Mixture Model Distributions

in Image Processing

PROJECT REPORT

(July, 2015 - June, 2016)

SUBMITTED BY

12102239
ANUSHREE BHATT

12102306
KARTIK.K

In partial fulfilment for the award of the degree
Of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Under the guidance of
DR. ABHINAV GUPTA (Assistant Professor)
ECE Department

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY

NOIDA – 201 307

TALBLE OF CONTENTS

CERTIFICATE

This is to certify that the work titled "**Mixture Model Distributions in Image Processing"** submitted by "**Anushree Bhatt**" and "**Kartik.K**" in partial fulfilment for the award of degree of B. Tech in Electronics and Communication Engineering of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor: ………………………

Name of Supervisor: Dr. Abhinav Gupta

Designation: Assistant Professor

Date:

ACKNOWLEDGEMENT

We consider it our profound privilege and express deep sense of gratitude to our project supervisor DR. ABHINAV GUPTA without whose proficient guidance, keen and enthusiastic interest it would have been impossible for us to develop the idea of this project and execute it successfully.

We would like to extend our gratitude to the E.C.E department of Jaypee Institute of Information Technology for all the help and resources made available to us and also to the people directly and indirectly related in the completion of the project.

We also want to thank our parents for their support and assistance throughout the course of the development of the project.

Signature of the student	....	....
Name of Student	Anushree Bhatt	Kartik .K

Date ……………………

SUMMARY

This project is divided into two parts. In part 1 of the project, the basics of image segmentation through various techniques mainly focussing on Gaussian Mixture Model are covered. Different algorithms are applied that are required to detect the objects in an image in stages that are Expectation Maximization algorithm, Morphological algorithm, Blob Detection technique. The parameters of Gaussian Mixture model are obtained through the Expectation Maximization algorithm. Further, noise is removed from the binary image obtained and then blob is applied on the detected object that is the vehicle on road in order to count it. The application of this study is presented through a MATLAB code.

An Automatic vehicle counting system makes use of video data acquired from stationary traffic cameras, performing causal mathematical operations over a set of frames obtained from the video to estimate the number of vehicles present in a scene. It is just the ability of automatically extract and recognize the traffic data e.g. total number of vehicles, vehicle number and label from a video. Counting vehicles gives us the information needed to obtain a basic understanding over the flow of traffic in any region under surveillance. [1]

In part 2, a method is studied to estimate the parameters of Rayleigh Mixture Model (RMM) in order to characterize vulnerable plaques in arteries using an Intravascular Ultrasound image data. In order to find the parameters of RMM an Expectation Maximization (EM) algorithm is studied and implemented. Various other methods have been implemented such as feature extraction using a Grey Level Co-occurrence (GLCM) matrix, K-Nearest Neighbour (KNN) classification for classifying the types of plaques and morphological operations for identifying plaque regions.

Characterization of plaque is really important as it blocks the normal flow of blood in arteries. This leads to serious problems like heart attack or stroke. With the help of Intravascular Ultrasound (IVUS) data it is easier to determine the type of plaque that exists and where exactly it is located so that it can be treated. The existing features of the three types of plaques are known. When a plaque is located and identified using KNN and morphology its parameters are extracted using RMM by applying EM algorithm. These parameters are then compared with the given values of the three types of plaques, hence, identifying the kind of plaque that is present in that particular IVUS image data. [6]

LIST OF FIGURES

1. INTRODUCTION

Vehicle detection and tracking plays an effective and significant role in the area of traffic surveillance system where efficient traffic management and safety is the main concern. In this project we have studied and applied a method of detecting vehicles from video frames. First, we differentiate the foreground from background in frames by learning the background. Here, foreground detector detects the object and a binary computation is done to define rectangular regions around every detected object. To detect the moving object correctly and to remove the noise some morphological operations have been applied. Then the final counting is done by tracking the detected objects and their regions [1].

Consider this automatic vehicle counting system that makes use of video data acquired from stationary traffic cameras using a Gaussian Mixture Model (GMM) based on the principles of image segmentation, it is understood that GMM based segmentation has been highly successful and the most common approach when it comes to developing such systems. It also leads to a few questions. [1]

1.1 Use of Gaussian distribution in a mixture model

Suppose there is a dataset, let's say there is a need to calculate the Mahalanobis distance criteria or covariance matrix or Bayes paradigm, assuming that the distribution or scattering of data in whatever dimension is a Gaussian distribution. It can be observed that this particular distribution comes out to be the one which is closest to the natural distribution. Also, it is easy to do mathematical manipulations if there is a Gaussian function involved. For instance, the differentiation exists of as much order as required. Furthermore, there may be situations where the distribution may not be strictly Gaussian in nature, yet the problem could be solved using multiple Gaussians. For instance, when there is a need to cluster the data into several components, it is assumed each cluster forms a Gaussian leading to the origin of GMM. Primarily GMM is used in order to achieve soft clustering (as an improvised alternate to K-means based hard clustering) to achieve image background subtraction. [1]

1.2 Use of RMM

Another question is that if GMM is so successful, then why can't it be applied to something like an ultrasound image? The reason being in ultrasound images, pixel intensity gets corrupted due to speckle multiplicative noise as a result GMM cannot model the details of this image accurately. Rayleigh distribution however, has the mathematical simplicity and along with the robust nature of mixture model, the so called RMM can be used to analyze the pixel and can accurately model in spite the speckle noise. So, mostly RMM has applications in medical and IVUS imaging. [6] [13]

2. BACKGROUND STUDY

2.1 Image Segmentation

Segmentation is the process of partitioning an image into regions where region is a group of connected pixels with similar properties like gray levels, colours, textures, motion characteristics (motion vectors) and edge continuity [2].

There are two approaches to segmentation: region segmentation, edge segmentation

One way of segmentation is through clustering.

2.1.1 Image Segmentation by Clustering Pixels

Clustering is a process whereby a data set is replaced by clusters, which are collections of data points that belong together. It is natural to think of image segmentation as clustering. Pixels may belong together because they have the same colour and/or they have the same texture and/or they are nearby.

Clustering and Segmentation by K-means clustering

A natural objective function can be obtained by assuming that we know there are k clusters, where k is known. Each cluster is assumed to have a center, we write the center of the i^{th}cluster as ci. The j^{th} element to be clustered is described by a feature vector $x_i$we are segmenting an intensity image, x might be the intensity at a pixel.

The following algorithm is iterated till the segments are achieved:

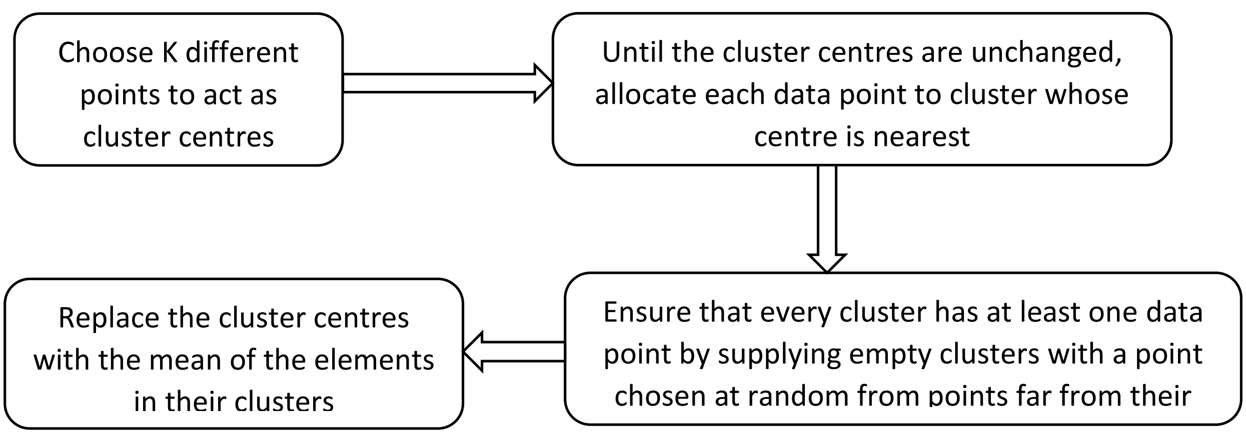


Figure 2.1 K-means algorithm

This is a form of hard clustering as a data point either belongs to a cluster or not.

There are soft clustering methods which are flexible such that they can assign a data point to more than one cluster. Gaussian Mixture Model (GMM) uses soft clustering. GMM clustering can accommodate clusters that have different sizes and correlation structures within them. Because of this, GMM clustering can be more appropriate to use than *k*-means clustering.

2.1.2 Image Segmentation Using GMM

Gaussian Mixture Model

Gaussian mixture models (GMM) are often used for data clustering. It is a function to measure parametric probability density represented as a weighted sum of Gaussian component densities [1].

Our aim is to separate foreground from background so that we can track vehicles in every frame.

Each image is a mixture of Gaussians. In order to find the Gaussian mixture for a given image Expectation Maximization (EM) Algorithm is applied.

Expectation Maximization Algorithm

Mixture models and EM algorithm are related with clustering methods. In GMM each cluster belongs to a probability distribution. . EM finds the probabilities of the pixels to be in a particular Gaussian. In other words through EM algorithm the aim is to find the parameters of probability distribution that are mean and covariance of each Gaussian. EM is broken down into two steps:

1. Expectation or estimation
2. Maximization

For understanding EM in 1-D see the following example.

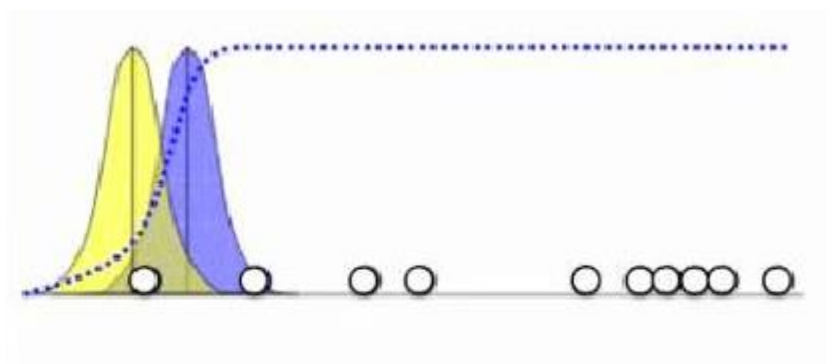


Figure 2.2 Considering a set of data points having unknown source. [9]

Let's say there are some data points each one represented as x_i where i=1, 2, 3…10 and it's not know which data point comes from which source or Gaussian it belongs to. Nor is the mean or variance of the Gaussians to which data points can be assigned to is known. Through EM say, two random Gaussians with random parameters Ga(yellow) (μ_a , σ_a^2) , Gb(blue) (μ_b, σ_b^2) (Refer Figure 2-1). Now, for each point x_i EM will try to figure out if with these current parameters the point x_i looks as if it came from Ga or Gb.

$$P(x_i|b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left[-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right] \qquad \text{……. (2.1)}$$

$$P(x_i|a) = \frac{1}{\sqrt{2\pi\sigma_a^2}} \exp\left[-\frac{(x_i - \mu_a)^2}{2\sigma_a^2}\right] \qquad \text{……. (2.2)}$$

It is going to assign these point Ga or Gb unlike K-means which is a hard assignment as it will compute the probability of each point of going to Ga or Gb and it will not quantize the probability to 0 or 1. Looking at the graph (Figure 2-1) the probability of the rightmost data point to lie in Ga is less but it is even less likely to lie in Gb. Now computing the Bayesian Posterior and taking both the probabilities of x_i lying in Ga and Gb.

$$Gb_i = P(b|x_i) = \frac{P(x_i|b)\,P(b)}{P(x_i|b)\,P(b) + P(x_i|a)\,P(a)} \qquad \text{……. (2.3)}$$

$$Ga_i = P(a|x_i) = \frac{P(x_i|a)\,P(a)}{P(x_i|b)\,P(b) + P(x_i|a)\,P(a)} \qquad \text{……. (2.4)}$$

or

$$Ga_i = 1 - Gb_i \qquad \text{……. (2.5)}$$

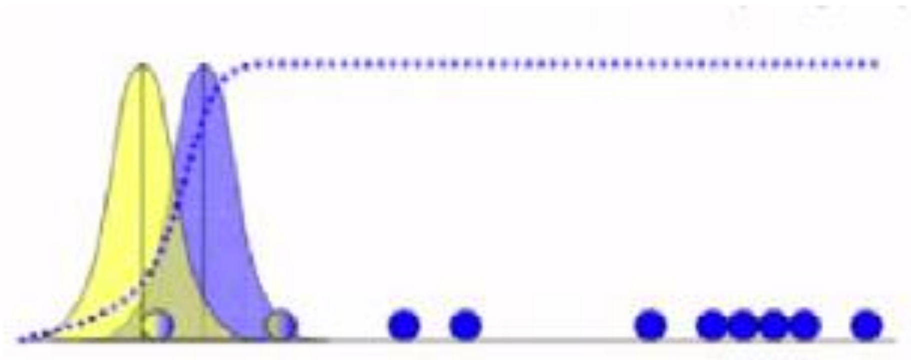


Fig. 2.3 Data points assigned Ga and Gb according to the posterior [9].

