# A Survey of Image Thresholding Techniques

Bernard Lampe, *Member, IEEE*

*Abstract*—In this paper, we survey two different approaches to non-parametric, gray-level image thresholding. The first approach computes a threshold as informed by the one dimensional histogram. This approach leverages the gray-level values solely to estimate an optimal threshold. We exemplified this approach by presenting results from three different algorithms which leverage the histogram in different ways. The second approach computes a threshold as informed by the gray-level values and the spatial correlations of those values in the image. This is accomplished by forming a two dimensional histogram which encodes the gray-levels as well as the spatial co-occurrences of those levels. As in the first approach, we present results of three different algorithms which take advantage of the spatial relationship of the pixels as represented by a two-dimensional histogram. Finally, this paper presents empirical results on a variety of different data sets. These data sets include portraits, natural scenes, medical and space-borne images. The goal is to give the reader an understanding of which techniques perform well in different imaging domains.

*Index Terms*—Thresholding, Histogram, Co-occurrence Matrix

## I. INTRODUCTION

NON-PARAMETRIC thresholding of gray scale images is often an early step in image processing, analysis and understanding. Frequently, it is employed in image segmentation and object detection algorithms. The goal of thresholding algorithms is to assign each pixel of the image into one of two groups, foreground or background. The definition of foreground and background is domain specific, but in the general case, a white object on a black background will have a bi-model histogram. Any gray level between the modes of the histogram will obviously assign pixels of the object to the foreground and all others to the background. In practice, this simple example of a white object on a black background is not useful as most scenes contain objects which contain coincident gray scale values with the background or interact with the foreground and background. Shadows produced by the object and partially obscured objects are examples of these interactions. In these cases, finding the valley between the histogram modes may not be clear or computationally stable. Clearly, a more discriminating technique will be needed to choose an appropriate threshold which will minimize the error rate of pixel assignment. This analysis examines six such discriminating techniques which leverage different properties of the gray levels and spatial information besides the histogram shape.

### A. Utilizing the One Dimensional Histogram

The process of thresholding can be viewed as assigning pixels to one of two groups as described above. Therefore, many thresholding algorithms use the magnitude of the pixels themselves to compute an appropriate decision threshold. Here we purpose three different algorithms from existing literature which compute thresholds utilizing the pixel grayscale values. Each algorithm examined leverages the gray levels by first computing the one dimensional histogram.

First off, we examine the 1-bit Lloyd-Max quantizing algorithm [5] [6]. The Lloyd-Max algorithm is a iterative, lossy, compression algorithm which aims to produce the best signal in a given number of bits. The criteria for best output signal was purposed by Max [6] as a signal which minimizes the mean squared error (MSE) between the input and output probability mass functions (PMF) in the discrete case. This algorithm can be used to perform thresholding by compressing the pixels dynamic range from 8-bits to 1-bit, thereby attaining an image containing only two gray levels. The foreground can be assigned to the group which has the larger gray value label. This algorithm operates on the image histogram and converges to a solution which produces two groups which describe the histogram most accurately in a MSE sense. For example, if the histogram was bi-model, each mode would be in a separate group, and the group labels would be the grayscale values associated with the histogram modes. Another illustrative characterization of the 1-bit Lloyd-Max algorithm is that the pixels are quantized into one of two groups. Each group will have an equal area, or equal number of pixels, under the one dimensional function, the histogram, and the quantized labels will be the modes of each group. The iterative algorithm is described as follows below.

