Chapter 3

Segmentation based on threshold selection

3.1 Introduction on threshold selection from multi-modal histogram

Previous chapter explained the developed algorithm "segmentation by down sample and up sample approach", which is based on the idea of evaluating difference between original and up sampled images. Initially, the original image is down-sampled and then up-sampled. It has been noticed that the up-sampled image contains only interference features. Finally, when the interference image is subtracted from original raw image, it results with significantly reduced amount of unwanted features. By this method, the targets, when they are in distinguishably small in size are extracted. From the result it has been observed that down sample and up sample approach is a spatial approach and is suitable only when the target size are relatively small. In target detection scenario
the target size may vary. This chapter presents a threshold selection algorithm using Gaussian curve fit on histogram of SAR image.

Creation of a two dimensional maps by segregating cohesive pixels with similar values into individual sets is called segmentation in image processing. Many different methodologies for threshold selection have been developed and many are still in the process of development [72][87]. In histogram threshold one always tries to find out an optimal point by which one can separate out background and object information from a given image. Bi level thresholding is used if an image contains only one object which can be distinguished from background. The gray level histogram of these images will always be a bimodal distribution. If an image contains multiple objects with different size then the gray level histogram of this type of images composed with several distribution. Several multi-modal threshold techniques[45][74] has been proposed to segment multi-modal images. The major difficulty for multi-modal thresholding technique is that all previously proposed algorithm [8] tries to smooth image histogram before finding the threshold point. Smoothening of the histogram can remove the details of small objects. Threshold selection is a crucial problem when the objects were distinguishingly small in size on a given image. It has been seen that histogram of an image contains some valleys, and each valley point corresponds to a region. In this chapter a novel algorithm is proposed to find these valley points.It is presumed that the frequency distribution of a series of consecutive pixel values which can be referred as an object may behave like a Gaussian distribution[59].

This chapter describes a threshold selection technique for clutter rejection. In this technique, two algorithms have been developed. First of all, a method is described for partitioning a multi-modal histogram into a sequence of Gaussian functions. One single Gaussian function may not be adequate to approximate the whole data set and hence
the data set is decomposed into some intervals and in each interval a Gaussian function may be fitted. The minimum RMSE value for Gaussian distribution fit has been used for creation of the first interval. Subsequent Gaussian fits approximate the subsequent intervals in a sequential manner. Thus, a number of Gaussian functions are required to approximate the entire range. Points of inflection for each gaussian fit are considered as threshold point for that corresponding interval. A union of these threshold images has been taken to get a single threshold image.

Secondly, a new method is adopted to extract target objects from the threshold image. For this, a ratio is computed of the longest to shortest dimensions for different targets, present in the binary image. This scheme significantly removes unwanted objects and detects the targets.

This scheme is effective in surveillance, where the object-histogram may vary because of the terrain (surrounding) and target-size may differ because of distance or shape of the target. Experimental results demonstrate the effectiveness of the proposed scheme.

### 3.2 Methodology [37]

Automated surveillance puts emphasis on finding the objects of interest. Hence, segmentation of the image into cohesive pixels, i.e., similar values into individual sets needs to be detected. Many different methodologies have been developed and many are still in the process of development. For threshold selection, the methods currently in use are the co-occurrence matrix [41] and entropy models [72] with or without graph theoretic models [87] and the generative model for Gabor wavelets [97]. It is evident that these processes do not have a simple matrix operator solution.

One of the fundamental problems of segmentation (geometric partition of a matrix) is
to know when it can correctly infer the number of regions. That is their exclusion and inclusion conditions. Through histogram threshold, one always tries to find out an optimal point through which one can separate out background and target information from a given image. Intuitively, this could be related to the sampling properties of the matrix cells. Sampling grid will have optimum pixel radius. In our technique, an assumption has been made that all segmented regions can be fitted with a Gaussian function. Threshold selection is a crucial problem when the targets were totally dominated by clutter. It has been seen that histogram of an image contains some valleys, and each valley point corresponds to a region.

Bi level thresholding is used if an image contains only one object that can be distinguished from the background. The gray level histogram of this kind of images will always have a bimodal distribution. If an image contains multiple objects with different sizes, then the gray level histogram of this kind of images is composed of several distributions. Threshold selection is a crucial problem when the targets were totally dominated by clutter. It is seen that the histogram of such an image contains some valleys, and each valley point corresponds to some region(s).

This chapter describes a new and novel algorithm to find the valley points. The proposed method incorporates a thresholding procedure which delineates the clutter and extracts target information from the threshold image. This procedure is a combination of three algorithms, consisting of partitioning the histogram into intervals that can be fitted with a Gaussian function, finding the point of inflection in each of these intervals (if it appears in the interval) and then adding them together to show all the segmented objects.

