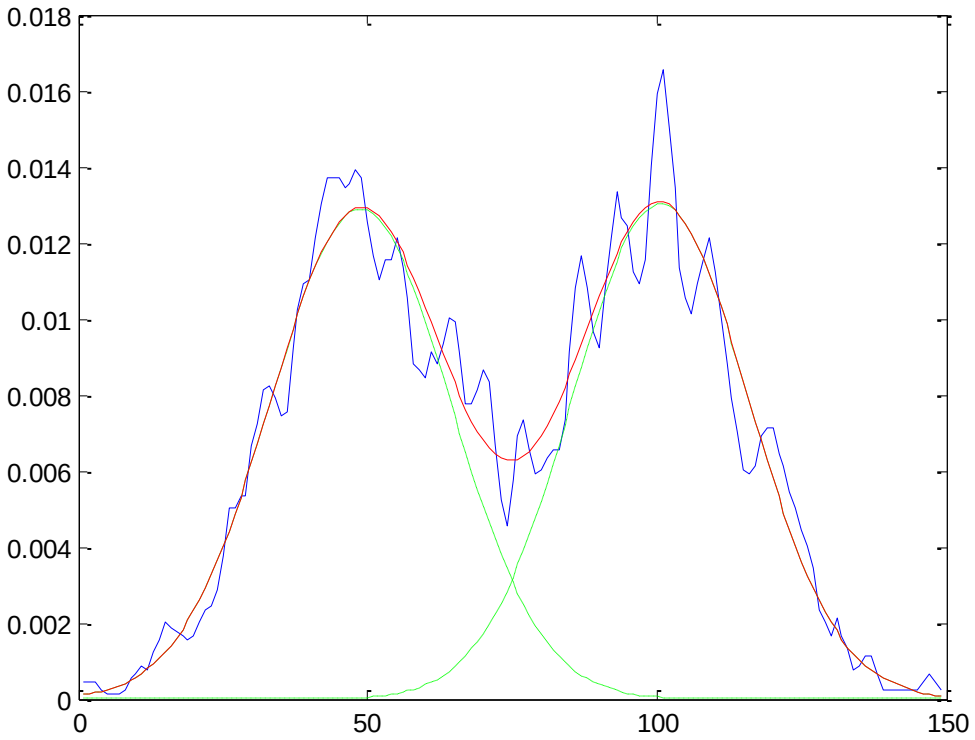

Figure 3.1: Histogram obtained for the simulated data (blue curve) and the Gaussian fit to that histogram (green curve) for test case 1.

Test Case 2:

$\mu 1 = 50$, $\mu 2 = 100$

$\sigma 1 = 15$, $\sigma 2 = 15$

P1=0.9, P2=0.1

Optimal threshold=85

Dataset=1000

Table 3.2: Comparisons of all the methods showing all the parameters and the threshold value that is been obtained for test case 2

	Methods	Classification	Mean		Variance		Proportion		Threshold
1	EM	PPF	47.48	98.71	220.81	200.13	0.91	0.08	84
2	Kmeans	-	43.16	79.63	143.94	324.20	0.76	0.23	64
3	EM with kmeans	PPF	41.66	74.44	133.13	312.68	0.69	0.30	64
4	Minimum distance method	PPF	49.25	100.66	196.90	218.44	0.89	0.10	85

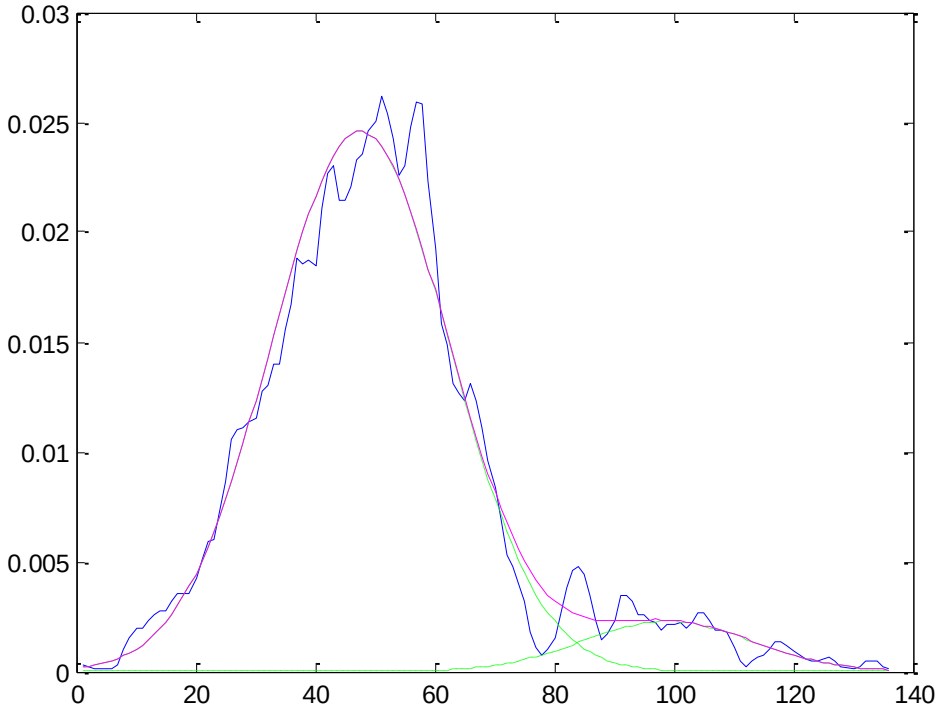

Figure 3.2: Histogram obtained for the simulated data (blue curve) and the Gaussian fit to that histogram (green curve) for test case 2

Test Case 3:

$\mu 1 = 50$, $\mu 2 = 100$

$\sigma 1 = 15$, $\sigma 2 = 15$

P1=0.9, P2=0.1

Optimal threshold=85

Dataset=100000

Table 3.3: Comparisons of all the methods showing all the parameters and the threshold value that is been obtained for test case 3

	Methods	Classification	Mean		Variance		Proportion		Threshold
1	EM	PPF	70.22	113.99	208.90	353.95	0.86	0.13	99
2	Kmeans	-	67.30	106.82	147.23	289.00	0.77	0.22	86
3	EM with kmeans	PPF	66.03	102.06	137.36	320.96	0.72	0.27	87
4	Minimum distance method	PPF	50.01	100.37	204.49	227.40	0.90	0.09	85

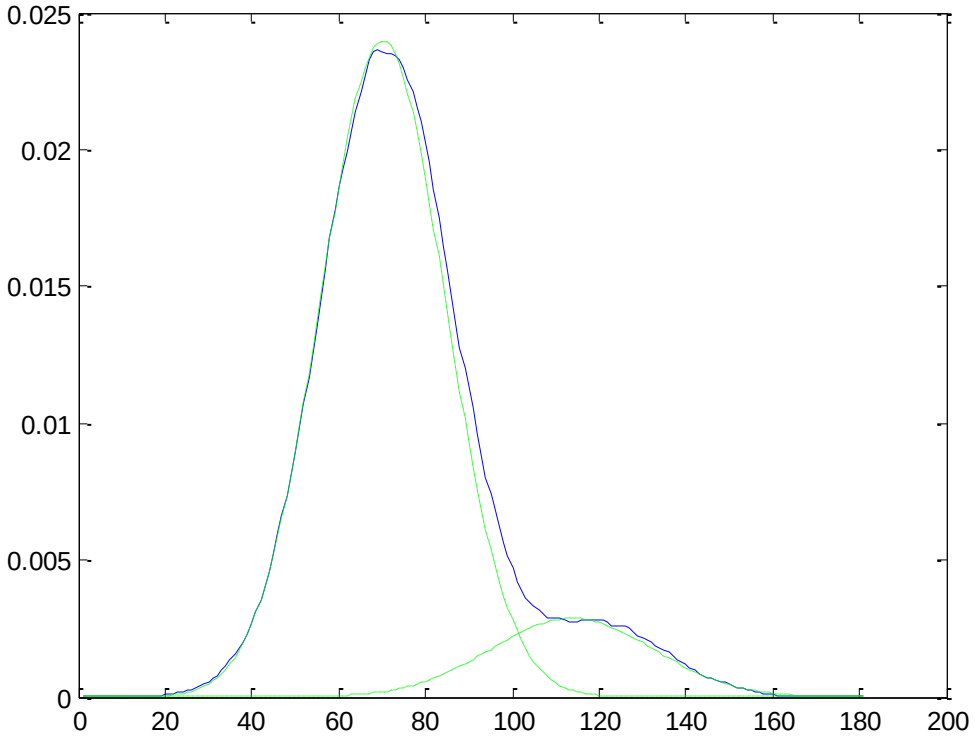

Figure 3.3: Histogram obtained for the simulated data (blue curve) and Gaussian fit to that histogram (green curve) for test case 3