After these points are distributed to the respective Gaussians according to their probability (Refer Figure 2-2), a new mean and variance of each Gaussian is calculated using the value of these data points.

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \cdots + b_n x_n}{b_1 + b_1 + \cdots + b_n} \qquad \text{……. (2.6)}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \cdots + a_n x_n}{a_1 + a_1 + \cdots + a_n} \qquad \text{……. (2.7)}$$

Where, n stands for the number of points that went to the blue cluster.

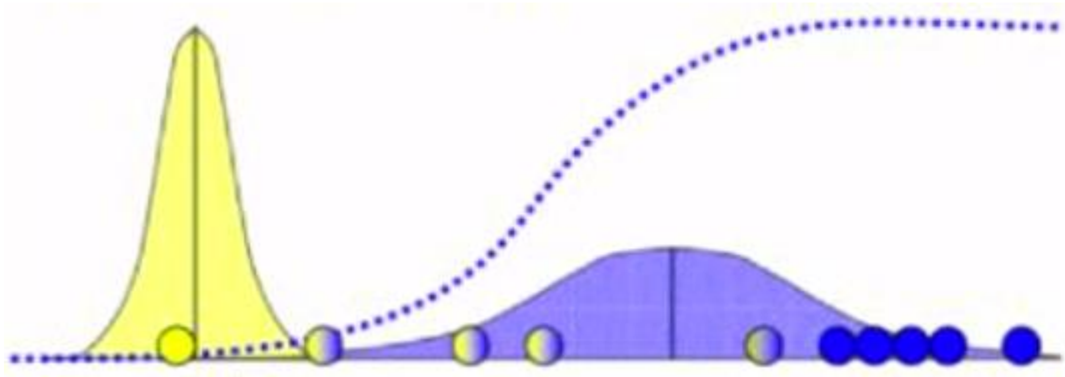


Figure 2.4 A new mean and variance calculated and data points arranged accordingly [9].

Here x_i is giving little bit of its mass to every centroid. Although it gives more mass to the centroid it is more associated with, just like the right most gives to blue and left most gives to yellow

Now computing the variance in the same way.

$$\sigma_b^2 = \frac{b_1 (x_1 - \mu_b)^2 + b_2 (x_2 - \mu_b)^2 + \cdots + b_n (x_n - \mu_b)^2}{b_1 + b_1 + \cdots + b_n} \quad \ldots\ldots. (2.8)$$

$$\sigma_b^2 = \frac{b_1 (x_1 - \mu_b)^2 + b_2 (x_2 - \mu_b)^2 + \cdots + b_n (x_n - \mu_b)^2}{b_1 + b_1 + \cdots + b_n} \quad \ldots\ldots. (2.9)$$

Here square differences are multiplied with the weights or confidences that these points are from Ga distribution or Gb (Equation 8 and 9). The position of Ga and Gb change according to the new centroids and the data points are again arranged (Refer Figure 2-3). This iteration goes on till convergence.

On observing, the mean of Gb will be in the middle of all the blue data points and it will have a high variance as it covers the maximum points. While yellow will remain narrow as only two points are covered by it. Variance will also be small.

Based on these two Gaussians a different posterior will be calculated, i.e. the blue dotted line, which will assign these data points the Gaussians they will lie in. With next iteration, Gb will move more towards the right while Ga will move more towards the left (Refer Figure 2-4).

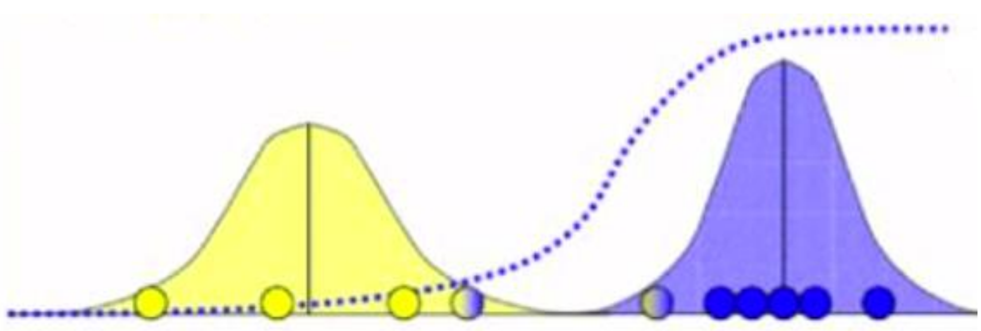


Figure 2.5 Convergence [9]

Priors are estimated using,

P (b) = $(b_1 + b_2 + \cdots + b_n)$/ n ……. (2.10)

P (a) = 1- P (b) ……. (2.11)

It tells what proportion of data that Gb / Ga distribution is describing.

Applying Gaussian Mixture Models to high dimensional data

There are vector valued random variables and d different attributes for data and it is assumed that every one of the data points come from one of k different sources and each source is a Gaussian in a d dimensional space [3][4].

Here three parameters to be found with respect to K Gaussian mixture models. Since value of the number of Gaussians to be taken is K, accordingly K means, K covariance matrices and K mixture coefficients or weights are determined.

For a multi-variate Gaussian distribution we have the following expression:

$$P(\vec{x}_i ; \mu, \Sigma) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp\{-\frac{1}{2}(\vec{x} - \mu)\, \Sigma^{-1}(\vec{x} - \mu)^T\} \quad \ldots\ldots (2.12)$$

Where μ = length d-row vector

Σ =d×d matrix

$|\Sigma|$= matrix determinant

The result will give us a scalar number which evaluates to the probability of a value x.

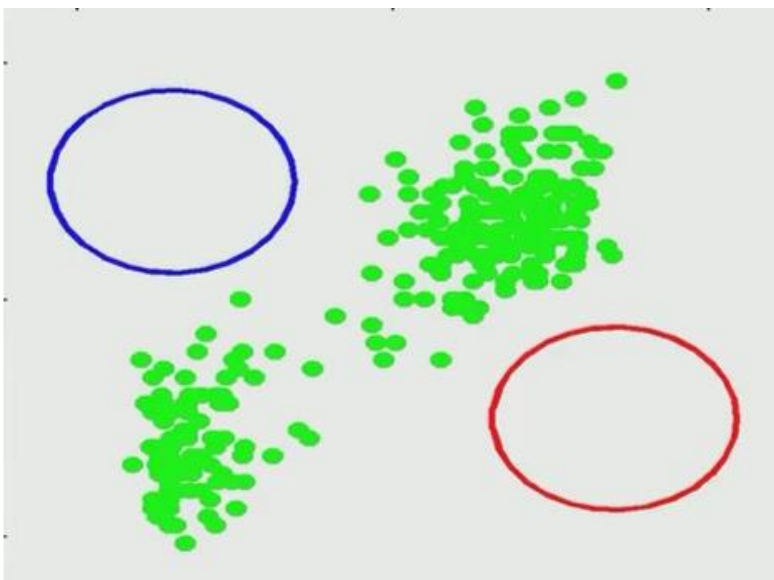


Figure 2.6 Placing two Gaussians randomly on a set of data points [10]

This case is proceed with the algorithm like the previous 1-d example. In 2-D we have a set of data points and let us say we place 2 Gaussians in that with random means and random covariance matrices somewhere in space (Refer Figure 2-5), let us take one red and one blue Gaussian over the data points. Starting from the first step we determine the probability of each data point x_i lying in each of the Gaussian. Finding how typical is x_i under source C

$$P(\vec{x}_i \mid c) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp \{-\frac{1}{2}(\vec{x} - \mu)\Sigma^{-1}(\vec{x} - \mu)^T\} \quad \text{……. (2.13)}$$

This covariance matrix is specifically for Gaussian C which determines the shape of the Gaussian and it is a multiple of the identity matrix since both the Gaussians taken are spherical .Otherwise in general it is a matrix that reflects how different attributes correlate with each other.

Now applying Bayes' rule we convert these probabilities to posteriors. Finding how likely x_i came from C

$$P(c \mid \vec{x}_i) = \frac{P(\vec{x}_i \mid c)\, P(c)}{\sum_{c=1}^{k} P(\vec{x}_i \mid c)\, P(c)} \quad \text{……. (2.14)}$$

Now, converting the posteriors into weights. These weights reflect how important a particular point is for a particular Gaussian. It's the weight that associates instant i with Gaussian number C. It's the posterior divided by the sum of all the posteriors. For example out of all the points that were coloured red how important is this particular point in relation to that.

$$w_{i,c} = \frac{P(c \mid \vec{x}_1)}{(P(c \mid \vec{x}_1) + P(c \mid \vec{x}_2) + \cdots + P(c \mid \vec{x}_n))} \quad \text{……. (2.15)}$$

Now using these weights to recompute the means and covariance of the two Gaussians:

So the mean for the Gaussian C, computing it individually for each attribute a and b

Mean of attribute a in items assigned to C:

$$\mu_{ca} = w_{c1}x_{1a} + w_{c2}x_{2a} + \cdots + w_{cn}x_{na} \quad \text{……. (2.16)}$$

Mean of attribute b in items assigned to C:

$$\mu_{ca} = w_{c1}x_{1a} + w_{c2}x_{2a} + \cdots + w_{cn}x_{na} \quad \text{……. (2.17)}$$

Finding how correlated attributes a and b are in source C, Covariance:

$$\Sigma_{cab} = \sum_{i=1}^{n} w_{ci}\,(x_{ia} - \mu_{ca})(x_{ib} - \mu_{cb}) \quad \text{……. (2.18)}$$

Finding out priors: how many items assigned to C:

$$P(c) = \frac{1}{n}\sum_{i=1}^{n} p(c \mid \vec{x}_i) \quad \text{……. (2.19}$$

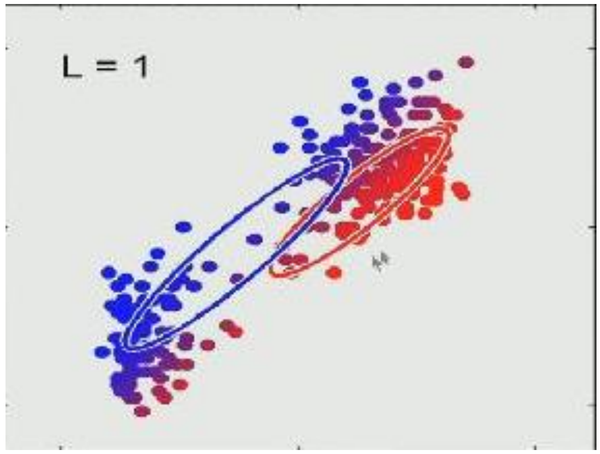


Figure 2.7 First iteration [10].