The whole procedure is based on the assumption that point distribution in a matrix is a Poisson distribution. This is valid if points are truly random[50], but this is not true for all images. If one plots histogram for the frequency distribution of pixel values from a
given image, then this is not necessarily a Poisson. So, in this thesis it has been tried to capture the trend of data across the entire range and see if a Gaussian distribution is possible. To increase accuracy and optimal threshold approximation on smaller intervals has been considered. If the intervals are divided into smaller intervals then it may separate the sets of pixel values which are in Gaussian distribution. It is presumed that the frequency distribution of a series of consecutive pixel values which can be referred to as a target, may behave like a Gaussian distribution[67][101].

The three algorithms in the procedure are algorithm 1, algorithm 2 and algorithm 3. Algorithm 1 creates small non-overlapping intervals or disjoint partitions from the multi-modal histogram. To extract threshold point from small interval, Gaussian curve is made to fit in each interval and the Point of inflections evaluated for a specified interval is considered as a threshold parameter of that interval defined in algorithm 2. Algorithm 3 describes how all the threshold images extracted from each interval are finally integrated by using the union concept of set theory.

3.2.1 Creating intervals from entire histogram

Algorithm 1 describes the procedure for creating intervals. The Gaussian function of the form has been used;

$$f(x) = a_n e^{\left(-\frac{x-b_n}{c_n}\right)^2}$$  \hspace{1cm} (3.1)

A trial has been made to fit the image histogram to this Gaussian function by varying parameters $a_n$, $b_n$ and $c_n$ such that it has least root mean square error (RMSE). The developed algorithm tries to find the best fit which minimizes the RMSE error. The first interval is the data set from gray value 1 to the point (say) $a$, where RMSE is less than 10. Second interval is the data set from the point $a$ to the point (say) $b$, where RMSE
is less than 10. Recursively the algorithm selects the remaining data set and evaluates
RMSE error till the point the error is less than 10. The iteration continues until it reaches
the maximum pixel value. The RMSE for each interval has been calculated by using the
following equations;

$$ RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} |f(x_k) - y_k|^2} $$  \hspace{1cm} (3.2)

Where N = number of existing data points; \( f(x_k) \) = result obtained by using (Eq1) ; \( y_k \) =
Actual data for the available data points of \( x \).

Once the intervals have been determined they would be disjoint subsets, i.e., the entire
histogram has been partitioned into 1,...., n non overlapping intervals. This can be sym-
bolically represented as; let \( V \) is a \( X \times Y \) dimensional Vector space and \( U \) is defined
as the partition of the user set with n disjoint subsets and \( V = U_1 \cup U_2 \cup ..U_n \) Where
\( U_i \cap U_j = 0 \).

The basis for the first part of the Gaussian fit is based on following premise; if the
coordinates be represented by \( \Theta \) an arbitrary set \( X \times Y \), and if an image \( I = P_{xy} | (x,y) \in \Theta \)
then \( I \) is indexed by \( \Theta \). The constraints in a gray scale image are a set integers from
0 to255. If \( i_n \) is a subset whose values are within \( \varepsilon \) distance from a threshold value \( \tau \)
and \( \varepsilon \neq 0 \) then one needs to find all \( P_{xy} \) lying within \( \tau - \varepsilon \) and \( \tau + \varepsilon \) such that they are
topologically connected that is find the constraints (range) for \( x, y \).

If the partition \( \theta \) is homogeneous, then it will follow a Gaussian process[56]. Using this,
one can say that an image space may be segmented by looking at histogram and finding
segments of adjacent pixel values that have a Gaussian distribution. This distribution
would not behave like a single peak Gaussian distribution. There would be peaks and
valleys.

Sen and Pal [85] uses the premise that bins of a histogram representing the smallest
and largest gray level value which contain the brightest and darkest regions respectively. They tried to carry out a multilevel thresholding by using two seed values. Each region node was obtained at a particular depth. Further they separated objects using the proposed bi-level thresholding method to get the regions at the next higher depth. Here in this approach bi-level thresholding is not considered because the object may be either darker or lighter than the background. In case, there are multiple objects some may be darker and some other may be lighter than the background. This suggests to select threshold through formation of different intervals.