Test Case 4:

$\mu 1 = 50$, $\mu 2 = 100$

$\sigma 1 = 15$, $\sigma 2 = 15$

P1=0.9, P2=0.1

Optimal threshold=85

Dataset=200000

Table 3.4: Comparisons of all the methods showing all the parameters and the threshold value that is been obtained for test case 4

	Methods	Classification	Mean		Variance		Proportion		Threshold
1	EM	PPF	72.24	116.64	211.61	343.30	0.86	0.13	102
2	Kmeans	-	68.52	106.55	139.64	299.02	0.75	0.24	87
3	EM with kmeans	PPF	67.44	102.60	133.78	326.91	0.69	0.30	88
4	Minimum distance method	PPF	52.01	101.90	208.43	232.57	0.89	0.10	85

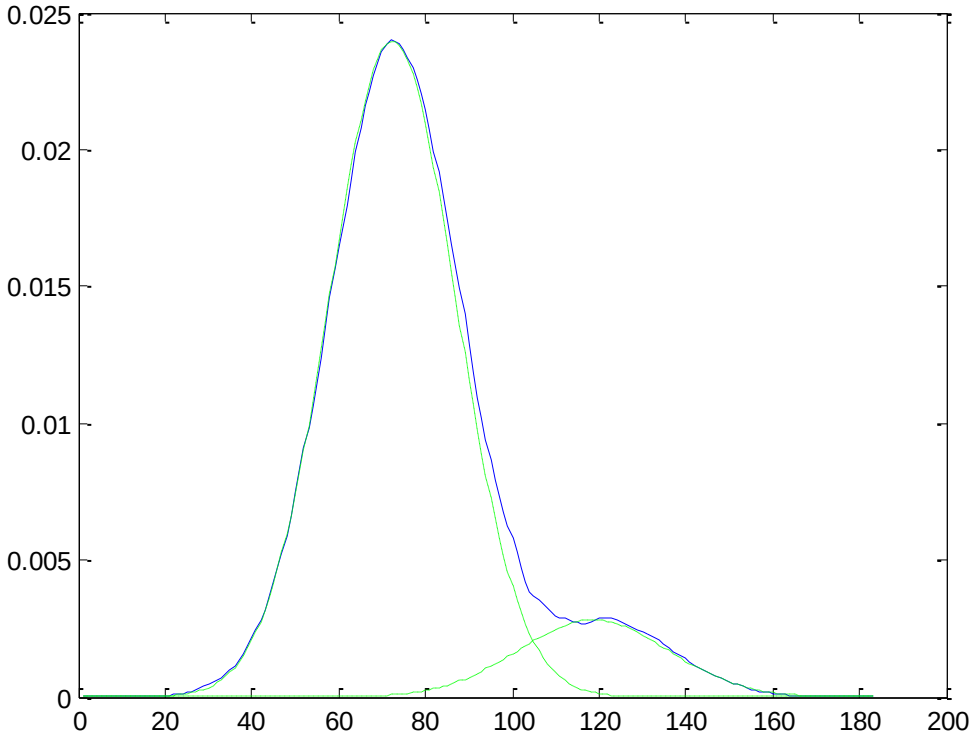

Figure 3.4: Histogram obtained for the simulated data (blue curve) and the Gaussian fit to that histogram (green curve) for test case 4.

Experiments on real images:

The performance of the algorithms are also tested with the real images of nanorods which is shown below