1) Compute an estimate of PMF using the histogram. $p_i = n_i/N$ where $n_i$ is the number of pixels with gray level $i$ and $N$ is the total number of pixels.
2) Divide the 8-bit grayscale dynamic range [0-255] into two equal sets by choosing a threshold $k = 128$.
3) For each set compute a representative value of each set $y_i$ as the centroid of the set members as in equation 1.
4) Compute a new threshold as the mid-point of the representative values of each set as in equation 2.
5) For each set compute the distortion metric between the original PMF and the newly quantized PMF as in equation 3.
6) Repeat steps 3, 4 and 5 until the distortion measure no longer decreases.
7) Applying the quantized sets will result in a two level image.

$$y_i = \frac{\sum i p_i}{\sum p_i} \qquad (1)$$

$$k = \frac{y_{i+1} - y_i}{2} \qquad (2)$$

$$MSE_i = \sum (x_i - y_i)^2 p_i \qquad (3)$$

Another, very prolific, thresholding method based on the gray-level values is by Nobuyuki Otsu [7]. This method assumes that within the distribution of gray level values there lies two classes which can be separated by an optimal threshold. The optimal threshold is the one which minimizes the within-class variance and maximizes between-class variance [7]. Intuitively, we can say that the algorithm will choose the threshold which produces two classes which are as "tight" as possible and as "far" away from each other as possible. The algorithm proceeds by exhaustively searching all the possible thresholds [0-255] and at each hypothesized threshold, computes the objective function derived by Otsu and re-stated here in equation 6. Finally, the threshold is chosen as the maximum of the objective function. The algorithm is described as follows.

1) Compute an estimate of PMF using the histogram. $p_i = n_i/N$ where $n_i$ is the number of pixels with gray level $i$ and $N$ is the total number of pixels.
2) For each possible threshold $k$ compute the value of the objective function as in equation 6.
3) The optimal threshold $k^*$ is the max of the objective function as in equation 7.

$$\mu(k) = \sum_{i=1}^{k} i p_i \tag{4}$$

$$\omega(k) = \sum_{i=1}^{k} p_i \tag{5}$$

$$\sigma^2(k) = \frac{[\mu(255)\omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \tag{6}$$

$$\sigma^2(k^*) = \max_{0 < k < 255} \sigma^2(k) \tag{7}$$

The next thresholding algorithm also based on the one dimensional histogram is by T. Pun [9]. This method makes use of the definition of Shannon entropy of the histogram to compute an anisotropy coefficient. This coefficient is the percentage of the information which is present in the first half of the image pixels. This coefficient is used to threshold the image by assigning the gray-levels containing half the information to the background and the other pixels to the foreground. The goal is to choose a threshold such that each class of pixel contains as much of the image information as possible. The algorithm is described as follows.

1) Compute an estimate of PMF using the histogram. $p_i = n_i/N$ where $n_i$ is the number of pixels with gray level $i$ and $N$ is the total number of pixels.
2) Compute the anisotropy coefficient $\alpha$ as in equations 8 and 9.
3) The threshold $k$ is computed as in equation 10 below.

$$\sum_{i=0}^{na} p_i \approx \frac{1}{2} \tag{8}$$

$$\alpha = \sum_{i=0}^{na} p_i log_2(p_i) / \sum_{i=0}^{255} p_i log_2(p_i) \tag{9}$$

$$\sum_{i=0}^{k} p_i = \frac{1}{2} + |\frac{1}{2} - \alpha| \tag{10}$$

### B. Utilizing the Two Dimensional Histogram

In addition to the gray-levels of the pixels, images contain considerable information in the spatial correlation of adjacent pixels. This information can be leveraged to further improve the discriminating power of image thresholding algorithms. This information is especially relevant when the spatial interactions become more dominant than the gray-levels [1]. For example, images that contain large amounts of noise and a low signal-to-noise ratio (SNR) exhibit considerable texture which can be exploited by examining the spatial correlations. Here we purpose three different algorithms from existing literature which compute thresholds from the pixel grayscale values and spatial information. Each algorithm examined represents the spatial information via co-occurrence relationships which count the number of times gray levels occur in adjacent pixels.