### 3.2.2 Extraction of threshold parameters

Algorithm 2 explains the creation of threshold images from the point of inflection in that specified interval. To find the point of inflection for an individual interval, Gaussian curve is fitted to the individual data set, obtained after applying algorithm 1. The point of inflection point is considered as the threshold parameter for that particular interval. The point of inflection is the point where \((\partial y/\partial x) = 0\) and \((\partial^2 y/\partial x^2) \geq 0\) in each Gaussian fit. Each individual interval is converted into binary image obtained with point of inflection as the threshold parameter. To receive one single threshold image, all the threshold images, extracted from each interval, are integrated.
Algorithm 1 Algorithm for creating intervals

1: \( m \leftarrow \) maximum gray value;
2: \( S \leftarrow \) dataset;
3: for all dataset do
4: \( 1 \) to \( m \leftarrow S; \)
5: \( f \leftarrow \) frequency distribution from \( 1 \) to \( m; \)
6: Plot histogram for \( S \) and \( f; \)
7: fit curve by using Eq. 1;
8: display coeffs a,b,c;
9: if RMSE \( \leq 10 \) then
10: print datapoint p;
11: print RMSE;
12: end if
13: end for
14: for \( 1 \) to \( p \leftarrow S \) do
15: \( f1 \leftarrow \) frequency distribution from \( 1 \) to \( p; \)
16: plot histogram for \( S \) and \( f1; \)
17: fit curve by Eq 1;
18: display Coeffs \( a_1, b_1, c_1; \)
19: display RMSE;
20: if \( (\partial y/\partial x) = 0 \) and \( (\partial^2 y/\partial x^2) > 0 \) then
21: print Data points;
22: else
23: print point of inflection not found;
24: end if
25: print \( 1 \) to \( p \leftarrow \) first interval;
26: end for
27: return \( S \leftarrow p \) to \( m; \)

3.2.3 Integrate the Threshold images

The integration of all the threshold images (\( T_1, T_2, \ldots, T_n \)) has been done by using the union concept of sets. Algorithm 3 explains the steps for creating integrated image. Let the threshold images are \( T_1, T_2, T_3, \ldots, T_n \) then integrated image \( I(0,1) = T_1 \cup T_2 \cup \ldots \cup T_n. \)

Algorithm 3 Algorithm for integrating threshold images

1: INPUT:Threshold Images \( T_1, T_2, \ldots, T_6 \)
2: OUTPUT: Integrated Threshold Image I
3: for \( i \leftarrow 1T6 \) do
4: \( I = T_1 \cup T_2 \cup T_3 \cup T_4 \cup T_5 \cup T_6; \)
5: end for
Algorithm 2 Algorithm for creating threshold images

1: \( t \leftarrow \) Point of inflection;
2: \( g \leftarrow \) Gray level for image matrix;
3: for image matrix do
4: \( \text{if} \ (g \geq t) \ \text{then} \)
5: \( g = 1 \)
6: \( \text{else} \)
7: \( g = 0 \)
8: \( \text{end if} \)
9: end for

3.3 Results and discussions

3.3.1 Results with SAR images

To justify the merit of the algorithm, several experiments have been carried out with both MSTAR and optical images. The cluttered dominated SAR images have multi-modal histogram. Fig. 3.1 contains the segmented images after ‘Gaussian threshold selection’ algorithm. This Fig. 3.1 includes the targets BTR 60, 2S1 and D7. All these images contain target shadow and hence their threshold selection is difficult. Sometime, shadow can create a confusion between target and non-target components classification. From the output, it is seen that target is properly extracted, while shadow part is significantly rejected. In Fig. 3.2, the targets displayed are BRDM_2, T62 and SLICY images. For all these images, the histograms are multi-modal. The result shows the effectiveness of the threshold selection algorithm.
Figure 3.1: Segmentation with 'Gaussian threshold selection' algorithm on SAR data
Figure 3.2: Segmentation with 'Gaussian threshold selection' algorithm on SAR data
3.3.2 Results with optical images for BSDS500 dataset

Fig. 5.8 contains the segmentation result for 'Gaussian threshold selection’ on Berkeley Segmentation Data Set (BSDS500) images.

Figure 3.3: Segmentation using 'Gaussian threshold selection’ algorithm on BSDS500 images

Figure 3.3: Segmentation using 'Gaussian threshold selection’ algorithm on BSDS500 images
3.3.3 Results with optical images captured by high resolution camera

The algorithm has been initially applied to MSTAR and BSDS500 images. The algorithm is also tested on optical images. The images obtained by cameras are grouped into categories depending on the size of objects, nature of background, and the difference between the modal pixel value of the background and the modal pixel value of the object. The following are the categories of images considered for experiment:

1. Images with noisy background when there is a single object present.

2. Single object present in an environment where intensity difference between object and background is significantly less.

