CHAPTER 4

SEGMENTATION

4.1 INTRODUCTION

One of the most efficient methods for breast cancer early detection is mammography. A new method for detection and classification of micro calcifications is presented. It can be done in four stages: first, preprocessing stage deals with noise removal, and normalized the image. Second stage, Fuzzy c-Means clustering (FCM) is used for segmentation and pectoral muscle extraction using area calculation and finally micro calcifications detection. Third stage consists of two dimensional discrete wavelet transforms are extracted from the detection of micro calcifications. And then, nine statistical features are calculated from the LL band of wavelet transform. Finally, the extracted features are fed as input to the Artificial Neural Network and are classified into normal or abnormal (benign or malignant) images. The given classification approach is applied to a database of 322 dense mammographic images, originating from the MIAS database. The results are analyzed using MATLAB.

Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze image segmentation...
is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

### 4.2 SEGMENTATION USING FUZZY C-MEANS CLUSTERING ALGORITHM

Clustering is the process of dividing data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Depending on the nature of the data and the purpose for which clustering is being used, different measures of similarity may be used to place items into classes, where the similarity measure controls how the clusters are formed. Some examples of measures that can be used as in clustering include distance, connectivity, and intensity.

In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. In **fuzzy clustering** (also referred to as **soft clustering**), data elements can belong to more than one cluster, and associated with each element is a set of membership levels. These indicate the strength of the association between that data element and a particular cluster. Fuzzy clustering is a process of assigning these membership levels, and then using them to assign data elements to one or more clusters.

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data image to belong to two or more clusters. This method (developed by Dunn in 1973 and improved by Bezdek in 1981) is frequently used in pattern recognition (Chan et al 1987). The FCM algorithm attempts to partition a finite collection of n elements $X = \{x_1, \ldots, x_n\}$ into a collection of c fuzzy clusters with respect to some given criterion. Given a finite set of
data, the algorithm returns a list of c cluster centers $\mathbf{C} = \{c_1, ..., c_c\}$ and a partition matrix.

**Syntax**

```
[Center, U, obj_fcn] = fcm (data, cluster_n)
```

**Description**

```
[Center, U, obj_fcn] = fcm (data, cluster_n) applies the fuzzy c-means clustering method to a given data set.
```

The input arguments of this function are,

- `data`: data set to be clustered; each row is a sample data point
- `cluster_n`: number of clusters (greater than one)

The output arguments of this function are,

- `Center`: matrix of final cluster centers where each row provides the center coordinates
- `U`: final fuzzy partition matrix (or membership function matrix)
- `obj_fcn`: values of the objective function during iterations

`fcm (data, cluster_n, options)` uses an additional argument variable, `options`, to control clustering parameters, introduce stopping criteria, set the iteration information display, or both.

- `options(1)`: exponent for the partition matrix U (default: 2.0)
options(2): maximum number of iterations (default: 100)

options(3): minimum amount of improvement (default: 1e-5)

options(4): info display during iteration (default: 1)

The clustering process stops when the maximum number of iterations is reached or when the objective function improvement between two consecutive iterations is less than the minimum amount of improvement specified.

The FCM clustering is obtained by minimizing an objective function shown in Equation (4.1).

$$ J = \sum_{i=1}^{n} \sum_{k=1}^{c} \mu_{ik}^{m} |p_i - v_k|^2 $$  \hspace{1cm} (4.1)

where

- $J$ is the objective function
- $n$ is the number of pixels in the image
- $c$ is the number of clusters
- $\mu$ is the fuzzy membership value from table
- $m$ is a fuzziness factor (a value > 1)
- $p_i$ is the $i$th pixel in image
- $v_k$ is the centroid of the $k$th cluster
- $|p_i - v_k|$ is the Euclidean distance between $p_i$ and $v_k$ defined by Equation (4.2).
The calculation of the centroid of the kth cluster is achieved using Equation (4.3).

\[ \nu_k = \frac{\sum_{i=1}^{n} \mu_{ik}^a p_i}{\sum_{i=1}^{n} \mu_{ik}^a} \]  

(4.3)

The fuzzy membership table is calculated using the original Equation (4.4)

\[ \mu_{ik} = \frac{1}{\sum_{i=1}^{c} \left( \frac{p_i - \nu_k}{|p_i - \nu_k|} \right)^{m-1}} \]  