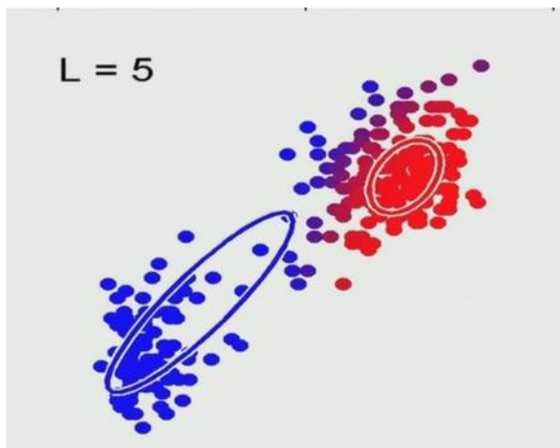


Figure 2.8 Fifth iteration [10].

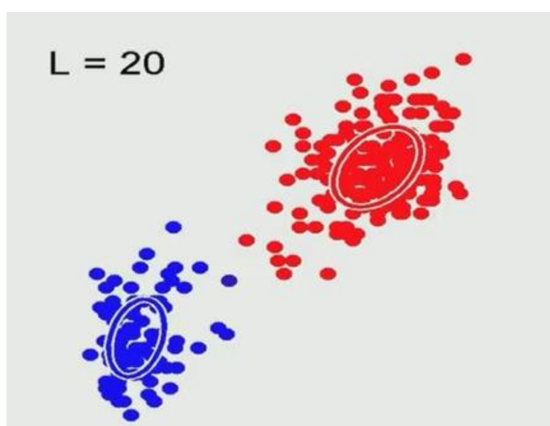


Figure 2.9 20th iteration (convergence) [10].

Convergence is obtained by the time 20th iteration is reached. This means that the parameter of the Gaussians will not be changing further.

2.2 Plaques

Plaque is formed when calcium, fat and other substances in blood combine with cholesterol. Plaque narrows down the arteries as it builds up and hardens hence blocking the arteries and restricting smooth flow of blood. This situation is called atherosclerosis. Vulnerable plaques are the major cause of carotid and coronary vascular problems, such as heart attack or stroke. A correct modelling of plaque echo morphology and composition can help the identification of such lesions. [6]

2.2.1 Types of Plaques

There are three different types of plaques:

Fibrotic- it is a pearly white area within an artery that causes the intimal surface to bulge into a lumen. It is composed of lipid, cell debris, and smooth muscle cells collagen, and, in older persons, calcium.

Lipidic– it is an irregular and eccentrically placed accumulation of fatty material in the walls of the coronary arteries.

Calcified– it happens due to calcification in coronary arteries. [6]

2.3 Intravascular Ultrasound

In medical ultrasound (US), a transmitted ultrasonic pulse interacts with an anatomical region providing information about internal tissue structures. The backscattered or received signal is corrupted by a characteristic granular pattern noise called speckle, which depends on the number of scatterers or reflectors and their size.

Intravascular US (IVUS) is an imaging technique that allows to clearly assess the arterial wall internal echo morphology. It is considered a suitable technique for in vivo characterization of the coronary plaques composition. [6]

2.3.1 Acquiring IVUS data

The technical procedure of acquiring IVUS data consists in introducing a catheter, it carries a rotating ultrasound emitter inside the vessel. During rotation, a piezoelectric transducer transmits ultrasound waves and collects the reflected components that are afterward converted into electrical signals and sampled by an analog-to-digital converter. The IVUS image obtained by processing the received echoes is a 360◦ topographic view of the inner arterial walls. [6] [13]

2.3.2 Speckle

This project intends to model the atherosclerotic plaque through the analysis of the backscattered IVUS data. A scanned tissue sample suffers from a certain number of scattering phenomena known as fully developed speckle. It is a tissue or region composed of a large number of scatterers, acting as echo reflectors.

These scatterers arise from in homogeneity and structures approximately equal to or smaller in size than the wavelength of the ultrasound. It is recognized that under fully developed speckle, pixel intensities in envelope images are well modelled by Rayleigh probability density functions (PDFs). [6]

2.4 Rayleigh Mixture Model

The Rayleigh distribution is widely used to describe (nearly) homogeneous areas in ultrasound images. A mixture of Rayleigh distributions is known as the Rayleigh Mixture Model (RMM). This project involves estimation of the RMM mixture parameters by means of the Expectation Maximization Algorithm (EM), which aims at characterizing tissue echo morphology in ultrasound. The performance of the proposed model is evaluated with a database of in vitro intravascular ultrasound cases. It is shown that the mixture coefficients and Rayleigh parameters explicitly derived from the mixture model are able to accurately describe different plaque types and to significantly improve the characterization performance of an already existing methodology. [6] [7]

2.5 EM Algorithm for RMM

An expectation–maximization (EM) algorithm is a method for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters, where the model depends on hidden variables. The EM iteration involves an expectation (E) step, which creates a function for the expectation of the log-likelihood found using the current estimate for the parameters, and a maximization (M) step, which calculates parameters maximizing the expected log-likelihood found on the E step. Finding a maximum likelihood solution requires taking the derivatives of the likelihood function with respect to all the unknown values. The derivative of the likelihood is zero at a point, which in turn means that the point is either a maximum or a saddle point or convergence point. [1] [6]

2.5.1 Estimation of RMM parameters using EM Algorithm

Formula for RMM

$$P\,(X_i) = \sum_{j=1}^{k} \prod_{ij} P\,(X_i/W_j) \qquad \text{……. (2.20)}$$

Here X_i – i^{th} pixel in the image

P (X_i) - pdf of X_i in RMM

$\prod_{ij}$ – prior probability

K – number of mixture components in RMM

W_j- parameter vector of the j^{th} component

P (X_i/W_j) – pdf of X_i in the j^{th} component

P (X_i/W_j) is calculated using Rayleigh distribution with parameter W_j :

$$P\,(X_i/W_j) = \frac{X_i - p_j}{\sigma_j^2}\, e^{\left(-\frac{(X_i - p_j)^2}{2\sigma_j^2}\right)} \qquad \text{……. (2.21)}$$

Here σ_j - mode p_j- translation component

W_j = {p_j,σ_j }

<u>Calculation of prior $\prod_{ij}$ in RMM :</u>

Step 1 : Let weight of the i^{th} pixel in the j^{th} component be ,

$$\textstyle\sum_j(X_i) == \frac{\overline{X_\iota} - r_j}{s_j^2}\, e^{\left(-\frac{(X_i - r_j)^2}{2s_j^2}\right)} \qquad \text{……. (2.22)}$$

J= 1,2,3….k r_j , s_j – parameters

$\overline{X_\iota}$ – mean value of pixels in the neighbourhood N_i of the i^{th} pixel

$$\overline{X_\iota} = \frac{1}{M+1}(X_i + \textstyle\sum_{m\in N_i} Y_m) \qquad \text{……. (2.23)}$$

Step 2 : Let the weight of the i^{th} pixel and its neighbourhood N_i in the j^{th} component be –

$$F_j\,(X_i) = [\tfrac{1}{M}\textstyle\sum_{m\in N_i}\sum_j(Y_m)]^{\alpha} \qquad \text{……. (2.24)}$$

α- used to control function shape of $F_j\,(X_i)$

Step 3 : Define the prior probability $\prod_{ij}$

$$\textstyle\prod_{ij} = \frac{F_j\,(X_i)}{\sum_{k=1}^{K} F_k(X_i)} \qquad \text{……. (2.25)}$$

The normalization of $F_j\,(X_i)$ aims to translate the range of $\prod_{ij}$ into [0,1]

Formulating the E and M step in EM algorithm.

<u>E-Step</u>

The objective function of EM algorithm for the RMM can be formulated as –

$$\phi\,(\Theta,\theta^t) = \textstyle\sum_{i=1}^{N}\sum_{j=1}^{k}\log(\prod_{ij})\, P(j/X_i\,,\theta^t) + \sum_{i=1}^{N}\sum_{j=1}^{k}\log(P(Y_i/W_j))\, P(W_j/X_i\,,\theta^t) \qquad \text{……. (2.26)}$$

here θ^t – known parameter in i^{th}iteration

Θ - unknown / updated parameter in i^{th}iteration

P $(W_j/X_i\,,\theta^t)$ – posterior distribution

$P(Y_i/W_j)$ - class- conditional probability density

In the above equation the first term does not contain the parameters W_j and the second term does not contain prior probability $\prod_{ij}$. Therefore, the maximization of the objective function

ϕ (Θ,θ^t) can be simplified to separately maximize the first term with respect to W_j and similarly second term with respect to $\prod_{ij}$.

<u>M- Step</u>

Applying steepest descent method

$\theta^{t+1} = \theta^t - \eta\frac{\partial \phi\,(\Theta,\theta^t)}{\partial\theta}$; η = 0.01 is the learning rate ……. (2.27)

Let Q be ϕ (Θ,θ^t)

$$\frac{\partial Q}{\partial\theta} = \begin{bmatrix} \frac{\partial Q}{\partial\theta_1} & \frac{\partial Q}{\partial\theta_2} & \cdots \frac{\partial Q}{\partial\theta_k} \\ \frac{\partial Q}{\partial p_1} & \frac{\partial Q}{\partial p_2} & \cdots \frac{\partial Q}{\partial p_k} \\ \frac{\partial Q}{\partial\sigma_1} & \frac{\partial Q}{\partial\sigma_2} & \cdots \frac{\partial Q}{\partial\sigma_k} \\ \frac{\partial Q}{\partial r_1} & \frac{\partial Q}{\partial r_2} & \cdots \frac{\partial Q}{\partial r_k} \\ \frac{\partial Q}{\partial s_1} & \frac{\partial Q}{\partial s_2} & \cdots \frac{\partial Q}{\partial s_k} \\ \frac{\partial Q}{\partial\alpha} & \frac{\partial Q}{\partial\alpha} & \cdots \frac{\partial Q}{\partial\alpha} \end{bmatrix} \quad \text{……. (2.28)}$$

Calculating $\frac{\partial Q}{\partial\theta_j}$:

1. The partial derivative of Q with respect to p_j

   $\frac{\partial Q}{\partial p_j} = -\sum_{i=1}^{N} P(j/X_i\ ,\theta^t\ )\ (\frac{X_i - p_j}{\sigma_j^2} - \frac{1}{X_i - p_j})$ ……. (2.29)

2. the partial derivative of Q with respect to σ_j

$$\frac{\partial Q}{\partial \sigma_j} = \sum_{i=1}^{N} P(j/X_i\ ,\theta^t\ )\ (-\frac{2}{\sigma_j} + \frac{(X_i - p_j)^2}{\sigma_j^3}) \qquad \text{……. (2.30)}$$