3. Images with objects of different sizes in the cluttered background.
Figure 3.4: Segmentation using ‘Gaussian threshold selection’ algorithm on high resolution camera captured images [37]

Fig. 3.4 depicts the optical images captured by digital camera. These images are of very high resolution. In image category 1, histogram describes a very large peak followed by a small peak at the edge. Images having such histograms are not amenable to Otsu’s [71] and another multimodal [44] method of thresholding. In image category 2, multiple
points of inflections are found to be present in the data set. Points of inflections helps to depict the object region. Image category 3 points to variable sized objects in cluttered background and describes the most complex situation. One can visualize its histogram where the peaks are concentrated on small area. It is very difficult to partition this kind of interval in disjoint sets. The final threshold image is found to have segmented small object-like structure.

Table 3.1, 3.2 and 3.3 contains the results corresponding to points of inflections and their intervals for image category 1, 2 and 3 respectively. Table 3.1 contains information for the intervals and their corresponding data set for image category 1 (Images with noisy background when a single object is present). For this image, first interval extracted by the algorithm extends from the pixel value from 1 to 9 and the computed RMSE is 1.412. Points of inflections for this data set are 7, 8 and 9. It can be said that in this region objects are present with gray levels 7, 8 and 9. So for this interval the points 7, 8 and 9 can be visualized as threshold points. The next interval is from 9 to 17. Points of inflections at this interval are 10 and 17, which depicts object information is available at gray level 10 and 17. For the interval 17 to 22, there is no point of inflection found. So the algorithm ignores this interval. The points of inflections are found in the interval with gray level 80 to 254. All these gray level points are considered as threshold points for category 1 image. The output for this image is shown in Fig. 3.4.

Table 3.2 contains the information for points of inflections of category 2 image (Single object present in an environment where intensity difference between object and background is significantly less). The data sets selected after multiple Gaussian-fit-algorithm are shown in the given table. Points of inflections are considered as threshold points for the given image. The output is shown in Fig. 3.4.
Table 3.1: For first category: Intervals, RMSE and point of inflection for threshold:

<table>
<thead>
<tr>
<th>Intervals</th>
<th>DataSet</th>
<th>RMSE</th>
<th>Point of inflection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 to 9</td>
<td>1.412</td>
<td>7,8,9</td>
</tr>
<tr>
<td>2</td>
<td>9 to 17</td>
<td>2.808</td>
<td>10,17</td>
</tr>
<tr>
<td>3</td>
<td>17 to 22</td>
<td>0.69</td>
<td>Not found</td>
</tr>
<tr>
<td>4</td>
<td>22 to 30</td>
<td>8.16</td>
<td>22,23</td>
</tr>
<tr>
<td>5</td>
<td>30 to 37</td>
<td>10.3</td>
<td>Not found</td>
</tr>
<tr>
<td>6</td>
<td>37 to 43</td>
<td>10.05</td>
<td>Not found</td>
</tr>
<tr>
<td>7</td>
<td>43 to 51</td>
<td>20.78</td>
<td>Not found</td>
</tr>
<tr>
<td>8</td>
<td>51 to 57</td>
<td>19.8737</td>
<td>Not found</td>
</tr>
<tr>
<td>9</td>
<td>57 to 63</td>
<td>26.9465</td>
<td>Not found</td>
</tr>
<tr>
<td>10</td>
<td>63 to 70</td>
<td>30.27</td>
<td>Not found</td>
</tr>
<tr>
<td>11</td>
<td>70 to 80</td>
<td>47.33</td>
<td>Not found</td>
</tr>
<tr>
<td>12</td>
<td>80 to 254</td>
<td>41.8638</td>
<td>200, 218, 236, 254</td>
</tr>
</tbody>
</table>

Table 3.3 includes data sets and their corresponding points of inflections for the image category 3 (Images containing variable sized objects in cluttered background). The result in Fig. 3.4 shows the robustness of the developed algorithm.

### 3.3.4 Ranges where point of inflections could not be found

There are few areas where histogram could not be fitted in the trial images which indicates the possibility for non-existence of object in this area.
Table 3.2: For second category: Intervals, RMSE and point of inflection for threshold:

<table>
<thead>
<tr>
<th>Intervals</th>
<th>DataSet</th>
<th>RMSE</th>
<th>Point of inflection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 to 106</td>
<td>31.01</td>
<td>9,10,88,95,100</td>
</tr>
<tr>
<td>2</td>
<td>106 to 143</td>
<td>166.6</td>
<td>131,135</td>
</tr>
<tr>
<td>3</td>
<td>143 to 152</td>
<td>472.6</td>
<td>147,151</td>
</tr>
<tr>
<td>4</td>
<td>152 to 160</td>
<td>61.28</td>
<td>152,156</td>
</tr>
<tr>
<td>5</td>
<td>160 to 172</td>
<td>39.12</td>
<td>Not found</td>
</tr>
<tr>
<td>6</td>
<td>172 to 180</td>
<td>210.9</td>
<td>173,179</td>
</tr>
<tr>
<td>7</td>
<td>180 to 227</td>
<td>44.57</td>
<td>Not found</td>
</tr>
<tr>
<td>8</td>
<td>227 to 234</td>
<td>157.5</td>
<td>230</td>
</tr>
<tr>
<td>9</td>
<td>234 to 240</td>
<td>20</td>
<td>238</td>
</tr>
<tr>
<td>10</td>
<td>240 to 254</td>
<td>31.77</td>
<td>241</td>
</tr>
</tbody>
</table>

Table 3.3: For third category: Intervals, RMSE and point of inflection for threshold:

<table>
<thead>
<tr>
<th>Intervals</th>
<th>DataSet</th>
<th>RMSE</th>
<th>Point of inflection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 to 24</td>
<td>0.9692</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>24 to 30</td>
<td>0.441</td>
<td>24,27</td>
</tr>
<tr>
<td>3</td>
<td>30 to 45</td>
<td>0.9635</td>
<td>30,37,41</td>
</tr>
<tr>
<td>4</td>
<td>45 to 75</td>
<td>56.64</td>
<td>45,50</td>
</tr>
<tr>
<td>5</td>
<td>75 to 82</td>
<td>22.94</td>
<td>Not found</td>
</tr>
<tr>
<td>6</td>
<td>82 to 92</td>
<td>307.21</td>
<td>85</td>
</tr>
<tr>
<td>7</td>
<td>92 to 204</td>
<td>878.16</td>
<td>128,139,150,160,182,193,200</td>
</tr>
</tbody>
</table>

3.4 Comparison with other multimodal threshold algorithm

A comparison has been done with one of the multi-modal histogram threshold selection methods [7]. Chang et al. [7] proposed a method for evaluating threshold by using statistical decision theory for a multi-modal histogram. In their work, a smoothing factor by
estimating Gaussian kernel on gray level image histogram was proposed. Smoothed histogram was decomposed into a few non-overlapping clusters by measuring their skewness. Unfortunately, smoothness of histogram can damage small peaks and valleys, and hence can destroy details for small objects from an image. The present ‘Gaussian threshold selection’ algorithm takes care of this important phenomenon. Creation of intervals, in our algorithm, is done automatically through evaluation of RMSE and subsequently, thresholds are determined in each interval. The final image is an integration of all threshold images. All the three categories of images have also been tested with the histogram smoothing algorithm of Chang et al. It is observed that our ‘Gaussian threshold selection’ algorithm detects and retains the small objects under the clutter afflicted environment. The comparison result is shown in Fig.3.5.

Figure 3.5: Comparison with another multi-modal histogram threshold and the proposed method
Table 3.4: Running time comparison with GMM

<table>
<thead>
<tr>
<th>Category</th>
<th>GMM [Time (ms)]</th>
<th>Proposed method [Time (ms)]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category 1 image</td>
<td>5.0333</td>
<td>1.3533</td>
</tr>
<tr>
<td>Category 2 image</td>
<td>3.0221</td>
<td>1.0631</td>
</tr>
<tr>
<td>Category 3 image</td>
<td>15.0121</td>
<td>5.1504</td>
</tr>
</tbody>
</table>

Another comparison analysis is done with Gaussian Mixture Model (GMM) which is based on Expectation Maximization (EM) algorithm [46]. GMM model selects global threshold using EM. Where as Gaussian threshold selection algorithm evaluate threshold points locally. The experiment are performed on the personal computer with 2.6 GHz AMD CPU and 2Gbytes of memory. This computer runs on windows XP with MATLAB 7.10 installed. Table 3.4 lists the running time evaluated for GMM and Gaussian threshold selection algorithm in three categories of optical images. From the table it is very clear that the proposed gaussian threshold selection algorithm is much faster than GMM.

### 3.5 Conclusion

This chapter discusses a novel ‘Gaussian threshold selection’ algorithm. The histogram for the frequency distribution of gray-levels in an image belongs to a finite set of families of distribution. This algorithm finds several disjoint intervals of gray-level histogram by fitting Gaussian curve on small intervals. The points of inflection evaluated, in an interval, are considered as the threshold parameters for that interval. Finally, all the threshold images are integrated to get one single threshold image through a union operator. Strength of the proposed method lies in identification of objects of different sizes, specially, when the objects are in variable sizes in the whole image space. Comparison analysis shows the running time efficiency of the proposed method.