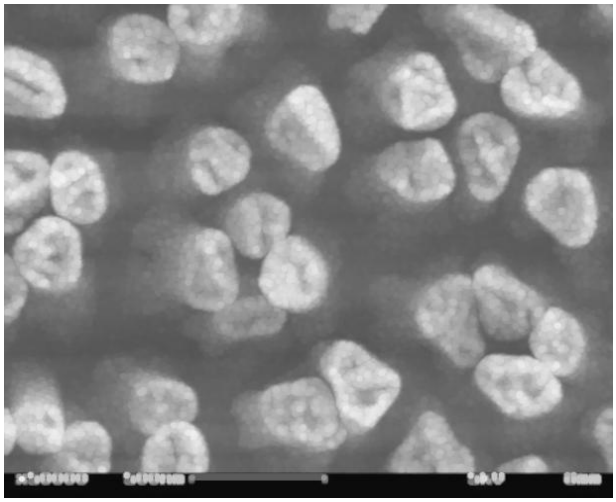

Figure 3.5: SEM image of nanorods

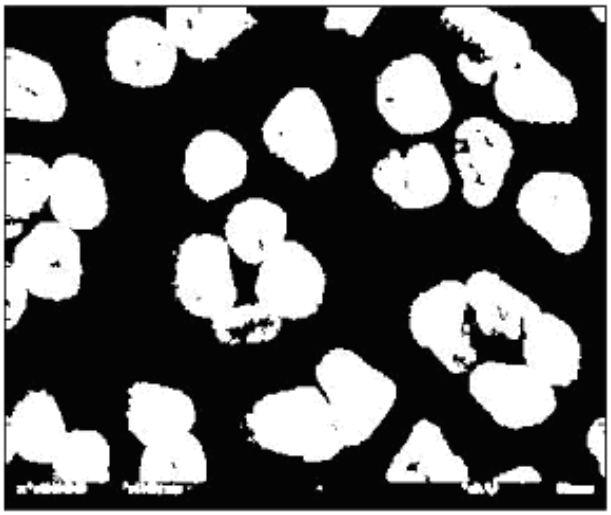

Figure 3.6: Image obtained after applying Kmeans

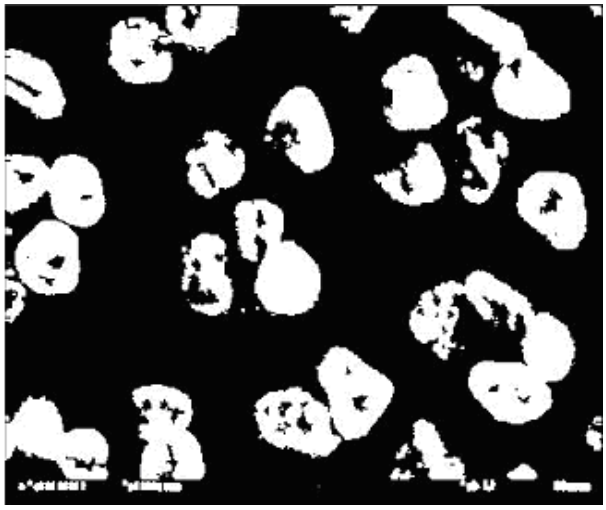

Figure 3.7: Image obtained after applying EM

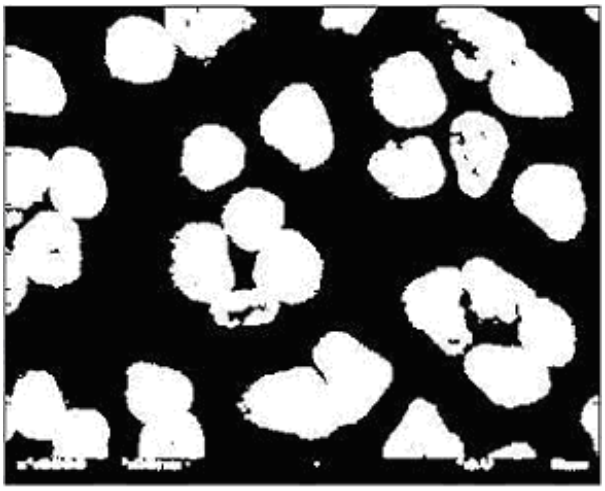

Figure 3.8: Image obtained after applying EM with initial values from kmeans

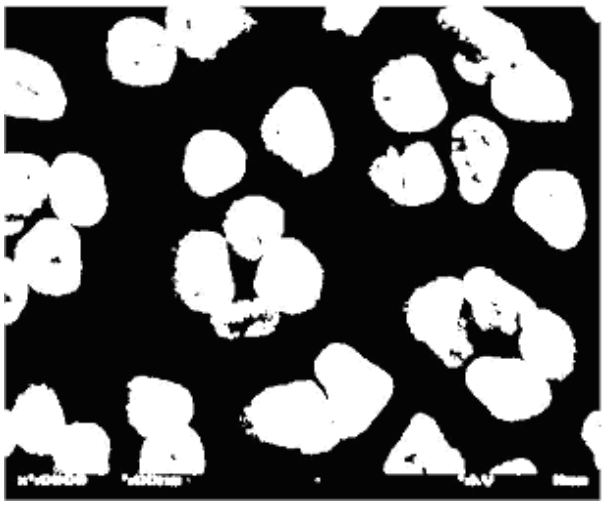

Figure 3.9: Image obtained after applying Otsu's method

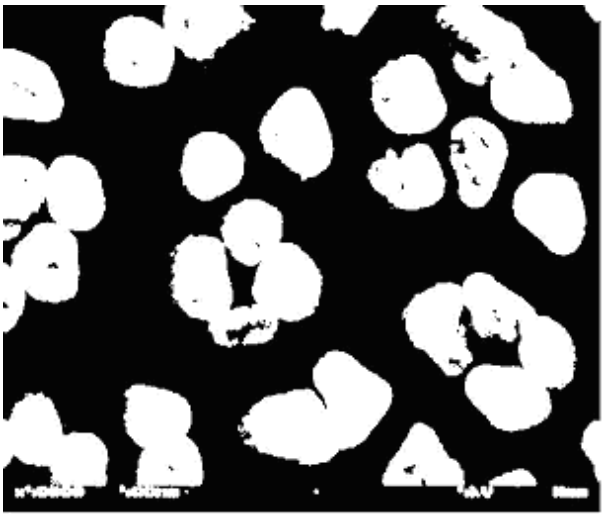

Figure 3.10: Image obtained after applying Ridler and Calvard's method

Discussion:

In test case 1, we have obtained the threshold values for the simulated data with statistical property mean=50,100, variance=225,225, probability=0.5,0.5 for the two classes. The dataset that we used is 1000. The optimal threshold value for this setup is 75. This is obtained by solving

the set of equations for the condition: The best classification criterion where the misclassification of the pixels is very least, allocates pixels with value t to category 1 if

$$p_1\phi_1(t) >= p_2\phi_2(t)$$

which we have already discussed. The final equation obtained is

$$t^2\left\{\frac{1}{\sigma_1^2}-\frac{1}{\sigma_2^2}\right\}-2t\left\{\frac{\mu_1}{\sigma_1^2}-\frac{\mu_2}{\sigma_2^2}\right\}+\left\{\frac{\mu_1^{\,2}}{\sigma_1^2}-\frac{\mu_2^{\,2}}{\sigma_2^2}+\log\frac{\sigma_1^2 p_2^2}{\sigma_2^2 p_1^2}\right\}$$

$$t^2A-2tB+C <= 0$$

In this we consider different cases,

(a) If $\sigma_1^2 = \sigma_2^2$, the equation is simplifies to one of allocating pixels with value t to class 1 if

$$2tB >= C$$

(b) If $p_1 = p_2$ and $\mu_1 < \mu_2$, the equation is simplified as $t <= \frac{1}{2}(\mu_1 + \mu_2)$

(c) If $B^2 < AC$, the equation has no real roots. All pixels are classified as 1 if $\sigma_1^2 > \sigma_2^2$ or 2 if $\sigma_1^2 < \sigma_2^2$

(d) The roots of the equation are obtained by $t_1, t_2 = \frac{B \pm \sqrt{B^2 - AC}}{A}$

When the threshold values obtained by EM, kmeans, EM with kmeans, minimum distance are considered for test cases 1 and 2, the results are

1) When the probabilities are equal, (i.e) $p_1 = p_2 = 0.5$, all the approaches produced a reasonable value equivalent to 75.

2) When the probabilities are unequal, (i.e.) $p_1 = 0.9, p_2 = 0.1$, the performance of kmeans degraded suddenly. This might be because of the kmeans properties like

    i. When the numbers of data are few, initial grouping will determine the cluster significantly.

    ii. The real cluster, using the same data, if it is inputted in a different order may produce different cluster if the number of data is less.

3) The minimum distance method performs well in all the above conditions.

Our concern is to check how the algorithm works in case of the unimodal situation although it is bimodal implicitly. This is the reason we choose the probabilities as 0.9 and 0.1. In order to check whether the same results continue for different dataset, we increased the sample size to 100000 and produced the results (test case 3). The results are little surprising at this point. The observations are

1) EM algorithm's performance decreased rapidly. This might be because of the initialized variables to the algorithm, as we know it is very sensitive to its initial values.

2) Kmeans is very sensitive to small dataset and the performance can be degraded at any point. When it is exposed to huge dataset like 100000, performance increased rapidly. The algorithm also performs very well when the number of clusters is small. This entire scenario helped the algorithm to perform well for this test case.

3) When the results of the kmeans are given as the seed point to the EM algorithm, performance increased. This is because of the obvious reason that the initial parameters are from a well established algorithm.

4) The minimum distance method have maintained its robustness even for the test case 3 by picking up a good threshold value

In order to confirm these results, we again tested the algorithm for one more case by increasing the sample size to 200000(test case 4). The results have not changed for any algorithm and this helped us to obtain the following conclusions.

1) EM performs well for dataset provided the input parameters are a good guess. We are suggesting using kmeans for the initial parameters.

2) When the output of kmeans is used as the initial parameters for the EM, the performance of EM gives better result for simulation setup as well as the real image.

3) The minimum distance method showed outstanding results and had retained its robustness considerably at all conditions.

4) The method we use for thresholding performed as good as the existing methods in the literature.

The results obtained from the real images confirm our conclusions, EM with random initial values do not perform well. This is confirmed from the image which shows loss of complete nanorods, noise in the homogeneous regions (figure 3.7). Kmeans as expected shows a better performance compared to EM by retaining the structure of the nanorods with little loss of data, removes lot of noise, but still has minute holes (figure 3.6). When EM is applied with initial

values from kmeans, the image shows a considerable improvement by removing even the small holes, removing noise and also retaining the structure of nanorods to a great extent (figure 3.8). Even the Otsu's method and Ridler and Calvard's method have not shown great performance when compared to our best methods.

CHAPTER 4A

LENGTH, ORIENTATION AND DENSITY DETERMINATION ALGORITHM FOR NANOTUBES

AFM:

Scanning Probe Microscope (SPM) is a branch of microscopy that is used to image and measure properties of chemical, material and biological surfaces. They form images of surfaces by raster scanning the probe on the specimen. This helps in recording the probe-surface interaction as a function of position. There are various scanning probe techniques and the most common techniques are Scanning Tunneling Microscopy (STM), Near-Field Scanning Optical Microscopy (NSOM), and AFM [40]. STM measures a weak electrical current that is flowing between the tip and the sample as they are placed a very large distance apart. NSOM scans a very small light source very close to the sample. Detection of this light energy forms the image. AFM is developed to overcome the drawbacks of these techniques which can image only conducting or semiconducting surfaces. AFM is capable of imaging any type of surfaces like polymers, ceramics, composites, glass, biological samples, etc. Therefore, AFM is considered to be the best compared to its conventional techniques. One of the main advantage of AFM is, it can probe the sample and make measurements in all the three dimensions, thus providing the 3-D images of the sample surface. For a good clean sample, AFM is capable of obtaining resolution in the X-Y plane ranges from 0.1 to 1.0 nm and in the z direction is 0.01 nm. AFM operates by rastering a sharp tip across the sample surface [40]. This sharp tip is attached

to a cantilever which maintains an extremely low force, thereby pushing the tip against the sample as it rasters. The repulsive force between the tip and sample or the actual tip deflection is recorded relative to spatial variation and then converted into an image of the sample surface.