3. $$\frac{\partial Q}{\partial r_j} = \sum_{i=1}^{N}[\ P(j/X_i\ ,\theta^t\ )\ \frac{\alpha}{M\mathrm{F}_j(X_i\ )^{\frac{1}{\alpha}}}\ \sum_{m\in N_i}(\frac{(\overline{X_m} - r_j)^2}{s_j^4} - \frac{1}{s_j^2})\ .\ e^{-\frac{(\overline{X_m} - r_j)^2}{2s_j^2}}\ ] +$$

$$\sum_{i=1}^{N}\sum_{j=1}^{k}[\quad \mathrm{P}(k/X_i\ ,\theta^t)\frac{\alpha\ \mathrm{F}_j X_i^{\ \frac{\alpha-1}{\alpha}}}{M\sum_{l=1}^{k}\mathrm{F}_l(X_i\ )}\ .\sum_{m\in N_i}(\frac{(\overline{X_m} - r_j)^2}{s_j^4} - \frac{1}{s_j^2})\ .\ e^{-\frac{(\overline{X_m} - r_j)^2}{2s_j^2}}\ ] \qquad \text{……. (2.31)}$$

4. $$\frac{\partial Q}{\partial s_j} = \sum_{i=1}^{N}[\ P(j/X_i\ ,\theta^t\ )\ \frac{\alpha}{M\mathrm{F}_j(X_i\ )^{\frac{1}{\alpha}}}\ .\sum_{m\in N_i}(\frac{(\overline{X_m} - r_j)^3}{s_j^5} - 2\ \frac{\overline{X_m} - r_j}{s_j^3})\ .\ e^{-\frac{(\overline{X_m} - r_j)^2}{2s_j^2}}\ ] +$$

$$\sum_{i=1}^{N}\sum_{j=1}^{k}[\quad \mathrm{P}(k/X_i\ ,\theta^t)\frac{\alpha\ \mathrm{F}_j X_i^{\ \frac{\alpha-1}{\alpha}}}{M\sum_{l=1}^{k}\mathrm{F}_l(X_i\ )}\ .\sum_{m\in N_i}(\frac{(\overline{X_m} - r_j)^3}{s_j^5} -2\ \frac{\overline{X_m} - r_j}{s_j^3})\ .\ e^{-\frac{(\overline{X_m} - r_j)^2}{2s_j^2}}\ ] \qquad \text{……. (2.32)}$$

5. $$\frac{\partial Q}{\partial s_j} = \sum_{i=1}^{N}\sum_{j=1}^{k}[\quad \mathrm{P}(j/X_i\ ,\theta^t)\ .\ [\ \log \mathrm{F}_j\ (X_i)^{\frac{1}{\alpha}} - \frac{\sum_{k=1}^{K}\mathrm{F}_k(X_i)\ \log(\mathrm{F}_k\ (X_i)^{\frac{1}{\alpha}})}{\sum_{l=1}^{k}\mathrm{F}_l(X_i\ )}\ ] \qquad \text{……. (2.33)}$$

<u>Estimating parameters of RMM by EM algorithm </u>–

Step 1: initialize the parameter set Θ

1. The class number k is set at 5 and α is initialized as $\alpha^{(0)}$ = 2

   Now k-means algorithm is used to cluster the pixel in the image based on the gray intensity in order to find the mean gray scale values of k classes $\mu_1^{(0)}, \mu_2^{(0)}, \ldots, \mu_k^{(0)}$ and denote

   $$m^{(0)} = [\mu_1^{(0)}, \mu_2^{(0)}, \ldots, \mu_k^{(0)}\ ] \qquad \text{……. (2.34)}$$

2. Let $p_j^{(0)}$ be the minimum gray value of pixels in the above class correlated to $\mu_j^{(0)}$ and denote translation component $p^{(0)} = [p_1^{(0)}, p_2^{(0)}, \ldots, p_k^{(0)}]$ ……. (2.35)

3. The relation between mean value μ , mode σ , and p is –

   $$\mu = p + \sigma\sqrt{\frac{\pi}{2}} \qquad \text{……. (2.36)}$$

   So the mode $\sigma^{(0)} = [\sigma_1^{(0)},\sigma_2^{(0)}, \ldots,\sigma_k^{(0)}]$ is $\sigma^{(0)} = (\ m^{(0)} - \ p^{(0)})\sqrt{\frac{2}{\pi}}$ ……. (2.37)

4. Let $r^{(0)} = p^{(0)}$ and $s^{(0)} = \sigma^{(0)}$

Step 2: computing the posterior probability

For i = 1, 2,3 … N and j = 1,2,3 … K

P $(X_i/W_j{}^t)$ and Π_{ij}^t are calculated

Then we compute $P(j/X_i\ ,\Theta^t\ )$ by Bayes Theorem as –

$$P(j/X_i\ ,\Theta^t) = \frac{P(j/X_i ,\Theta^t)}{P(X_i /\Theta_t)} = \frac{P(j/\Theta^t).P(X_i /\Theta^t,j)}{\sum_{k=1}^{K} P(k/\Theta^t).P(X_i /\Theta^t,j)} \quad \text{……. (2.38)}$$

$$= \frac{\Pi_{ij}^t\ P(X_i /W_j{}^t)}{\sum_{k=1}^{K} \Pi_{ik}^t\ \mathrm{P}\ (X_i/W_k{}^t)} \quad \text{……. (2.39)}$$

Step 3: computing the parameter set

Substituting the above equation into the partial derivative equations and finally updating

Θ^t to Θ^{t+1} .

Step 4: Reach convergence

After i^{th} iteration, if | Θ^{t+1} - Θ^{t-1} | < 0.0001, the iteration process can reach the convergence. If not, return to step 2 and continue the iteration. [13]

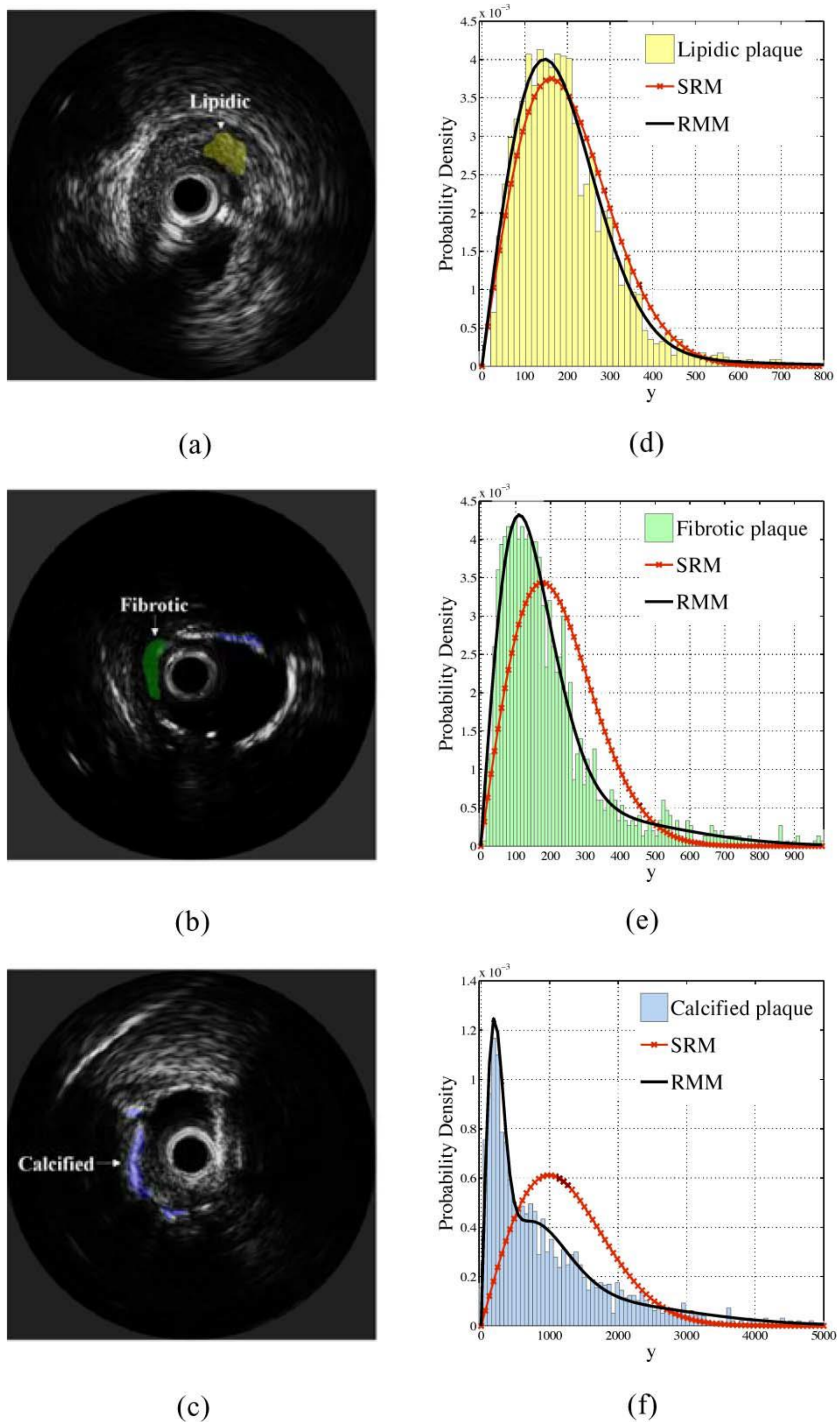


Figure 2.10 (a)–(c) RMM modelling of three tissue types. (d)–(f) Three-component mixture PDFs estimated for each tissue type, overlapped with single Rayleigh PDFs adopted from [6].

3. APPLYING MORPHOLOGICAL OPERATIONS

Morphological operations or algorithms are applied to binary images for removing imperfections or noise from binary images. It is based on set theory concepts. The key points covered in this work are set theory, structuring element, erosion, dilation, erosion, opening and closing [14].

Morphological operations are applied on the black and white image (binary image).

3.1 Set Theory

Looking at the Set theory concepts,

1) Considering 2D binary images, it can be said that A is the (unordered) set of pairs (x, y) such that the image value at (x, y) is equal to 1
2) A = {(x, y) | I_A(x, y)=1}

   The vertical bar is used to convey the meaning- such that
3) Union of two sets A and B