(4.4)

This algorithm has been extended for clustering of color images in the RGB color space. Hence, the computation given in equation (2) to compute the Euclidean distance between the values pi and vk is modified to incorporate RGB colors, and is shown in Equation (4.5).

\[ |p_i - \nu_k| = \sqrt{\sum_{l=1}^{3} (p_{il} - \nu_{kl})^2} \]  

(4.5)

**Pseudo-Code**

As mentioned earlier, this is an iterative process. The pseudo-code is as follows:

- Step 1: Set the number of clusters, the fuzzy parameter (a constant > 1), and the stopping condition
- Step 2: Initialize the fuzzy partition matrix
- Step 3: Set the loop counter \( k = 0 \)
- Step 4: Calculate the cluster centroids, calculate the objective value \( J \)
- Step 5: For each pixel, for each cluster, compute the membership values in the matrix
- Step 6: If the value of \( J \) between consecutive iterations is less than the stopping condition, then stop; otherwise, set \( k = k + 1 \) and go to step 4
  - Step 7: Defuzzification and segmentation

![Figure 4.1 Flowchart for FCM calculation](image-url)
Advantages:

1) Gives best result for overlapped data set and comparatively better than k-means algorithm.

2) Unlike k-means where data point must exclusively belong to one cluster center, here data point is assigned membership to each cluster center as a result of which data point may belong to more than one cluster center.

4.3 PECTORAL MUSCLE EXTRACTION USING AREA CALCULATION

First, the above grayscale FCM three clustering segmented mammogram image is taken, and this applied for first and second clustering value is zero, because of minute point of calcium deposits at the maximum no of given clustered image. After that applied for gray thresholding. And then converted into binary [0, 1] format. Breast region has the largest area. We use the connected component labeling to divide the binary image into different labels. For this reason, we use the area criteria to select the label that represents the breast region and to eliminate the unlikely labels (tape artifacts
and high intensity labels). Finally, to obtain the effective breast region, the result of this step was multiplied with the original mammogram. They are labeled using the bwlabel function in MATLAB.

The ‘Area’ (actual number of pixels in the region) of all objects is calculated using the region props function in MATLAB. The area is less than 1000 value is used to extract the pectoral muscle and finally detect the micro calcifications (or) segmented tumor.

Algorithm

Step 1 : The segmented fcm three clustering is taken. The first and second clustering value is equal to zero.

Step 2 : Then applied for thresholding and then image converted into binary. Then use a connected component labeling to divide the binary image into different labels.

Step 3 : This connected component labels to find out the area using region props function in Matlab.

Step 4 : The area calculation value is less than 1000 should be discarded and finally detects the micro calcifications or segmented tumor.

Level = gray thresh (im1); %equivalent mat lab code

I2 = im2bw (im1, level);

Figure, imshow (I2);

cc = bwconncomp (I2);
Stats = region props (cc, 'Area');

Idx = find (stats. Area < 1000);

I3 = ismember (labelmatrix (cc), idx); and finally detect the micro calcifications or segmented tumor.

4.4 FEATURE EXTRACTION

Feature Extraction Introduction:

Feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification algorithm which over fits the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.
4.4.1 Discrete Wavelet Transforms

Nowadays, wavelets have been used quite frequently in image processing. They have been used for feature extraction, de-noising, compression, face recognition, and image super-resolution. The decomposition of images into different frequency ranges permits the isolation of the frequency components introduced by “intrinsic deformations” or “extrinsic factors” into certain sub bands. This process results in isolating small changes in an image mainly in high frequency sub band images. Hence, discrete wavelet transform (DWT) is a suitable tool to be used for designing a classification system (Cercone 1996).

The 2-D wavelet decomposition of an image is performed by applying 1-D DWT along the rows of the image first, and, then, the results are decomposed along the columns. This operation results in four decomposed sub band images referred to as low–low (LL), low–high (LH), high–low (HL), and high–high (HH). The frequency components of those sub band images cover the frequency components of the original image as shown in Figure 3.