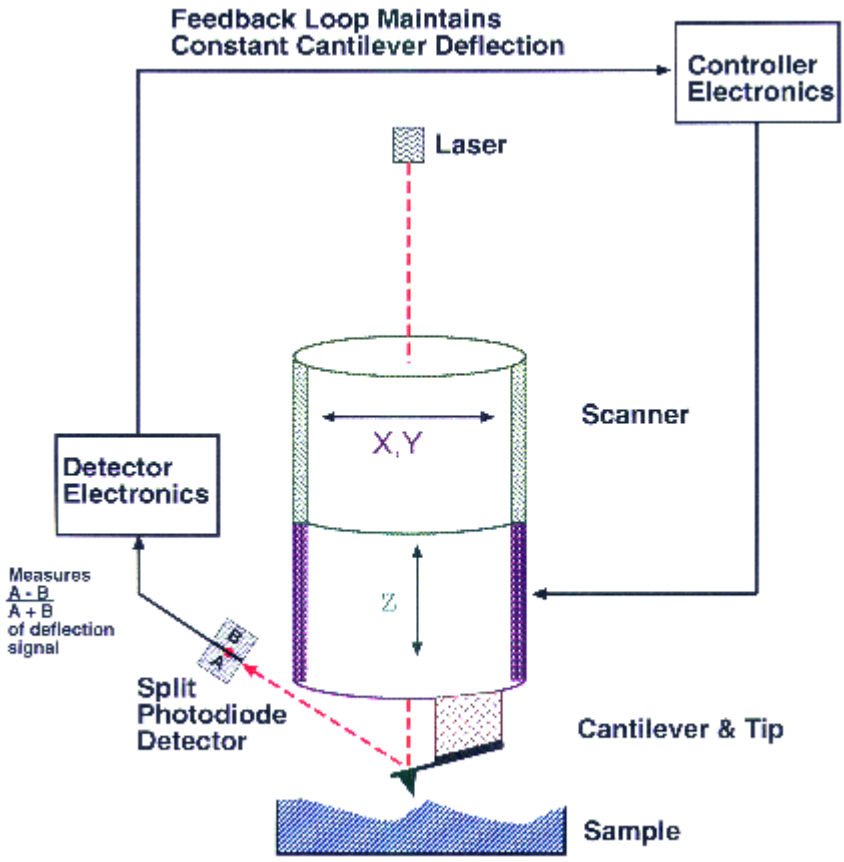

Figure 4.1: Schematic diagram showing the operating principles of the AFM in the contact mode (courtesy of digital instruments, Santa Barbara, CA)

The principal behind the working of an AFM is shown in the above figure [40]. The sharp tip is first brought close to the sample surface, and then the scanner makes final adjustment in the tip. The sample distance is determined by the user. The tip is scanned across the sample under the

action of piezoelectric actuator, either by moving the tip or the sample relative to the other. A laser beam is aimed at the back of the cantilever-tip assembly which reflects off the cantilever surface to a split diode which detects the small cantilever deflections. A feedback loop is shown in the above figure which maintains constant tip-sample separation by moving the scanner in the z-direction to maintain the setpoint deflection. Without this, the tip crashes into a sample for even small topographic features. The force between the tip and the sample is calculated by using Hooke's law and also by maintaining a constant tip-sample separation. The distance the scanner moves in the z direction is stored in the computer relative to spatial variation in the x-y plane to generate the topographic images of the sample surface.

Length, orientation and density determination algorithm:

1) Segmentation

Segmentation is a process that partitions the given image into some meaningful regions that are easy to analyze. Mostly segmentation is aimed at segmenting the image into foreground and the background. By doing this, the objects are easily separated from the background. There are three methods to achieve image segmentation. They are intensity based segmentation, edge based segmentation and region segmentation. Among these, intensity based segmentation performs better because the nanotube AFM image exhibits unimodal distribution which is easily handled only by the thresholding methods [10]. We have studied many thresholding methods for our nanotube image in the Chapter 1. The nanotube image that we are considering is very challenging and peculiar because of the rough substrate and unusual noise. Therefore, if the threshold value is low, the binary image picks up lot of noise and if the threshold value is high, nanotubes are broken causing more blobs than the number of actual fragments. To tackle this

scenario, we follow a different strategy of finding the threshold value at which the number of resulting blobs does not change with increasing threshold [4].

The number of blobs is a better parameter to use for thresholding compared to area and perimeter because breaks in the nanotubes at higher thresholds cause a more noticeable change in the number of blobs. We have computed the number of blobs for all the threshold values from 1 to 255 on the image (figure 4.3). At lower thresholds the number of blobs remained constant. At little higher thresholds the number of blobs rapidly increased and remained constant at some points. Further increase in the threshold values resulted in the decrease in the number of blobs. So, we have selected a threshold value above 230 where derivative is equal to zero. The thresholded image (figure 4.4) obtained for the AFM image of the nanotubes (figure 4.2) are shown below.

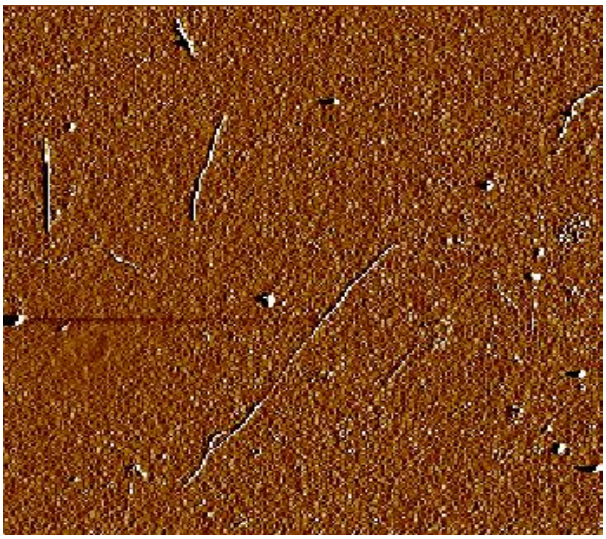

Figure 4.2: AFM image of SWNT on gold substrate

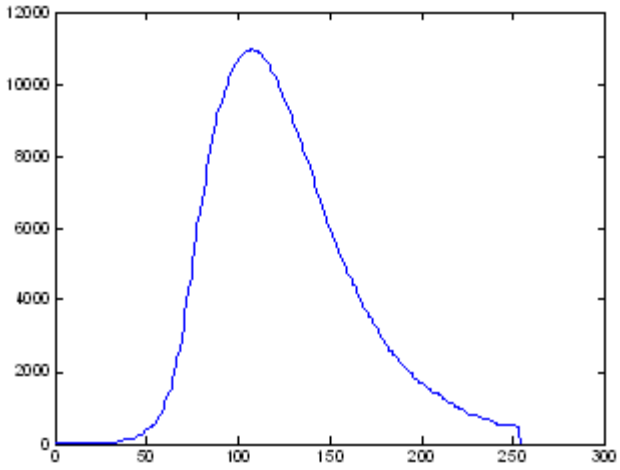

Figure 4.3: Bar chart showing X-axis- threshold values versus Y-axis-number of blobs

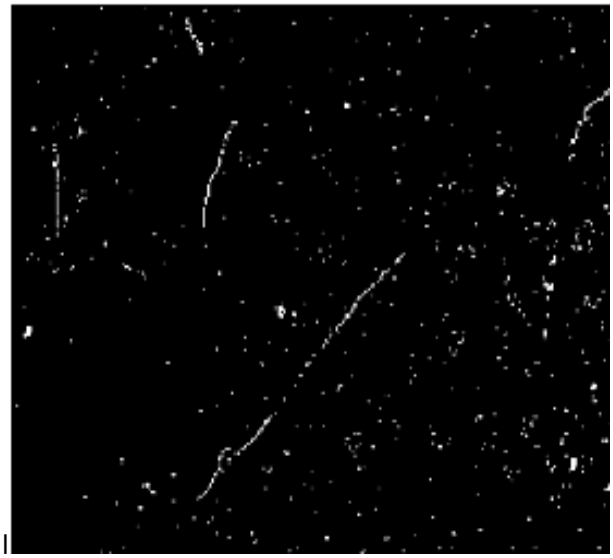

Figure 4.4: Binary image obtained after thresholding

2) Dilation:

Dilation is one of the two basic operators in the mathematical morphology. This is basically applied to binary image to enlarge the regions of boundaries of foreground pixels. This technique is used to fill in the small gaps thus growing the foreground region. We use this technique to join the broken parts of the nanotube.