One of the earliest successful attempts to harness the spatial information of grayscale images in thresholding algorithms was developed by Abutaleb [1]. Abutaleb's method computes a two dimensional co-occurrence histogram/matrix. The histogram counts are formed by the tuples of pixel gray-levels and the average of the adjacent pixel gray levels comprising the 8-connected neighborhood. The co-occurrence matrix captures the coincidences of pixel patterns, such as those seen in textures. This matrix is then normalized to form an estimate of the two dimensional probability mass function (PMF) of the two discrete random variables. After computation of the co-occurrence matrix, the threshold is computed by exhaustively searching over all the threshold tuples in the two dimensions and computing the maximum of the entropic objective function. This objective function is the sum of the entropy of the hypothesized foreground and background and is re-stated here in equations 11 and 14. By maximizing the objective function the threshold seeks to find a bi-level image which preserves the maximum amount of information in each class of pixels. The algorithm is described below.

1) Compute an estimate of joint PMF using the two dimensional histogram comprised of counts of the tuples of pixel gray levels and neighborhood averages. $p_{ij} = f_{ij}/N$ where $f_{ij}$ is the number of pixels with gray level $i$ and neighborhood average $j$. Also $N$ is the total number of pixels.
2) Exhaustively search all tuples of possible thresholds $(l, k)$. For each possible threshold tuple, divide the estimated joint PMF into four orthogonal regions as depicted in figure 1 and evaluate the entropic objective function as in equation 14. Due to the assumption of pixel spatial correlation, two of the four regions of the PMF will be very sparse and only two will be considered in the objective function.
3) The optimal threshold will be the $(l, k)$ that are associated with the largest value of $\psi$. Applying this threshold tuple to the histogram will produce a two level

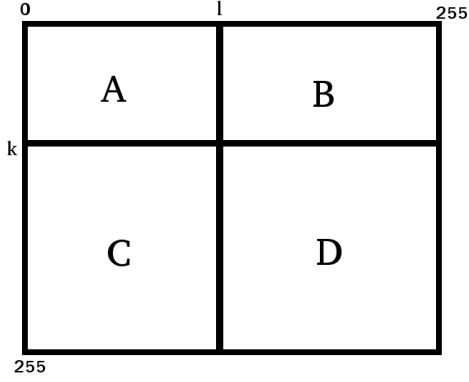Fig. 1: Quandrants of Abutaleb's Co-occurrence Matrix at Threshold (l,k)



Fig. 2: Quandrants of Pal's Co-occurrence Matrix at Threshold k

image where pixel falling in region D are classified as foreground pixels.

$$\psi(l,k) = H(A) + H(D) \tag{11}$$

$$P_{lk} = \sum_{i=1}^{l}\sum_{j=1}^{k} p_{ij} \tag{12}$$

$$H_{lk} = -\sum_{i=1}^{l}\sum_{j=1}^{k} p_{ij} * log_2(p_{ij}) \tag{13}$$

$$\psi(l,k) = log_2(P_{lk}(1-P_{lk})) + H_{lk}/P_{lk} + (H_{mm} - H_{lk})/(1-P_{lk}) \tag{14}$$