<table>
<thead>
<tr>
<th>LL</th>
<th>HL</th>
</tr>
</thead>
<tbody>
<tr>
<td>LH</td>
<td>HH</td>
</tr>
</tbody>
</table>

Figure 4.3 2D DWT decomposition

*equivalent Matlab code*

```
[ca1, ch1, cv1, cd1]=dwt2 (segmented tumor image,’db1’);
[ca2, ch2, cv2, cd2]=dwt2 (ca1,’db1’);
imshow ([ca2, ch2,
```
cv2, cd2]);

title ('2D DWT');

4.4.2 Statistical features calculation:

There are nine statistical features extracted from the LL band of wavelet transform (Chandrasekhar and Attikiouzel 2000).

Such as,

Mean Standard Deviation Variance Covariance Entropy Energy Kurtosis Area Sum

1. Mean

The mean, \( m \) of the pixel values in the defined window, estimates the value in the image in which central clustering occurs. The mean can be calculated using the formula:

\[
\mu = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} p(i,j)
\]  (4.6)

The Mean block computes the mean of each row or column of the input, along vectors of a specified dimension of the input, or of the entire input.

\%
mean  %equivalent mat lab code
ml = mean (double (ima(:)))

Calculating the mean example

To calculate the mean, you sum up all the numbers in the sample and then divide by the sample size. The sum is 6+9+8+9+2= 34. Since the sample size is 5, the mean is 34/5 = 6.8.
This then is the average of the sample.

2. Standard Deviation

The Standard Deviation, $\sigma$ is the estimate of the mean square deviation of grey pixel value $p(i,j)$ from its mean value.

\[
\text{%standard deviation} \quad \text{%equivalent mat lab code}
\]

\[
SD = \text{std}(\text{std}(\text{double}(ima)))
\]

Calculating the Standard Deviation example:

The standard deviation, like variance, is a measure of the variation of the sample data. The larger the standard deviation, the more random the answers appear. Standard deviation is more popular as a measure than variance.

The method for calculating the standard deviation is different depending on whether we are calculating the variance of a population (everyone) or the variance of a sample (some but not all). The method is the same as variance with one additional step.

Here are the steps

1. Figure out the mean. This is the sum of the numbers given divided by the sample size (i.e. the average).

\[
(6+9+8+9+2)/5 = 34/5 = 6.8
\]

2. Figure out the difference between each number and its mean so that we have:

\[
(6 - 6.8), (9 - 6.8), (8 - 6.8), (9 - 6.8), (2 - 6.8) = -0.8, 2.2, 1.2, 2.2, -4.8
\]
(3) get the square of each difference in step #2 so that we have:

\((-0.8)\times(-0.8), (2.2)\times(2.2), (1.2)\times(1.2), (2.2)\times(2.2), (-4.8)\times(-4.8)\)

= 0.64, 4.84, 1.44, 4.84, 23.04

(4) get the sum of all the squares in step #3 so that we have:

Sum of squares = 0.64 + 4.84 + 1.44 + 4.84 + 23.04 = 34.8

(5) We divide the sum in step #4 by the sample size - 1

34.8/ (5-1) = 34.8/4 = 8.7

6) Last, we take the square root of the value in step #5.

Standard Deviation = sqrt (8.7) = roughly 2.95

3. Variance

Variance is the square root of standard deviation. The variance is a measure of how far a set of numbers is spread out. It is one of several descriptors of a probability distribution, describing how far the numbers lie from the mean (expected value). In particular, the variance is one of the moments of a distribution. In that context, it forms part of a systematic approach to distinguishing between probability distributions. While other such approaches have been developed, those based on moments are advantageous in terms of mathematical and computational simplicity.

%variance %equivalent mat lab code

va = var (double (ima (:)))

Calculating Variance example:

The variance is a measure of the variation of the sample data. The larger the variance, the more random the answers appear. Many people find standard deviation to be a more useful measure of variability.
The method for calculating the variance is different depending on whether we are calculating the variance of a population (everyone) or the variance of a sample (some but not all).