This technique takes two input data. One is the image that needs to be dilated. The second is the structuring element usually called as kernel which determines the precise effect of the input image. We represent the foreground pixels by 1's and the background pixels by 0's. We also considered the 3x3 structuring element with the origin at the center and it is represented by

1	1	1
1	1	1
1	1	1

To compute the dilation, we consider each of the background pixels in the input image. Then we superimpose the kernel on top of the background pixel so that the origin of the kernel coincides with the background pixel position. The surrounding neighbors of the background pixels are checked to see whether there is at least one pixel in the kernel that coincides with the foreground pixel in the image underneath. If it is so, then the background pixel is set to the foreground value and if it is not so, then the background pixel is left at the background value.

Dilation of the object *A* by the kernel *B* is given by

$$A \oplus B = \{x : \hat{B}_x \cap A \neq \emptyset\}.$$

The resultant set is made up of all points generated by obtaining the reflection of *B* about its origin and then shifting this reflection by *x*. The image obtained after dilation process is shown below.

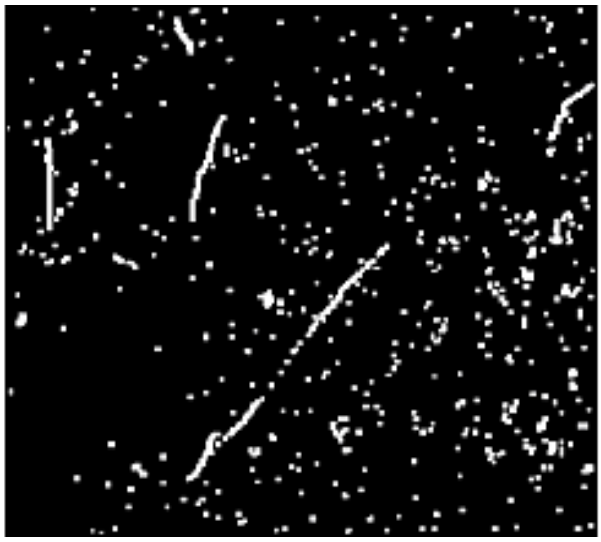

Figure 4.5: Dilated image obtained after dilation

3) Connected Component Labeling:

In order to obtain the blobs, we use Connected Component Labeling which scans an image and groups the pixels into components based on 4 pixels or 8 pixel connectivity. The criteria behind grouping these pixels into components are the similar pixel intensity values. Connected component labeling works on binary or gray level images and is based on different measures of connectivity. However, we assume thresholded input image and 8-connectivity. The connected component labeling scans the image by moving along a row until it comes to a point P for which V is a white pixel [41]. When this is true, it examines the four neighbors of P which have already been encountered in the scan (*i.e.* the neighbors (i) to the left of P, (ii) to the top of P, and (iii and iv) the two upper diagonal terms). Based on this information, the labeling of P occurs as follows [41]:

- If all four neighbors are 0, assign a new label to P, else
- if only one neighbor has a white pixel, assign its label to P, else

- if more than one of the neighbors have white pixels, assign one of the labels to P and make a note of the equivalences.

After completing the scan, the equivalent label pairs are sorted into equivalence classes and a unique label is assigned to each class. As a final step, a second scan is made through the image, during which each label is replaced by the label assigned to its equivalence classes. Once the blobs have been obtained by the connected component labeling, area of each blob have to be found. The next step we have done is the removal of unwanted blobs (noise) from the nanotube blobs. With the factor that the nanotube blobs are larger than the noise blobs, a minimum value have been set for the nanotube blobs. All the blobs less than that minimum value have to be converted to the background pixel. The image obtained, after setting a minimum value of 100 pixels is shown below.

Figure 4.6: Image obtained after connected component labeling

Finding the length:

The blob is fitted in an ellipse and the scalar specifying the length of the major axis of the ellipse that has the same normalized second central moments as the region, is the length of the nanotube which is represented in pixels.

The equation for ellipse is $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$

The length of the major axis is 2a.

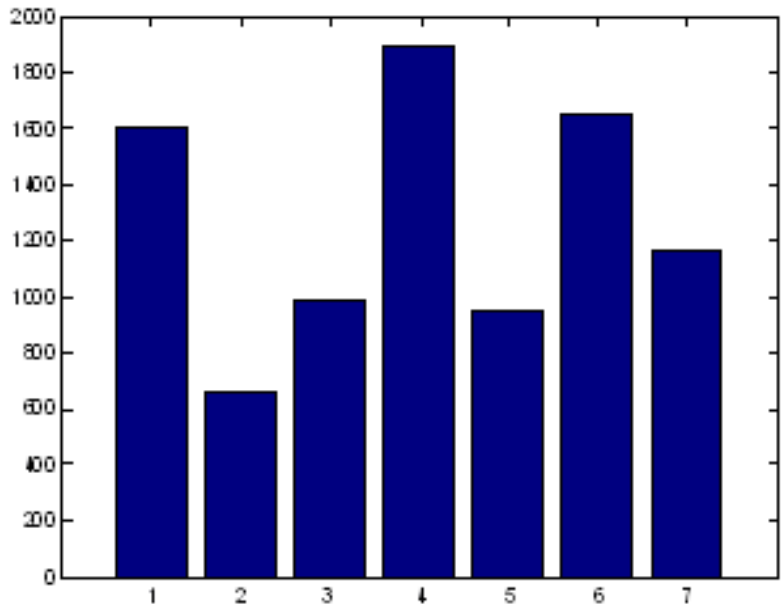

Figure 4.7: Bar chart showing X-axis- Number of nanotubes versus Y-axis- Length in nanometers

Finding the orientation:

It is the angle between the *x*-axis and the major axis of the ellipse that has the same second-moments as the region ranges from -90 to 90 degrees. Considering the second order moments,

$$a = \sum_{i=1}^{n}\sum_{j=1}^{m}(j-\bar{x})^2 B(i,j)$$

$$b = \sum_{i=1}^{n}\sum_{j=1}^{m}(j-\bar{x})\ (i-\bar{y})B(i,j)$$

$$c = \sum_{i=1}^{n}\sum_{j=1}^{m}\ (i-\bar{y})^2 B(i,j)$$

The orientation is given by

$$\tan 2\theta = \frac{b}{a.c}$$

$$\theta = \frac{1}{2}\tan^{-1}(\frac{b}{ac})$$

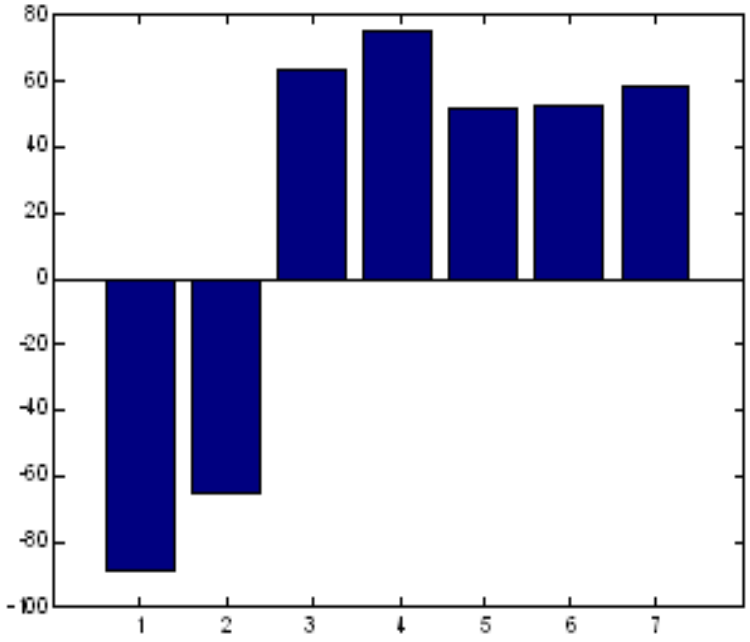

Figure 4.8: Bar chart showing X-axis- Number of nanotubes versus Y-axis- Angle between -90 to +90 degrees

4) Thinning:

In order to find the density of the nanotubes, we obtained the medial axis of the binary image by a process called thinning. Like the dilation process, thinning uses certain kernel which determines the thinning effect in the image. We made use of the following kernels and all the 90º rotations of it for this thinning process. At each iteration, the image is first thinned by the left kernel and then by the right kernel and then by the remaining 90º of the two kernel until none of the thinning produces any further change.