Another image thresholding algorithm that leverages the spatial information was developed by Pal-Pal [8]. This algorithm computes the two dimensional co-occurrence histogram/matrix similar to Abutaleb's method. The histogram counts are formed by the tuples of pixel gray levels and the gray levels of the adjacent pixels in the 4-connected neighborhood. In practice, only two of the 4-connected neighborhood pixels are used to compute the histogram/matrix. This speeds computation and has been shown to have negligible effect on the final threshold chosen [2]. After computing the co-occurrence matrix, a single threshold is exhaustively searched and two objective functions are maximized. The threshold associated with the maximum of either objective function may be used for final thresholding. Both objective functions are the sum of entropies of the segmented co-occurrence matrix. The local entropy is defined as the sum of entropies computed from the regions of the thresholded co-occurrence matrix. These regions are the transitions within the hypothesized background and foreground. The joint entropy is defined as the sum of the entropies computed from the other regions of the thresholded co-occurrence matrix. These regions are the transitions between the foreground and background. The algorithm is described below.
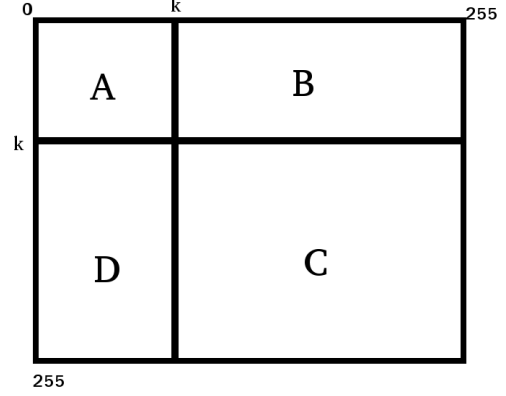
1) Compute an estimate of joint PMF using the two dimensional histogram comprised of counts of the tuples of pixel gray levels and neighborhood gray levels. $p_{ij} = f_{ij}/N$ where $f_{ij}$ is the number of pixels with gray level $i$ and neighborhood gray level $j$. Also $N$ is the total number of pixels.
2) Exhaustively search each threshold $k$. For each threshold, divide the joint PMF into four orthogonal regions as depicted in figure 2 and evaluate the two objective functions for joint entropy and local entropy re-stated in equations 19 and 20.
3) The optimal threshold will be the $k$ associated with the max of either objective function. Applying the local entropy threshold will give a two level image that discriminate between foreground and background. Using the joint entropy threshold will result in identifying pixels which are transitions between foreground and background.

$$p_{ij}^A = p_{ij}/P_A(k), \quad P_A(k) = \sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij} \tag{15}$$

$$p_{ij}^B = p_{ij}/P_B(k), \quad P_B(k) = \sum_{i=0}^{k}\sum_{j=k+1}^{255} p_{ij} \tag{16}$$

$$p_{ij}^C = p_{ij}/P_C(k), \quad P_C(k) = \sum_{i=k+1}^{255}\sum_{j=k+1}^{255} p_{ij} \tag{17}$$

$$p_{ij}^D = p_{ij}/P_D(k), \quad P_D(k) = \sum_{i=k+1}^{255}\sum_{j=0}^{k} p_{ij} \tag{18}$$

$$H_{joint}^{(2)} = -\sum_{i=0}^{k}\sum_{j=k+1}^{255} p_{ij}^B log_2(p_{ij}^B) - \sum_{i=k+1}^{255}\sum_{j=0}^{k} p_{ij}^D log_2(p_{ij}^D) \tag{19}$$

$$H_{local}^{(2)} = -\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}^A log_2(p_{ij}^A) - \sum_{i=k+1}^{255}\sum_{j=k+1}^{255} p_{ij}^C log_2(p_{ij}^C) \tag{20}$$

The last image thresholding algorithm examined also leverages the spatial information of the co-occurrence matrix and was developed by Chang et. al. [2]. The two dimensional co-occurrence histogram/matrix is computed as in the algorithm by Pal-Pal above. In addition, an exhaustive search of the possible image thresholds is conducted and a new objective function is maximized. The new objective function takes advantage of the entropy of all the segmented regions of the co-occurrence matrix to compute the relative entropy between the original image and the thresholded image. By using the relative entropy as an objective function, the algorithm computes the bi-level image which best represents the input image. The computation of the co-occurrence matrix and threshold searching are identical to the Pal-Pal algorithm above so only the new objective function will be given here in equation 25. In addition the region probabilities, such as $P_A(k)$ are computed as defined above in equations 15, 16, 17 and 18.