   The set of elements belonging to either A, or B, or both

   $A \cup B$
4) Intersection of two sets A and B

   The set of elements belonging to both A and B

   $A \cap B$
5) When we say w is an element of set A

   $w \in A$
6) Complement

   The set of elements that are not in A

   $A^c = \{ w \mid w \nexists A \}$
7) Difference of two sets A and B

   The set of elements that belong to A but not to B

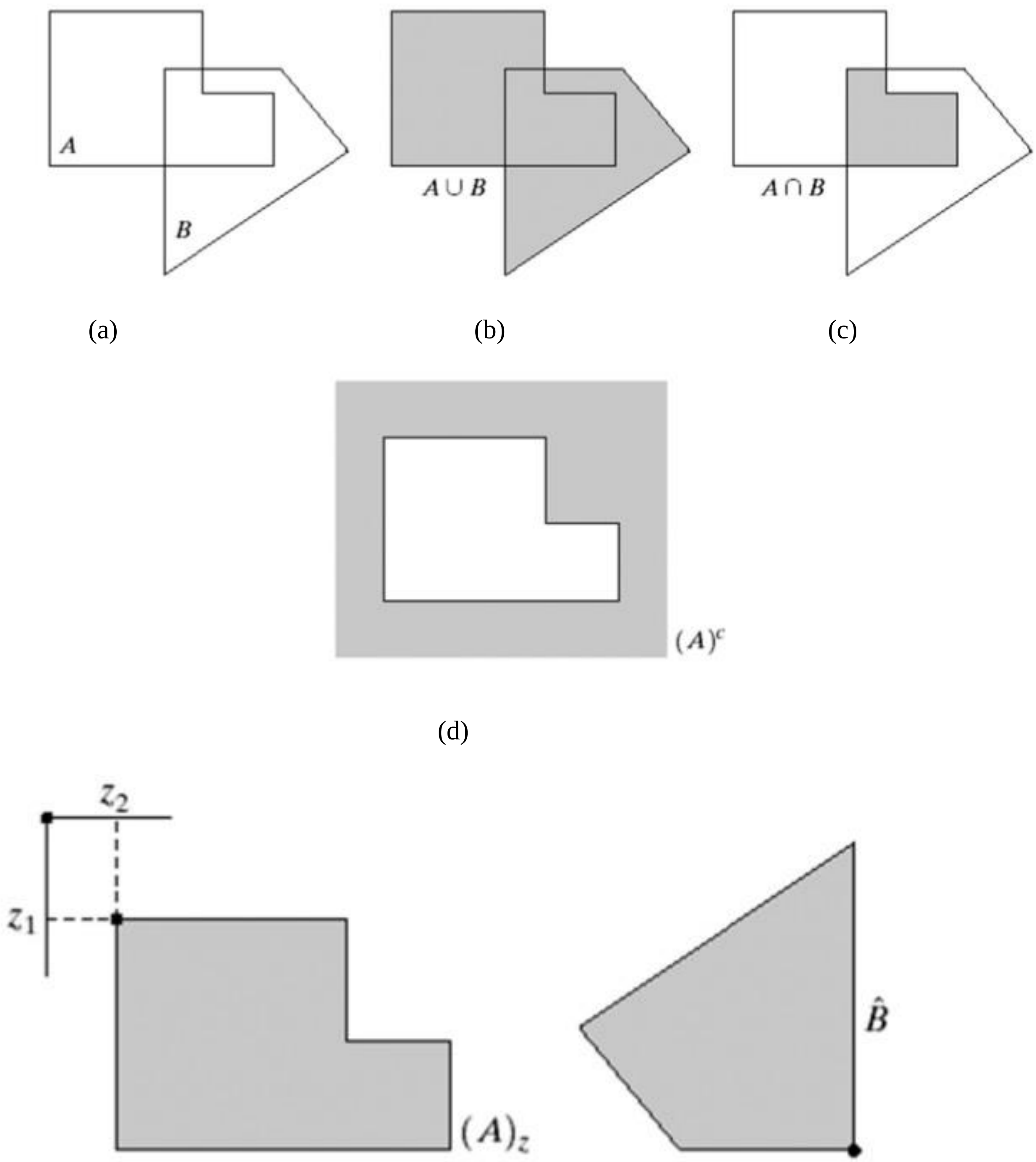
   $A - B = \{w \mid w \in A, w \nexists B \}$
8) Subset

   A becomes a subset of B if every element of A is also present in B

   $A \subset B$
9) Empty set {}

   $\emptyset$
10) Reflection and Translation

$$\hat{B}=\{ \vec{w} \mid \vec{w} = -\hat{b} \text{ , for } \hat{b} \in \mathrm{B} \} , \qquad (B)_z=\{ \vec{c} \mid \vec{c} = \vec{b}+\vec{z} \text{ , for } \vec{b} \in \mathrm{B}\}$$



(a) (b) (c)

(d)

(e) (f)

Figure 3.1 (a) Two sets A and B. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A and B. (e) The translation of A. (f) The reflection of B adopted from [14].

3.2 Structuring Element

A structuring element is a small set or sub image, used to probe for structure [14].

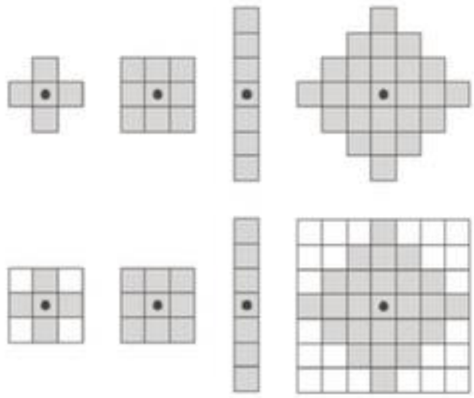


Figure 3.2 First row: Examples of Structuring elements. Second row: Structuring elements converted to rectangular arrays with centers of the SEs seen in bold adopted from [14].

3.3 Erosion

Erosion of a set A by some other set (i.e. a structuring element) B is defined as A eroded by B is all element z such that the translate of B by z, it is entirely within A(i.e. a subset of A). If the shifted B is inside A completely, output a 1(binary) at that location. Erosion enlarges holes, breaks thin parts and shrinks objects [14].

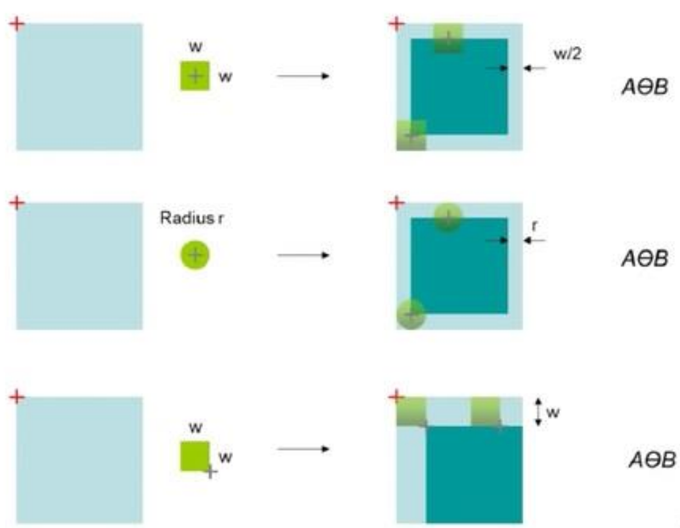


Figure 3.3a Example of Erosion for an image with different structuring elements. The darker shade of blue is what is left in all the cases due to erosion adopted from [14].

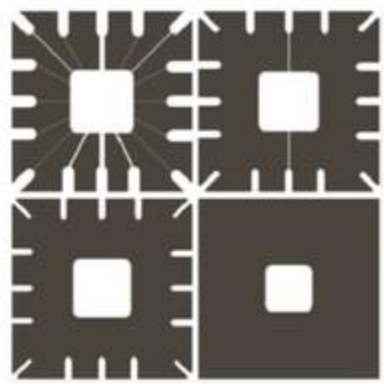


Figure 3.3b Using erosion to remove image components. (a) A 456x456 binary image of a wire-bond mask. (b)-(d) Image eroded using square structuring elements of sizes 11x11, 15x15 and 45x45, respectively adopted from[14].

3.4 Dilation

Dilation of a set A by a set (Structuring element) B means that all the points z are desired such that the reflected value of B offset by z followed by checking if it intersects with A at all (i.e. intersection is not empty), then the point is in the dilation. It can be interpreted as reflect B, shift it by z and if it overlaps at all with A, output 1(binary) at the center of B [14].

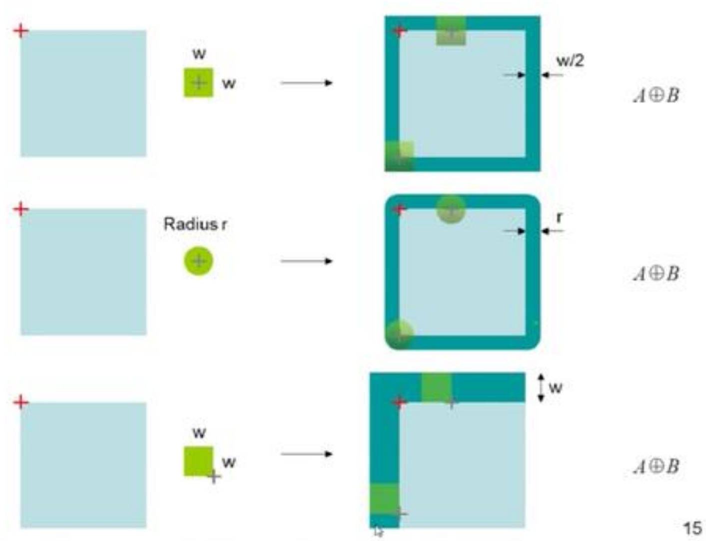
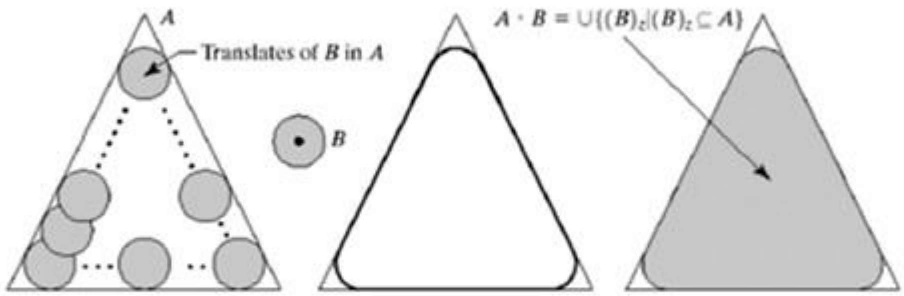


Figure 3.4 Dilation of an image using different structuring elements resulting in the portion visible as the darker shade of blue [14].

3.5 Opening

Opening is a combination of both erosion and dilation. It is defined as an erosion followed by a dilation where the same structuring element B is used for both. In other words, it is the union of all the translations of B that fit into A [14].

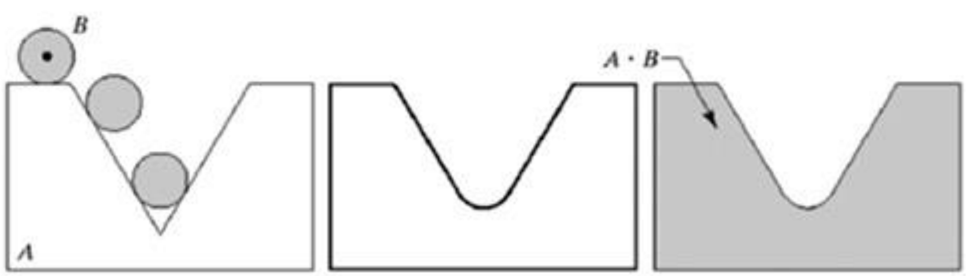


(a) (b) (c)

Figure 3.5 Example of opening. (a) Structuring element B moving along the inside of the boundary of A. (b) The heavy line is the outer boundary of the opening. (c) Complete opening (shaded) adopted from [14].

3.6 Closing

Closing is opposite of erosion. It is dilation followed by erosion where the structuring element B remains the same for both. In other words, the union of all the translation of B that do not intersect A, closing is the compliment of that [14].



(a) (b) (c)

Figure 3.6 Example of closing. (a) Structuring element B moves on the outside boundary of set A. (b) Heavy line is the outer boundary of the closing. (c) Complete closing (shaded) adopted from [14].

4. BLOB DETECTION

A binary large object (blob) is a large cluster of pixels from which objects can be detected [5]. Image acquired is pre-processed to make it suitable for the operation of blob detection. That is why GMM (Refer Chapter 2, Section 1) and morphological operations (Refer Chapter 3) are done prior to blob detection.

4.1 Pre-processing step

Pre-processing step prepares image for the blob detection process. Every image is processed in the form of matrix where each location is considered as a pixel.

Image is considered as a matrix of n x m where n is number of rows and m is number of columns. Start column, start row, end column and end row are required for identifying connected pixels. Once the connected pixels in each row are identified, the counted pixels in different rows are labelled. [5]

4.2 Area and Centroid of Blob

Area of an object is calculated as the total number of pixels in the given connected component. The column positional values are added together for the same label and divided by area of the same label. The obtained value is the column of the centroid. The same process is repeated for the row positional values to obtain the row of the centroid. Figure 4 shows how different objects are distinguished in an image matrix.