0	0	0
	1	
1	1	1

	0	0
1	1	0
	1	

By doing this, we determine the octagonal skeleton of the binary shape, the set of points that lie at the centers of octagons that fit entirely inside the shape and also which touches the boundary of the shape by at least two points. The skeletons produced by this method contain short spurs which is one of the minor shortcomings. These spurs need to be removed which requires one additional step called pruning. The image obtained after applying thinning process is shown below.

Figure 4.9: Image after thinning

5) Pruning:

Pruning is another sort of thinning which helps in removing the spurs. This uses different set of kernels like

0	0	0
0	1	0
0		

0	0	0
0	1	0
		0

At iteration, each element is used in each of its four *90°* rotations. We carried out pruning for only a limited number of iterations to remove short spurs, since pruning until convergence will actually remove all pixels except those that form closed loops. The image obtained after applying pruning process is shown below.

Figure 4.10: Image after pruning

Finding the density:

After pruning, the number of white pixels are calculated which denotes the density of the nanotubes on the substrate. We have calculated the ratio of the white pixels and black pixels and plotted a graph showing the density of nanotubes and the substrate respectively.

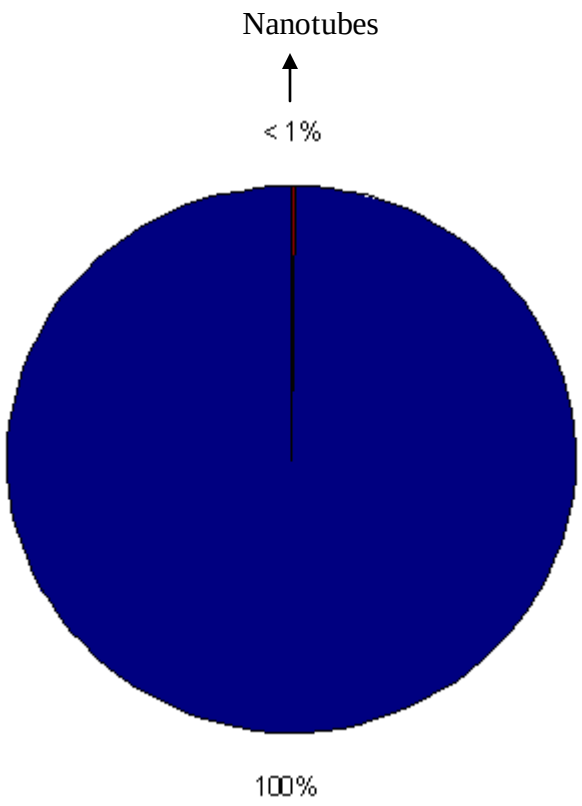

Figure 4.11: Pie chart showing the density of nanotubes

Testing:

The obtained results have been tested by using software called Pico-Scan, which is generally used by researchers to measure the parameters of each nanotube manually. We have adopted this software to verify the length, orientation and density of each nanotube obtained by our method.

CHAPTER 4B

DIAMETER DETERMINATION OF THE NANOPARTICLES USING HOUGH TRANSFORM

Hough transform:

Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing. The purpose of this technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. The classical Hough transform was concerned with the identification of lines in the image, but later the Hough transform has been extended to identify positions of arbitrary shapes, most commonly circles or ellipses.

The Hough transform is used to determine the parameters of the circle when a number of points that fall on the perimeter are known. A circle with radius *R* and center (*a*,*b*) is described in the parametric equation as

$$x = a + R\cos(\theta)$$

$$y = b + R\sin(\theta)$$

When the angle θ sweeps through the complete 360º, the points (*x*,*y*) trace the complete perimeter of the circle. If an image contains many points, some of which fall on the perimeter of the circles, then the parameters *a*, *b*, *R* are found to describe each circle.

The parameter space is 3D in this case. If the radius of the circles *R* in the image is known, then the search is reduced to 2D finding only the parameters (*a*,*b*). The locus of (*a*,*b*) points in the parameter space that fall on a circle of radius R centered at (*x*,*y*). The true center point is common to all parameter circles and is found using a Hough accumulation array. Multiple circles with the same radius are also found with the same technique. The center points are represented as red cells in the parameter space drawing. Overlap of circles causes spurious centers to also be found like the blue cell. Spurious circles are removed by matching to circles in the original image [42].

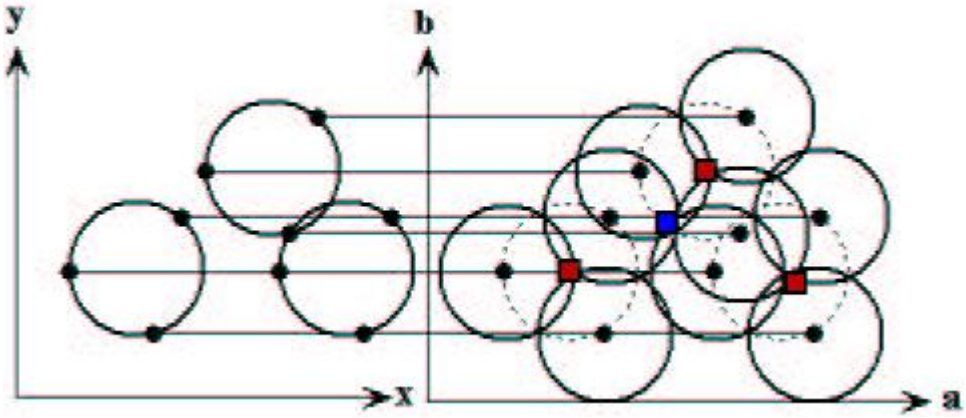

Figure 4.12: Each point in geometric space (left) generates a circle in parameter space (right).

The circles in parameter space intersect at the (*a*, *b*) that is the center in geometric space [42]. If the radius is unknown, then the locus of points in the parameter space falls on the surface of the cone. Each point on the perimeter of the circle produces a cone surface in parameter space. The parameters *a*, *b*, *R* corresponds to the accumulation cell where the largest numbers of cone surfaces interact.

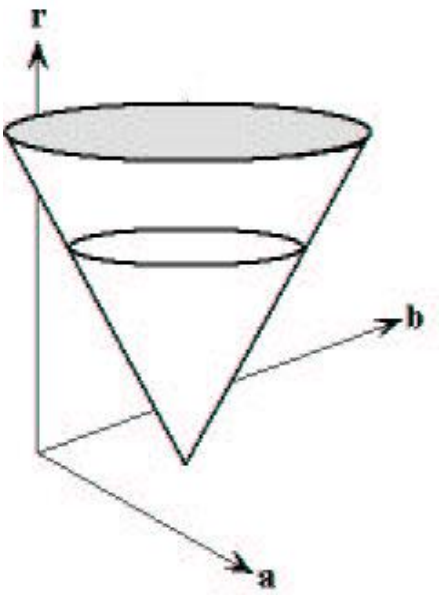

Figure 4.13: Generation of the conical surface in the parameter space for one (*x*,*y*) point

The above figure illustrates the generation of the conical surface in the parameter space for one (*x*,*y*) point. A circle with different radius is constructed at each level of *r*. The search for circles with unknown radius is conducted by using a three dimensional accumulation matrix.