Fig. 3: Lena Original Image

$$p_{ij}^{'(A)}(k) = q_A(k) = \frac{P_A(k)}{(k+1)(k+1)} \tag{21}$$

$$p_{ij}^{'(B)}(k) = q_B(k) = \frac{P_B(k)}{(k+1)(255-k)} \tag{22}$$

$$p_{ij}^{'(C)}(k) = q_C(k) = \frac{P_C(k)}{(255-k)(255-k)} \tag{23}$$

$$p_{ij}^{'(D)}(k) = q_D(k) = \frac{P_D(k)}{(255-k)(k+1)} \tag{24}$$

$$\max L(p; p') = \sum_{i=0}^{255} \sum_{j=0}^{255} p_{ij} log_2(p'_{ij}) =$$
$$P_A(k)log_2(q_A(k)) + P_B(k)log_2(q_B(k)) +$$
$$P_C(k)log_2(q_C(k)) + P_D(k)log_2(q_D(k)) \tag{25}$$

## II. EXPERIMENTS

### A. Image Data Set

Images from different modalities were chosen to highlight the effectiveness of each thresholding algorithm. All images used were taken from those publicly available accompanying the Digital Image Processing Book by Gonzalez [3]. The empirical analysis will take each data set in-turn and present the thresholded output produced from each algorithm.

### B. Application of Thresholding to Portrait

The first data set considered is the famous portrait of Lena. The image is pictured in figure 3. This data set will give insight into how each of the thresholding algorithms perform on a typical portrait.

Each of the one dimensional histogram thresholding algorithms was applied to the Lena image and the results are depicted in table I. The captions are annotated with the thresholding values for each image. The algorithms of Lloyd-Max, Otsu and Pun all performed well on the portrait. Each was able to compute a threshold such that the shoulder, face and hat where classified as foreground. However, Pun's
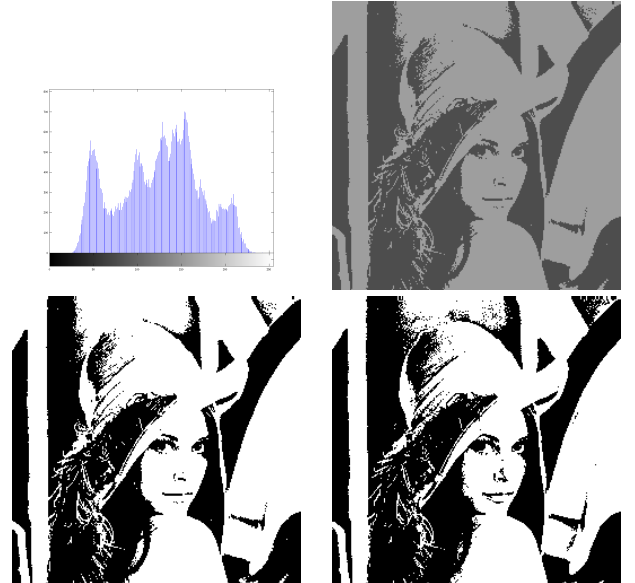


TABLE I: Top left: 1-D Histogram, Top right: Lloyd-Max (k=118), Bottom left: Otsu's (k=117), Bottom right: Pun's (k=125)

algorithm did slightly better at removing background pixels present directly above the hat.

Next, each of the two dimensional histogram thresholding algorithms was applied to the Lena image and the results are in table II. Upon examining the results, it is clear that the inclusion of spatial information in the thresholding process did not help in all cases. Abutaleb's method failed to identify any pixels as foreground. Pal's local entropy method produced results similar to the one dimensional histogram methods above. Pal's joint entropy method produced a threshold that is a little too high to discriminate some of the image features. Overall, Chang's method worked quite well in that it identified the hat, face and shoulder as foreground.