0	0	1	1_C	1	0
0	1	1	1	0	2
0	0	0	0	2	2
3_C	3	3	0	2	2
3	3	0	0	2	2_C
3	0	0	0	0	0

Figure 4 Centroid for labelled objects. Objects are labelled as 1, 2, and 3 respectively with their corresponding centroid as 1_ C, 2_ C, and 3_ C adopted from [5].

This detection mechanism finds the blob's position in successive image frames. The blob area must be defined before any detection of blob where Pixels with similar light values or colour values are grouped together to find the blob. Every surface has subtle variations in real world scenario, so if only one light or colour value is selected, a blob might be only few pixels.

The system must detect the blobs in the new image and make meaningful connections between the seemingly different blobs present in each frame. It needs to define the relative importance of factors including location, size and colour to decide if the blob in the new frame is similar enough to the previous blob to receive the same label. [5]

4.3 Contour Detection Algorithm

In this research blob detection uses contrast in a binary image to compute a detected region, it's centroid, and the area of the blob. The GMM supplies the pixels detected as foreground. These pixels are grouped, in current frame, together by utilizing a contour detection algorithm. The contour detection algorithm groups the individual pixels into disconnected classes, and then finds the contours surrounding each class. Each class is marked as a candidate blob (CB). These CB are then checked by their size and small blobs are removed from the algorithm to reduce false detections. The positions of the CB, in current frame, are compared using the k-Means clustering that finds the canters of clusters and groups the input samples CB around the clusters to identify the vehicles in each region. The moving vehicle is counted when it passes the base line. When the vehicle passes through that area, the frame is recorded. In each region the blob with the same label are analysed and the vehicle count is incremented.

Blob analysis identifies potential objects and puts a box around them. An additional rule that the ratio of area of blob to the area of rectangle around a blob should be greater than 0.4 ensures that unnecessary objects are not detected.

The foremost challenge however in the implementation of blob detection algorithm is to reduce the computational time. [5]

5. GAUSSIAN FILTER

It is a part of the pre-processing in the project. Applying Gaussian filter is also called Gaussian smoothing. It is the result of blurring an image by a Gaussian function. It is typically used to reduce image noise and reduce detail. The visual effect of this blurring technique is a smooth blur resembling that of viewing the image through a translucent screen. Since the Fourier transform of a Gaussian is another Gaussian, applying a Gaussian blur has the effect of reducing the image's high-frequency components; a Gaussian blur is thus a low pass filter. [7]

6. FEATURE EXTRACTION USING GREY LEVEL CO-OCCURENCE MATRIX (GLCM)

This refers to the textural information that can be extracted from an image. It is calculated by finding how often a pixel with intensity say 'i' occurs in a particular spatial relationship to another pixel say 'j' in the image. The GLCM features are computed by either varying the distances or by using the directions.

It should be noted that pixel intensity is often referred to as the variation in gray level. The GLCM element is a function say E (i, j, d, Ө) representing the probability of the pair of pixels, which are located with an inter sample distance d and a direction Ө, having a gray level i and a gray level j. [12]

6.1 Various features that are commonly extracted

Various features that can be extracted are contrast, correlation, energy, homogeneity, etc. Contrast measures the intensity between the pixel and its neighbor. Correlation can be found by finding the joint probability of a pair of pixel of GLCM occurring. Energy is estimated by the finding the sum of square of pixel values of GLCM. It is used to determine uniformity. Homogeneity specifies as to how close the distribution of elements is to the GLCM diagonal. Textures obtained by homogeneity usually contain structures that are repeating. The feature used in this work is homogeneity. It is seen that one can obtain high frequency of information through GLCM measurements by setting the inter-pixel distance to either 1 or 2. [12]

6.2 Representation in different grey levels

Image is applied to 8 and 16 grey levels and after that the features are extracted. One can plot the measurement of features extracted in both 8 and 16 level representation. Analysis and observation can then be made as to which feature is prominent in which of the two representations. For instance, say that contrast is the only feature which is prominently high for 16 level representation whereas energy, correlation and homogeneity are higher for 8 level representation. [12]

7. K- NEAREST NEIGHBOUR CLASSIFIER (KNN CLASSIFICATION)

KNN classification is used in this model to categorize as to which type of plaque is present in the image. Is it fibrotic, lipidic or calcified? Suppose there is a dataset that has been divided into two classes- red and blue. This is considered to be the training set. Also, consider a testing point on which the probe is to be carried out. The task is to classify this point into one of the two classes. How can this be done? One way of doing this is to draw the Gaussian or calculate threshold. This can be done when the complexity is high. However, in low complexity cases (where KNN is mostly used), one can really look at the point and say it is really closer to the red example (let) than to the blue example and so this point should be red. This is the intuition behind KNN classification. Also, once the distances are being calculated, it is observed that not only this particular testing point but also the region around this testing point belongs to the red class. This region signifies that it is closer to a certain training example than to any other. Using Euclidian distances this region forms a polygon. So the entire training set can be divided into these small-

small regions and a decision boundary could be obtained that separates the two classes. Nearest neighbor classifiers have lots of freedom and is flexible to over fit the training data. [8]

7.1 Significance of K

A particular example is suppose there is an outlier, i.e. some portion of training data mislabeled, the decision boundary changes dramatically as a result. This is a drawback as efficient algorithms shouldn't really get affected by small changes in the training data. This happens with KNN because there is no computation of priors or confidence in this method. This problem can however be reduced by considering more than one nearest neighbor and which also brings to the significance of K in KNN. Let there be two blue neighbors and one red neighbor, the result can be obtained by a majority vote. [8]

7.2 The algorithm

There are three things given- training set representation say A, the class label B and the testing point C. What the algorithm does is it computes the distance between C and every training example in A. Out of these distances the K closest neighbors are picked and the corresponding classes of the K closest neighbors are looked at and the most frequent one is selected. [11]

7.3 Choosing the value of K

As K is varied, KNN starts behaving in different ways. If K is too large, entire training set will come under nearest neighbor and then the more probable class will be the resultant. If K is too small, it will be very sensitive to outliers. The best way to select K is to split the training set into two halves. Apply KNN classifier to each part separately and then vary K by trying different values and select the one which gives the best results. [11]

7.4 Applying KNN classification

Given that the dataset is in the form of matrices namely sample matrix A, training matrix B and group matrix C.

A= [51, 62	B= [2, 0	C= {'First row'
8, 3	201, 32	'Second row'
14, 11	20, 11]	'Third row'}
101, 201]		

Each row of sample matrix is compared with each row of training matrix to identify the nearest or closest row among them. For instance nearest row for 51, 62 is 20, 11. One of the conditions that must be followed is that the number of columns in both A and B should be same. Each row of group matrix represent a value to training matrix. It is a column matrix. The number of rows in training and group matrix should be same. The result will be in terms of group matrix. It is as follows.

'Third row'

'First row'

'Third row'

'Second row'

This indicates that first row of A is closest to third row of B, the second row of A is closest to first row of B, the third row of A is closest to third row of B and the fourth row of A is closest to second row of B. [11]

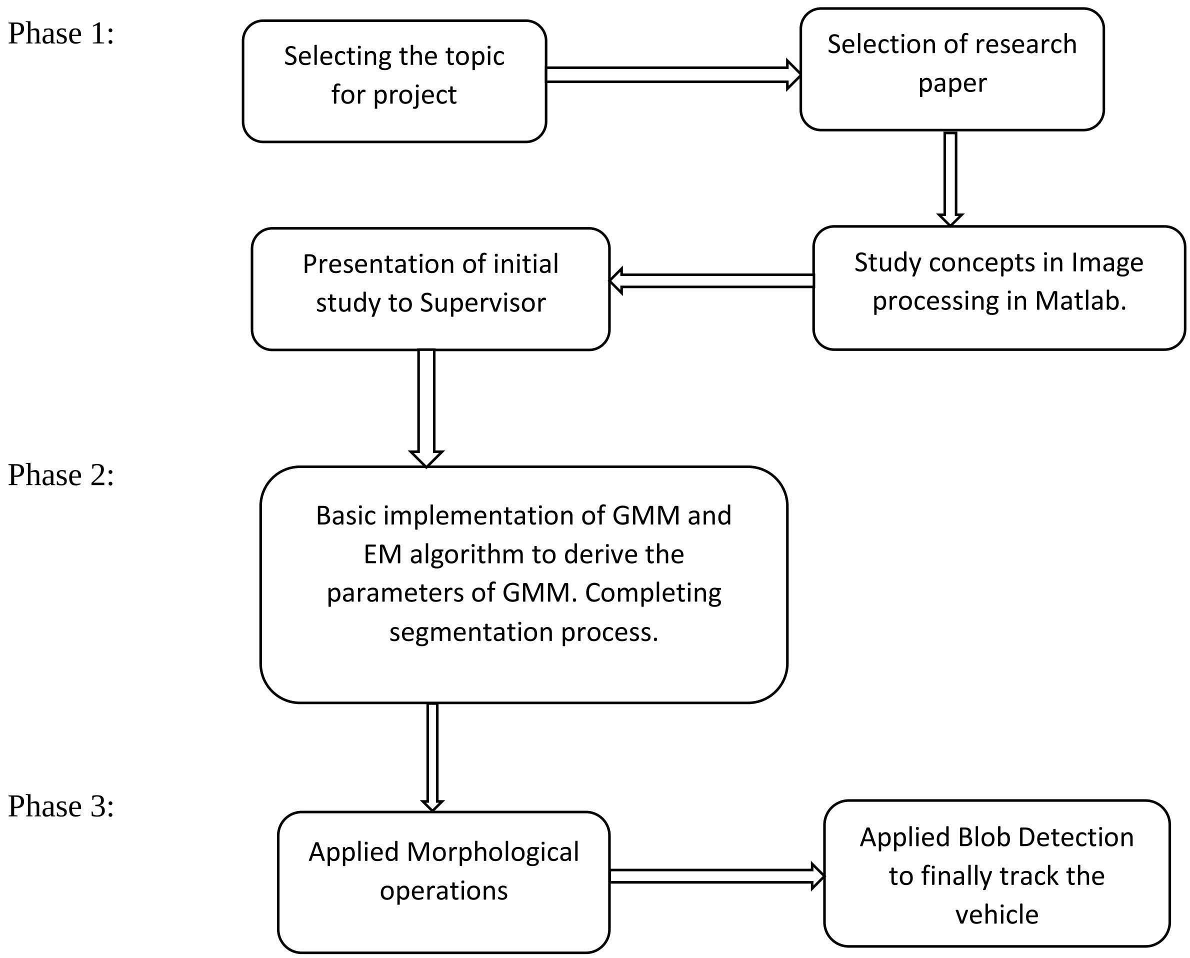
8. PRIMARY TOOLS

MATLAB

The software is used as a mathematical tool that plots all the mathematical equations derived in this work. This tool helps in showing the variation of one parameter on changing the other parameter. MATLAB provides a unique way of data analysis and development of algorithms as well as applications with a variety of visualization tools. Complex mathematical calculations are made simpler. It features a large number of mathematical, statistical, and engineering functions. The primary reason it is chosen over other simulation tools for this work is because of its accurate modelling of images. In MATLAB, an image is treated as a matrix and every pixel of the image as a matrix element. All the operators in MATLAB like +, -, *, /, ^, sqrt, sin, cos etc. which are defined for matrices can therefore be defined for images. Not only this, all the common image formats like bmp, jpeg, gif, png etc. can be imported or exported by MATLAB. Also, a wide variety of data types are supported here some of which are Double, Single, Int32, Int16, Int8, Uint32, Uint16, Uint8. The different types of images that are possible here are firstly a binary image that simply contains 1 for white and 0 for black. Then there are grey scale images also called as intensity images that contain range of numbers from 0 to 255. Also, there are coloured images which may be represented as RGB image or an indexed image. RGB has three indexes namely the red portion, green and blue portions. There are inbuilt functions available for image type conversion as well like rgb2gray converts RGB image to Intensity image and im2bw which converts RGB image to Binary image and many more. The version of MATLAB used in this work is R2013a (8.1). [9]