Here are the steps:

1. Figure out the mean. This is the sum of the numbers given divided by the sample size (i.e. the average).
   
   $$(6 + 9 + 8 + 9 + 2) / 5 = 34 / 5 = 6.8$$

2. Figure out the difference between each number and its mean so that we have:
   
   $$(6 - 6.8), (9 - 6.8), (8 - 6.8), (9 - 6.8), (2 - 6.8) = -0.8, 2.2, 1.2, 2.2, -4.8$$

3. Get the square of each difference in step #2 so that we have:
   
   $$(-0.8)^2, (2.2)^2, (1.2)^2, (2.2)^2, (-4.8)^2$$

   $$0.64, 4.84, 1.44, 4.84, 23.04$$

4. Get the sum of all the squares in step #3 so that we have:
   
   $$\text{sum of squares} = 0.64 + 4.84 + 1.44 + 4.84 + 23.04 = 34.8$$

5. Now, for the sample variance, we divide the sum in step #4 by the sample size - 1
   
   $$\text{Variance} = 34.8 / (5-1) = 34.8 / 4 = 8.7$$

4. Covariance

Covariance is a measure of how much two random variables change together. If the greater values of one variable mainly correspond with the greater values of the other variable, and the same holds for the smaller
values, i.e. the variables tend to show similar behavior, the covariance is a positive number. In the opposite case, when the greater values of one variable mainly correspond to the smaller values of the other, i.e. the variables tend to show opposite behavior, the covariance is negative. The sign of the covariance therefore shows the tendency in the linear relationship between the variables.

```matlab
%covariance %equivalent mat lab code
co_v=cov (double (ima));
co_v = sum (sum (co_v))/ (length (co_v)*1000)
```

5. Entropy

Entropy is a statistical measure of randomness that can be used to characterize the texture of the input image. \( E = \text{entropy}(I) \) returns \( E \), a scalar value representing the entropy of grayscale image \( I \). Entropy is a statistical measure of randomness that can be used to characterize the texture of the input image. Entropy is defined as

\[
-\sum (p.*\log2 (p))
\]

where \( p \) contains the histogram counts returned from \text{imhist}

```matlab
%entropy %equivalent mat lab code
Etp = entropy (ima)
```

6. Energy

Energy returns the sum of squared elements in the Grey Level Co-Ocurrence Matrix (GLCM). Energy is also known as uniformity. The range of energy is \([0 1]\). Energy is 1 for a constant image. Energy is also known as
uniformity of ASM (angular second moment) which is the sum of squared elements from the GLCM.

\[ \text{%energy} \quad \text{%equivalent mat lab code} \]

\[ \text{Egy} = \text{sum (h\_norm)} \]

7. Kurtosis

Kurtosis, K measures the peakness or flatness of a distribution relative to a normal distribution. Kurtosis is a measure of how outlier-prone a distribution is. Kurtosis is any measure of the "peakedness" of the probability distribution of a real-valued random variable. In a similar way to the concept of skewness, kurtosis is a descriptor of the shape of a probability distribution and, just as for skewness; there are different ways of quantifying it for a theoretical distribution.

\[ \text{%kurtosis} \quad \text{%Equivalent mat lab code} \]

\[ \text{Ku} = \text{kurtosis (double (ima (:)))} \]

8. Area

The area is a measure of the size of the foreground of the image. The area is the number of pixels in the image. Area is a quantity that expresses the extent of a two-dimensional surface or shape in the plane. Area can be understood as the amount of material with a given thickness that would be necessary to fashion a model of the shape, or the amount of paint necessary to cover the surface with a single coat. It is the two-dimensional analog of the length of a curve (a one-dimensional concept) or the volume of a solid (a three-dimensional concept). The area of a shape can be measured by comparing the shape to squares of a fixed size.
9. Sum

**Summation** is the operation of adding a sequence of numbers; the result is their **sum** or total. If numbers are added sequentially from left to right, any intermediate result is a partial sum, prefix sum, or running total of the summation.

**concatenate the features**

data = [ag;ahg;egy;etp;sd;co_v;m1;va;ku;ar;bag]/100.

After extracting these features of segmented mass/tumor, then the dataset has to be constructed in the proper format, so that it can be given to the standard classifier tools.

**Comparison of Time complexity FCM and FCM with DWT**

The time complexity of FCM is O(ndc2i) and time complexity of HCM is O(ncdi). Now keeping no. of data points constant, let’s assume n=150, d=2, i=10 and varying no. of clusters, we obtain the following table and graph.

**Table 4.1 Time Complexity FCM and FCM with DWT (M S)**

<table>
<thead>
<tr>
<th>Sl.no</th>
<th>Number of cluster</th>
<th>FCM</th>
<th>FCM with DWT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>35</td>
<td>38</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>120</td>
<td>129</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>240</td>
<td>270</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>300</td>
<td>320</td>
</tr>
</tbody>
</table>
Fig 4.3 Time Complexity FCM and FCM with DWT (M S)