TABLE II: Top left: Abutaleb's (l,k=242,228), Top right: Chang's (k=170), Bottom left: Pal's Local Entropy (k=118), Bottom right: Pal's Joint Entropy (k=82)
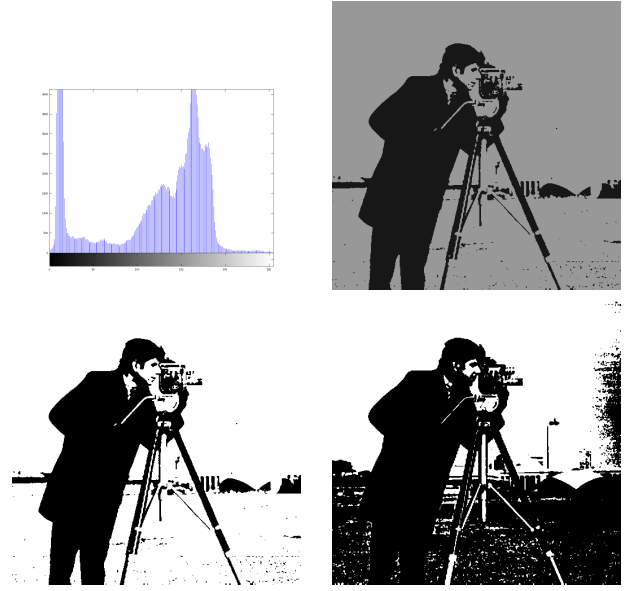


TABLE III: Top left: 1-D Histogram, Top right: Lloyd-Max (k=88), Bottom left: Otsu's (k=87), Bottom right: Pun's (k=148)

### C. Application of Thresholding to Outdoor Scene

The next data set considered is the famous image known as Cameraman. The image is pictured in figure 4. This data set will give insight into how each of the thresholding algorithms perform on an outdoor scene.



Fig. 4: Cameraman Original Image

Each of the one dimensional histogram thresholding algorithms was applied to the Cameraman image and the results are depicted in table III. The histogram is bi-modal with the scene background pixels consisting of the builds making up the histogram mode of the larger grayscale values. The algorithms of Lloyd-Max and Otsu performed equally well on the scene separating the background from the human figure and camera.

Pun's algorithm did not perform as well in that many of the pixels of the grassy areas were classified with the human figure.
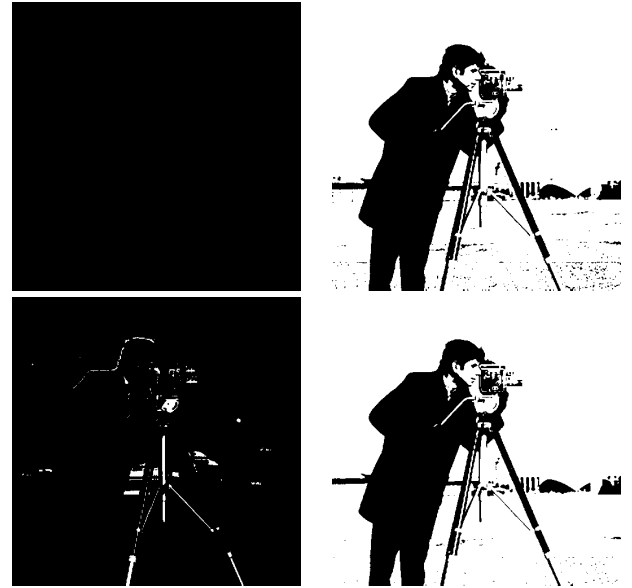


TABLE IV: Top left: Abutaleb's (l,k=254,252), Top right: Chang's (k=93), Bottom left: Pal's Local Entropy (k=193), Bottom right: Pal's Joint Entropy (k=80)

Next, each of the two dimensional histogram thresholding algorithms was applied to the Cameraman image and the results are in table IV. The results show that the inclusion of spatial information in the thresholding process did not help in all cases. Again, Abutaleb's method failed to identify any pixels as foreground. Pal's local entropy method produced results which highlighted some of the edge features of the

image. Pal's joint entropy method and Chang's method produced a threshold that is similar to Otsu's method giving good discrimination between the human figure with camera and the background buildings with grass.