9. PROJECT DEVELOPMENT

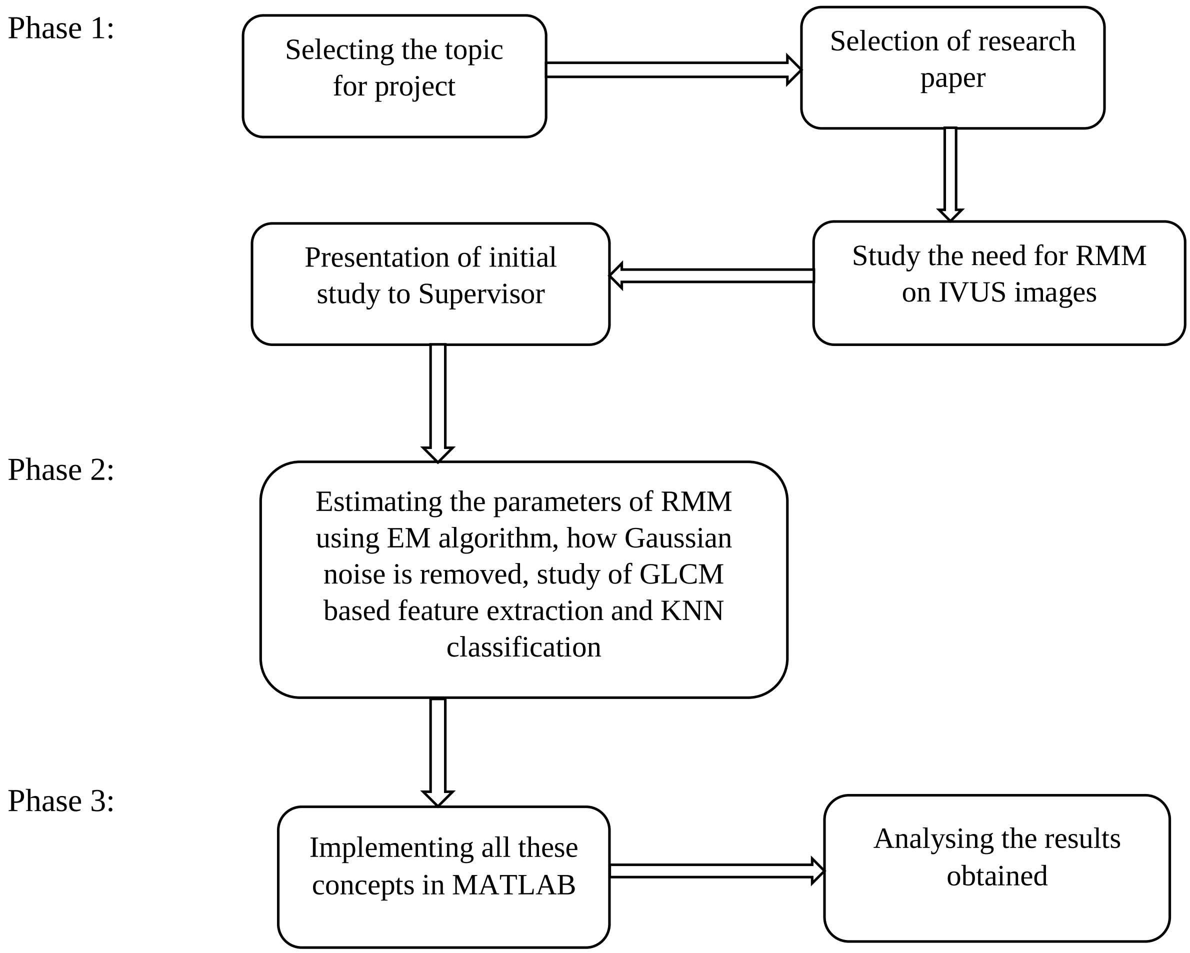
9.1 Part 1: Vehicle Detection and Tracking Method



(a)

Figure 9(a) Project part 1 development step by step

9.2 Part 2: Plaque Characterization in Intravascular Ultrasound



(b)

Figure 9(b) Project part 2 development step by step

10. IMPLEMENTATION AND RESULTS

10.1 Part 1: Vehicle Detection and Tracking Method

A video clip from the surveillance camera is taken as input (Refer Figure 10.1). This video is then converted into sets of 50 frames. Each frame is read one after the other in a loop. Now background subtraction is applied to each frame one by one. The background and foreground pixels are separated. The pixels that remain unchanged throughout a set of 50 frames are considered as a part of the background while the pixels changing in the set are taken as a part of the foreground. This is carried out by applying GMM using EM algorithm.

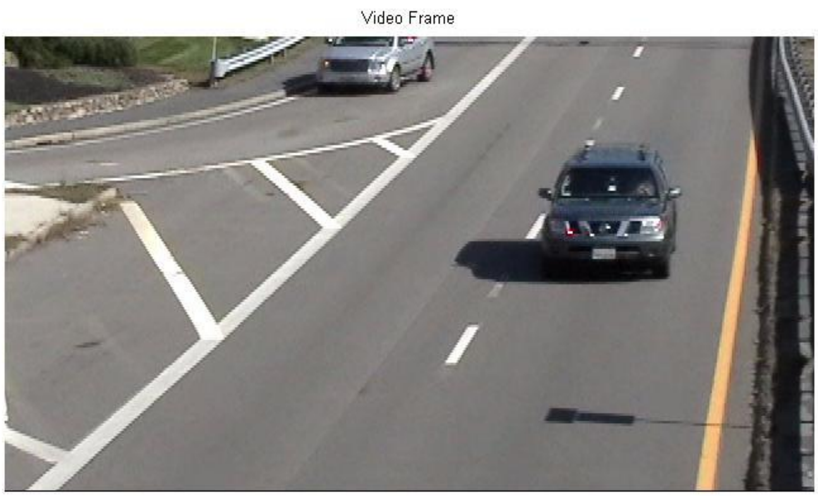


Figure 10.1 Original image [Source: MATLAB 2013a, vision toolbox, visiontraffic.avi].

Number of Gaussians per frame is 3

The initial parameters for applying GMM through EM are as follows:

Mean μ is a matrix calculated using number of Gaussians and the number of pixels

Covariance matrix should not be less than 30×30

Weight w should not be less than 0.05

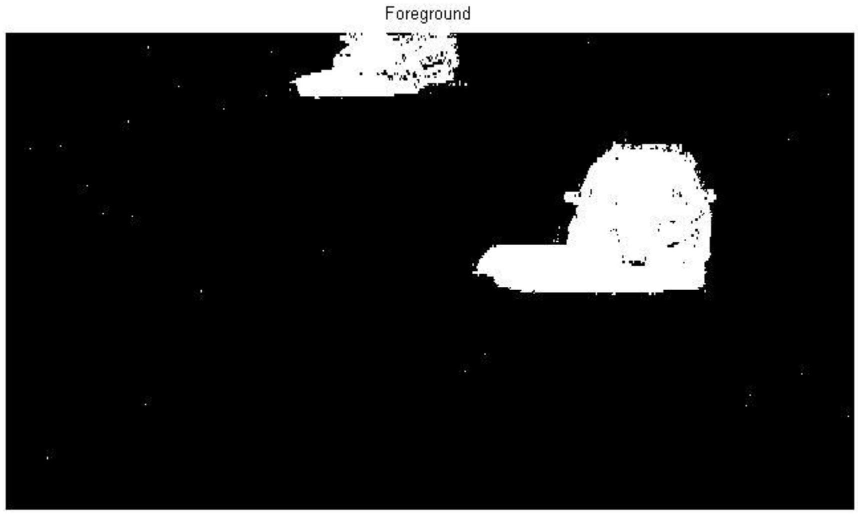


Figure 10.2 After applying Gaussian Mixture Model [Source: MATLAB 2013a, vision toolbox, visiontraffic.avi].

Morphological operation is carried out to remove noise from the image received after GMM.
The structuring element taken here is a 3×3 square matrix. Opening operation is performed using this structuring element.

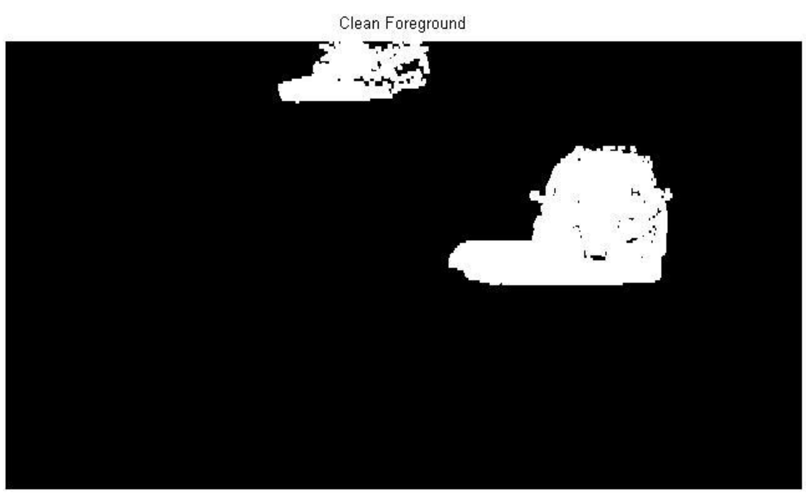


Figure 10.3 After applying Morphological Operations [Source: MATLAB 2013a, vision toolbox, visiontraffic.avi].

From the above image blobs are identified and a bounding box is defined for each block. This bounding box is a rectangle and has a minimum blob area of 150 pixels. The number of bounding boxes obtained in a frame represent the number of cars detected. The count for the cars detected are displayed at the top left corner of the frame as shown in Figure 10.4.

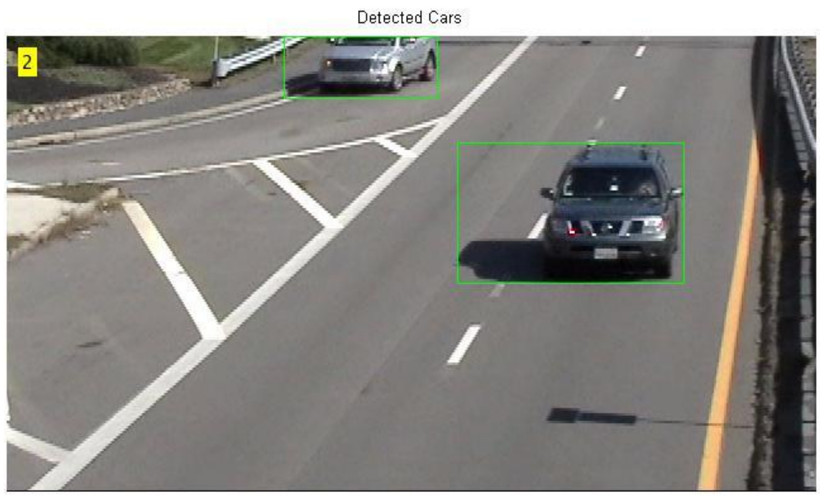


Figure 10.4 After applying Blob Detection [Source: MATLAB 2013a, vision toolbox, visiontraffic.avi].