Diameter determination algorithm:

The algorithm for Circular Hough Transformation can be summarized as [43]:

1. Find edges.

2. Begin the Hough transform [44].

3. For each edge point,

   Draw a circle with center in the edge point with radius *r* and increment all coordinates that the perimeter of the circle passes through in the accumulator.

4. Find one or several maxima in the accumulator.

5. End the Hough transform.

6. Map the found parameters (*r*,*a*,*b*) corresponding to the maxima back to the original image.

Results:

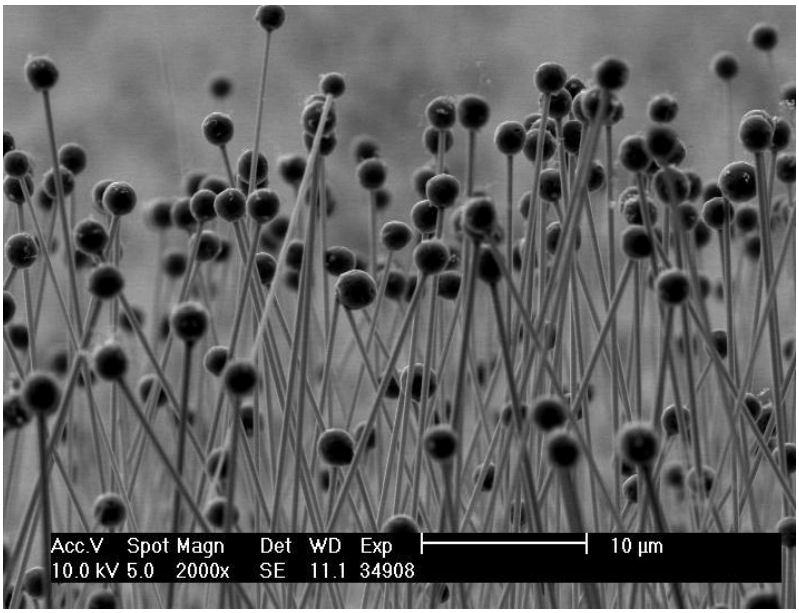

Figure 4.14: SEM image of the nanowires

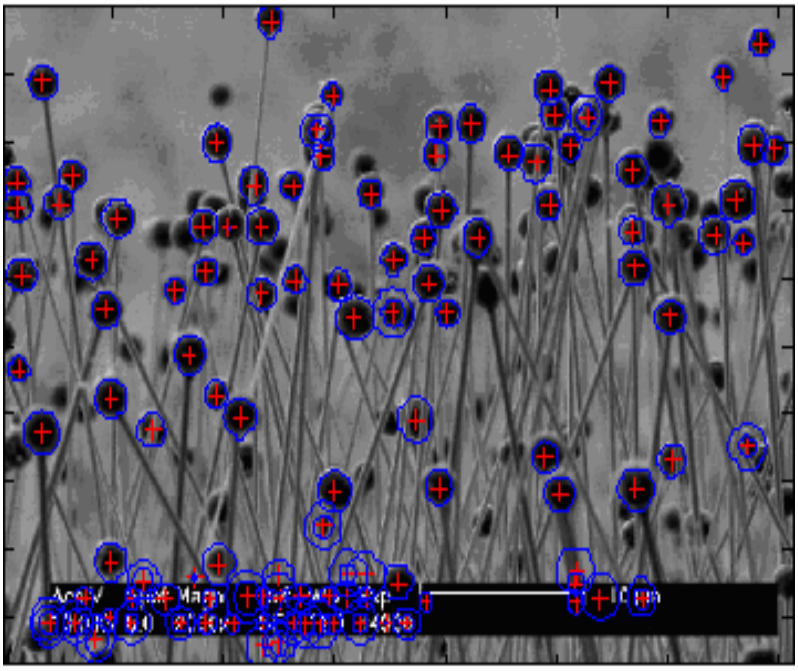

Figure 4.15: Circle detected with center positions and radii marked

CHAPTER 5

CONCLUSIONS AND FUTURE DIRECTION

The AFM and SEM nanostructure images have been segmented using various statistical methods like EM, kmeans and minimum distance method. The segmentation processes works fine for the AFM and SEM images with less noise. The statistical techniques that we have studied for thresholding seem to work better than standard automatic thresholding methods like Otsu, Ridler and Calvard method.

For images with dense nanotubes (figure 5.1) and more noise, it becomes extremely difficult to segment and extract the nanotubes. Dealing with denser nanotube images is a challenging task where segmentation and the extraction of the nanotubes remain promising. The image with overlapping nanotubes is shown below.

Figure 5.1: AFM image of SWNT on gold substrate

The segmentation techniques can also be applied to the SEM image (zinc oxide nanowire with germanium nanoparticle) shown in figure 5.2, to extract information such as length of the nanowire, diameter of the nanoparticle to find the total surface area. Also, finding the degree of blurriness of the nanowires would help in determining the relative distance between the nanowires in different zones.

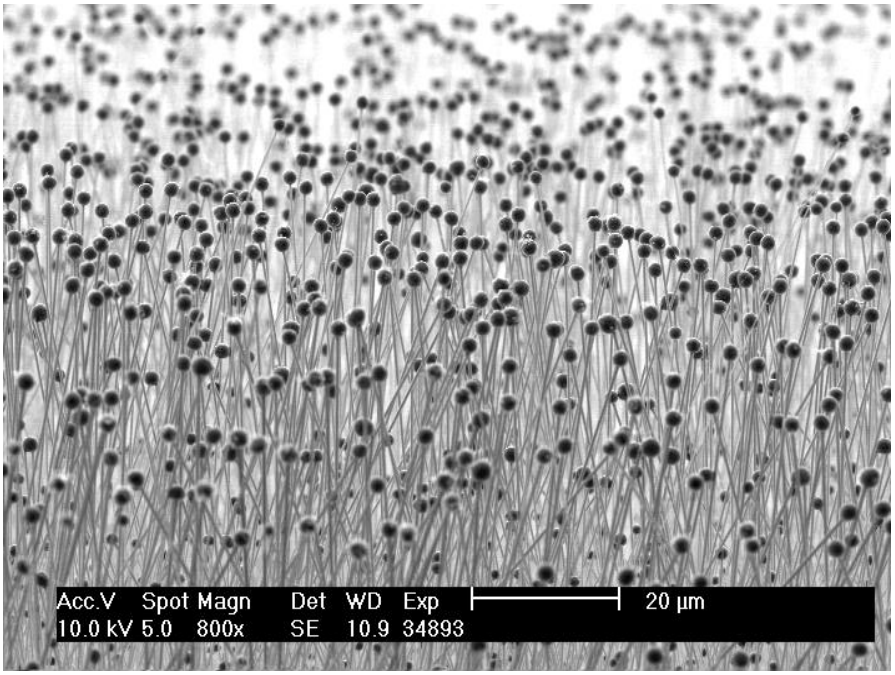

Figure 5.2: SEM image of zinc oxide nanowires with germanium nanoparticle on top

REFERENCES

1. M.F.M. Costa, "Application of image processing to the characterization of nanostructures," Reviews on Advanced Materials Science, vol. 6, no. 1, 2004, pp. 12-20.
2. A. P. Chuklanov, S. A. Ziganshina and A. A. Bukharaev, "Computer program for the grain analysis of AFM images of nanoparticles placed on a rough surface," Surface and Interface Analysis, vol. 38, no. 4, 2006, pp. 679-681.
3. E. Ficarra, L. Benini, E. Macii and G. Zuccheri, "Automated DNA fragments recognition and sizing through AFM image processing," IEEE Transactions on Information Technology in Biomedicine, vol. 9, no. 4, 2005, pp. 508-517.
4. T. S. Spisz, Y. Fang, R. H. Reeves, C. K. Seymour, I. N. Bankman and J. H. Hoh, "Automated sizing of DNA fragments in atomic force microscope images," Medical & Biological Engineering & Computing, vol. 36, no. 6, 1998, pp. 667-672.
5. G. Poletti, F. Orsini, C. Lenardi and E. Barborini, "A comparative study between AFM and SEM imaging on human scalp hair," Journal of Microscopy-Oxford, vol. 211, no. 7, 2003, pp. 249-255.
6. Y. C. Cheng and S. Y. Chen, "Image classification using color, texture and regions," Image and Vision Computing, vol. 21, no. 9, 2003, pp. 759-776.
7. H. Zhang, J. E. Fritts and S. A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," Computer Vision and Image Understanding, vol. 110, no. 2, 2008, pp. 260-280.