10.2 Part 2: Plaque Characterization in Intravascular Ultrasound

An IVUS image is taken as input. It is in jpeg format with dimensions of 256x256. [15]

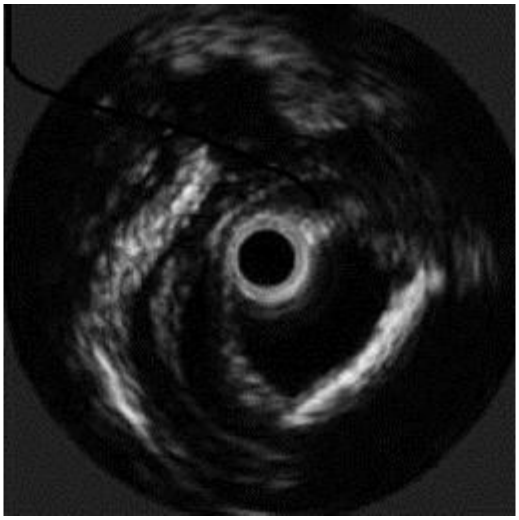


Figure 10.5 Original image adopted from [13]

Pre-processing is the first operation that is carried out (Refer chapter 5). First a predefined 2-D filter is created. A Gaussian low pass filter of size [5, 5] and standard deviation 2 is used in this case. A predefined function takes in the image and filter as argument and performs filtering on the image.

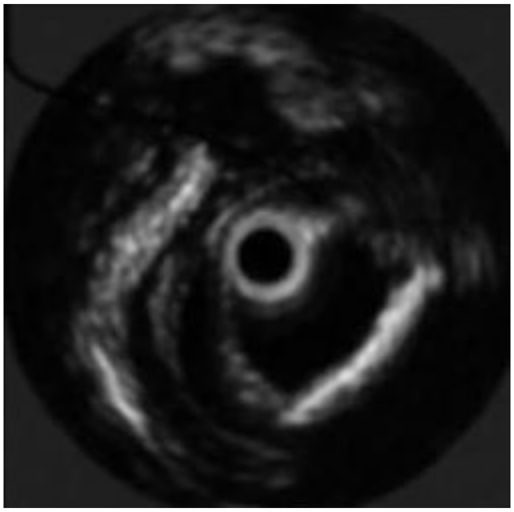


Figure 10.6 After pre-processing [Source: MATLAB 2013a]

The next step is feature extraction using GLCM (Refer chapter 6). Its inbuilt function takes the input image as argument. Another parameter that has a role to play is offset which is an array specifying the distance between the pixel of interest and its neighbour. The offset is taken as [2, 0; 0, 2].

Feature:	Contrast	Correlation	Energy	Homogeneity
Values:	3.0055	0.2993	0.8534	0.4457

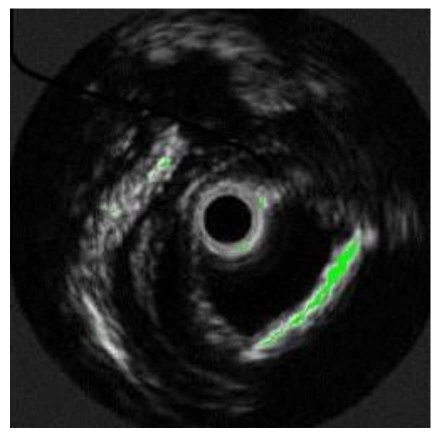
Figure 10.7 After feature extraction using GLCM [Source: MATLAB 2013a]

This leads to the KNN classification part. The classifier function takes three set of matrices as argument namely the sample matrix, the training matrix and the group matrix (Refer chapter 7). The output will be a popup box saying which type of plaque it is. For this particular input image the plaque found was 'Fibrotic Plaque'.

Morphological operation is done next in order to distinctly identify the region of plaque in the image (Refer chapter 5). For this the image is converted to black and white. Then a disc shaped structuring element is considered whose radius is 3. Next, erosion of the image by this structuring element is carried out. From the eroded image the region of plaque is identified and highlighted. Finally, the volume of this affected region is calculated.

(a)

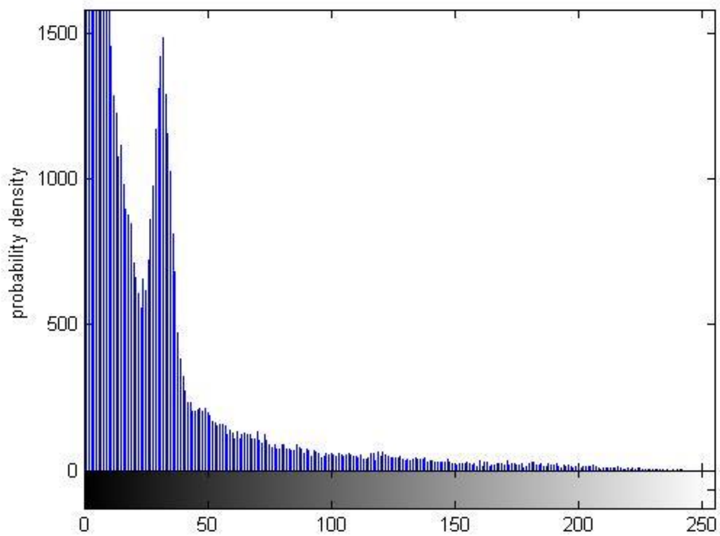
(b)



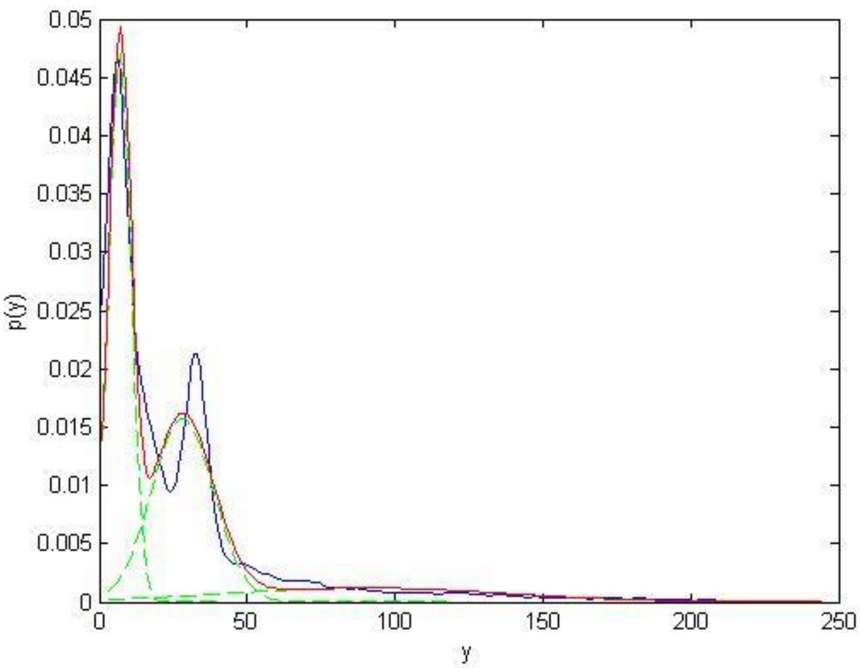
(c)

Figure 10.8 Different stages in morphological operations. (a) The image converted to BW. (b) The BW image eroded by the structuring element. (c) The Fibrotic plaque highlighted in the original image. [Source: MATLAB 2013a]

Lastly, the analysis of the pdf of the fibrotic plaque that is done by estimating the parameters of the Rayleigh distribution in the mixture model using the EM algorithm (Refer chapter 2, section 5). The values of the parameters are compared to the experimental data.



(a)



(b)

Figure 10.9 (a) The plot of Rayleigh probability density function and the number of histograms. (b) Different curves from dark to light for a range of σ (from 100-1000). [Source: MATLAB 2013a]

CONCLUSION

Through the implementation of part 1 of the project, the basics of clustering in image processing mainly Gaussian Mixture model and ways of enhancing binary images and detecting objects are learnt. The concepts of GMM has implications in various other image processing based work like pattern recognition, gesture recognition, etc.

Rayleigh Mixture Model is implemented for characterizing plaque in IVUS data for part 2 of the project. Gaussian filter is used to remove noise. GLCM is used for feature extraction. KNN is implemented to classify three types of plaques. Morphology is applied to identify the plaque regions distinctly. Finally, the coefficients and parameters of the mixture model are used to analyse fibrotic, lipidic and calcified plaques. RMM could significantly contribute to a more accurate study of plaque composition and consequently to an objective identification of vulnerable plaques.

REFERENCES

[1] Bhaskar, P.K.,Yong, S.P.“Image Processing Based Vehicle Detection and Tracking Method”. Image Processing, IEEE 2014 International Conference, June 2014. vol., no., pp: 1-5.

[2] Gu, D.B., Sun, J.X. “EM image segmentation algorithm based on an inhomogeneous hidden MRF model”. Vision, Image and Signal Processing, IEEE Proceedings, 8 April 2005, vol. 152, no. 2, pp: 184 – 190.

[3] Evangelio, R.H et al.“Adaptively Splitted GMM with Feedback Improvement for the Task of Background Subtraction”. Information Forensics and Security, IEEE Transactions, 26 March 2014, vol. 9, pp: 863-874.

[4] Blekas, K et al. “A spatially constrained mixture model for image segmentation”. Neural Networks, IEEE Transactions, , March 2005, vol. 16, pp: 494-498.

[5] Nguyen, T.B., Chung, S.T. “An Improved Real-time Blob Detection for Visual Surveillance”. Image and Signal Processing, CISP 2009. 2nd International Congress on 17-19 Oct. 2009, vol., no.,pp: 1-5.

[6] Seabra JC, Ciompi F, Pujol O, Mauri J, Radeva P, et al., “Rayleigh mixture model for plaque characterization in intravascular ultrasound”, IEEE Transactions on Biomedical Engineering, vol. 58, no. 5, pp. 1314–1324, May 2011.

[7] Sorensen K, Andersen S, “Rayleigh mixture model-based hidden Markov modeling and estimation of noise in noisy speech signals,” IEEE Trans. Audio, Speech, Lang. Process., vol. 15, no. 3, pp. 901–917, Mar.2007.

[8] Demˇsar J, “Statistical comparisons of classifiers over multiple data sets,”J. Mach. Learning Res., vol. 7, pp. 1–30, 2006.

[9] Lovrenko V, Sutton C, “Mixture models and EM”,2011. [Online]. Available: http://www.inf.ed.ac.uk/teaching/courses/iaml/2011/slides/em.pdf. [Accessed: 10-Oct-2015].

[10] Murthy CA, “Gaussian Mixture Model”, 2014. [Online]. Available: http://www.cse.iitm.ac.in/~vplab/courses/DVP/PDF/gmm.pdf. [Accessed: 1-Nov-2015].

[11] Lovrenko V, Goddard N, “K-NN1: nearest neighbour algorithm”,2014. [Online]. Available: https://www.youtube.com/watch?v=k_7gMp5wh5A. [Accessed: 24-May-2016].

[12] Chandran S, “Comparison and analysis of feature extraction algorithms”, 2012. [Online]. Available: https://www.youtube.com/watch?v=QIvBqdgnxEc. [Accessed: 24-May-2016].

[13] Gao Z, Guo W, et al., “Automated Detection Framework of the Calcified Plaque with Acoustic Shadowing in IVUS Images”, 2014. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4220935. [Accessed: 24-May-2016].

[14] Gonzalez RC, Woods RE, “Morphological Image Processing”, in Digital Image Processing, Prentice Hall, 2008, pp: 541-550.

[15] Marques O, “MATLAB Basics”, in Practical Image and Video Processing using MATLAB, Wiley-IEEE Press, 2011, pp: 35-60.