8. R. Nagarajan, “Intensity-based segmentation of microarray images,” IEEE Transactions on Medical Imaging, vol. 22, no. 7, 2003, pp. 882-889;

9. M. A. Wani and B. G. Batchelor, “Edge-region-based segmentation of range images,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 3, 1994, pp. 314-319.

10. P. L. Rosin,“Unimodal thresholding,” Pattern Recognition, vol. 34, no. 11, 2001, pp. 2083-2096.

11. P. K. Sahoo, S. Soltani, A. K. C. Wong and Y. C. Chen, “A survey of thresholding techniques,” Computer Vision Graphics and Image Processing, vol. 41, no. 2, 1988, pp. 233-260.

12. Y. Bazi, L. Bruzzone and F. Melgani, “Image thresholding based on the EM algorithm and the Generalized Gaussian distribution,” Pattern Recognition, vol. 40, no. 2, 2007, pp. 619-634.

13. F. H. Y. Chan, F. K. Lam and H. Zhu, “Adaptive thresholding by variational method,” IEEE Transactions on Image Processing, vol. 7, no. 3, 1998, pp. 468-473.

14. V. Anandan, Y. L. Rao and G. G. Zhang, “Nanopillar array structures for enhancing biosensing performance,” International Journal of Nanomedicine, vol. 1, no. 1, 2006, pp. 73-79.

15. C. H. Li and C. K. Lee, “Minimum cross entropy thresholding,” Pattern Recognition, vol. 26, no. 4, 1993, pp. 617-625.

16. N. Otsu, “Threshold selection method from gray-level histograms,” IEEE Transactions on Systems Man and Cybernetics, vol. 9, no. 1, 1979, pp. 62-66.

17. T. W. Ridler and S. Calvard, “Picture thresholding using an iterative selection method,” IEEE Transactions on Systems Man and Cybernetics, vol. 8, no. 8, 1978, pp. 630-632.

18. T. Sund and K. Eilertsen, “An algorithm for fast adaptive image binarization with applications in radiotherapy imaging,” IEEE Transactions on Medical Imaging, vol. 22, no. 1, 2003, pp. 22-28.

19. J. Kittler and J. Illingworth, “Minimum error thresholding,” Pattern Recognition, vol. 19, no. 1, 1986, pp. 41-47.

20. H. J. Trussell, “Picture thresholding using an iterative selection method - Comments,” IEEE Transactions on Systems Man and Cybernetics, vol. 9, no. 5, 1979, pp. 311-314.

21. A. N. G. Parra-Vasquez, I. Stepanek, V. A. Davis, V. C. Moore, E. H. Haroz, J. Shaver, R. H. Hauge, R. E. Smalley and M. Pasquali, “Simple length determination of single-walled carbon nanotubes by viscosity measurements in dilute suspensions,” Macromolecules, vol. 40, no. 11, 2007, pp. 4043-4047.

22. M. D. Lay, J. P. Novak and E. S. Snow, “Simple route to large-scale ordered arrays of liquid-deposited carbon nanotubes,” Nano Letters, vol. 4, no. 4, 2004, pp. 603-606.

23. S. D. Zheng, W. Pan, C. M. Rouleau, and D. H. Lowndes, “Germanium-Catalysed Growth of Zinc Oxide Nanowires: A semiconductor Catalyst for Nanowire Synthesis,” Angewandte Chemie International Edition, 2005, pp. 274-278.

24. Maximum Likelihood,

URL: http://www.weibull.com/AccelTestWeb/mle_maximum_likelihood_parameter_estimation.htm.

25. W. H. Tsai, “Moment-preserving thresholding - A new approach,” Computer Vision Graphics and Image Processing, vol. 29, no. 3, 1985, pp. 377-393.

26. J. N. Kapur, P. K. Sahoo and A. K. C. Wong, “A new method for gray-level picture thresholding using the entropy of the histogram,” Computer Vision Graphics and Image Processing, vol. 29, no. 3, 1985, pp. 273-285.

27. A. K. C. Wong and P. K. Sahoo, “A gray-level threshold selection method based on maximum-entropy principle,” IEEE Transactions on Systems Man and Cybernetics, vol. 19, no. 4, 1989, pp. 866-871.

28. A. S. Abutaleb, “Automatic thresholding of gray-level pictures using two-dimensional entropy,” Computer Vision Graphics and Image Processing, vol. 47, no. 1, 1989, pp. 22-32.

29. A. Brink, “Maximum entropy segmentation based on the autocorrelation function of the image histogram,” Journal of Computing and Information Technology, vol. 2, no. 2, 1994, pp. 77 - 85

30. T. Kurita, N. Otsu and N. Abdelmalek, “Maximum-likelihood thresholding based on population mixture-models,” Pattern Recognition, vol. 25, no. 10, 1992, pp. 1231-1240.

31. H. D. Cheng, Y. H. Chen and X. H. Jiang, “Thresholding using two-dimensional histogram and fuzzy entropy principle,” IEEE Transactions on Image Processing, vol. 9, no. 4, 2000, pp. 732-735.

32. L. K. Huang and M. J. J. Wang, “Image thresholding by minimizing the measures of fuzziness,” Pattern Recognition, vol. 28, no. 1, 1995, pp. 41-51.

33. K. Oshida, T. Nakazawa, T. Miyazaki and M. Endo, “Application of image processing techniques for analysis of nano- and micro-spaces in carbon materials,” Synthetic Metals, vol. 125, no. 2, 2001, pp. 223-230.

34. P. Belhomme, D. Houivet, W. Lecluse and J.M. Haussonne,, "Image analysis of multi phased ceramics," Journal of the European Ceramic Society, vol. 21, no. 10, 2001, pp. 2149-2151.

35. M. Ziabari, V. Mottaghitalab, S. T. McGovern and A. K. Haghi, "A new image analysis based method for measuring electrospun nanofiber diameter," Nanoscale Research Letters, vol. 2, no. 12, 2007, pp. 597-600.

36. T. S. Korting, L. V. Dutra, L. M. G. Fonseca, G. Erthal, F. C. D. Silva "Improvements to Expectation-Maximization approach for unsupervised classification of remote sensing data," IX Brazilian Symposium on GeoInformatics, 2007, pp. 3-11.

37. M. A. T. Figueiredo, "Lecture Notes on the EM Algorithm," URL: http://www.stat.duke.edu/courses/Spring06/sta376/Support/EM/EM.Mixtures. Figueiredo.2004.pdf, May 19, 2004.

38. Segmentation, URL: http://www.bioss.ac.uk/staff/chris/ch4.pdf.

39. D. W. Scott, "Parametric statistical modeling by minimum integrated square error," Technometrics, vol. 43, no. 3, 2001, pp. 274-285.

40. C. R. Blanchard and J. B. Campbell, "Atomic-force microscopy," Advanced Materials & Processes, vol. 148, no. 2, 1995, pp. 62-66.

41. Connected Component Labeling tutorial, URL: http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm.

42. H. Rhody, "Hough Circle Transform," Rochester Institute of Technology-Lecture notes, 2005. URL: http://www.cis.rit.edu/class/simg782/lectures/lecture_10/lec782_05_10.pdf.

43. S. J. K. Pedersen, "Circular Hough Transform," Aalborg University, 2007. URL:http://www.cvmt.dk/education/teaching/e07/MED3/IP/Simon_Pedersen_Circular

HoughTransform.pdf.

44. T. Peng, A. Balijepalli, S. K. Gupta and T. LeBrun, "Algorithms for on-line monitoring of micro spheres in an optical tweezers-based assembly cell," Journal of Computing and Information Science in Engineering, vol. 7, no. 4, 2007, pp. 330-338.

45. A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society Series B, vol. 39, no. 1, Nov. 1977, pp. 1-38.

46 T. Moon, "The Expectation-Maximization algorithm," IEEE Signal Processing Magazine, Nov. 1996, pp. 47-60.