### D. Application of Thresholding to Mammography Images

The next data set considered is the mammography containing a large mass. The image is pictured in figure 5. This analysis will help determine which thresholding algorithm can discriminate the mass from the background.
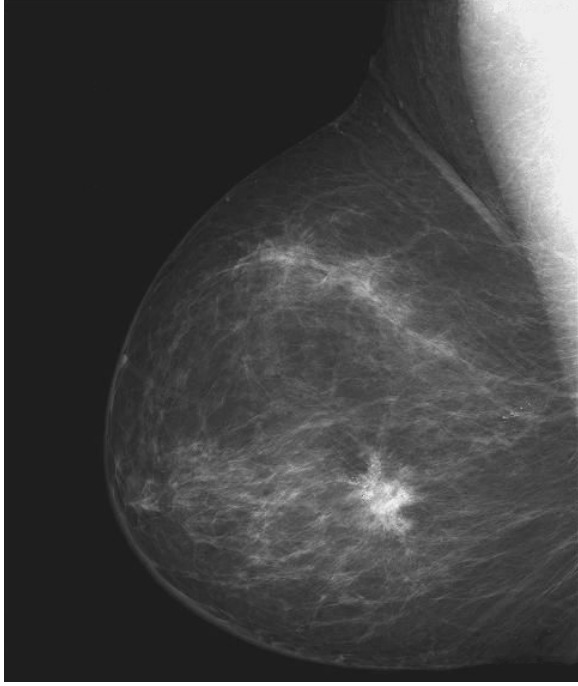


Fig. 5: Original Mammography Image

Each of the one dimensional histogram thresholding algorithms was applied to the mammography image and the results are depicted in table V. The histogram has a large number of pixels at the extremes of the dynamic range. These are due to imaging artifacts from the background and saturation in the top right of the original image. The objects of interest are contained in the skewed log-normal distribution of the histogram. Evaluating the one dimensional histogram algorithms we see that the Lloyd-Max and Otsu's method were able to classify the two masses as foreground. However, Pun's method chose a threshold which was too low resulting in low discrimination of the mass.

Next, each of the two dimensional histogram thresholding algorithms was applied to the mammography image and the results are in table VI. Abutaleb's, Chang's and Pal's local entropy method all performed well on this image. They computed a threshold slightly lower than the previous algorithms and therefore classified more of the pixels from the mass as foreground. Pal's joint entropy method performed slightly worse than all the others, but was still able to isolate the mass.
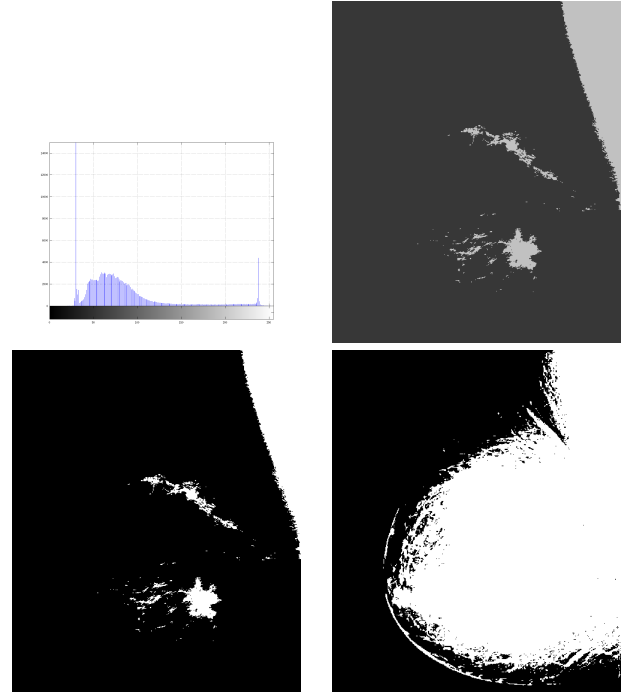


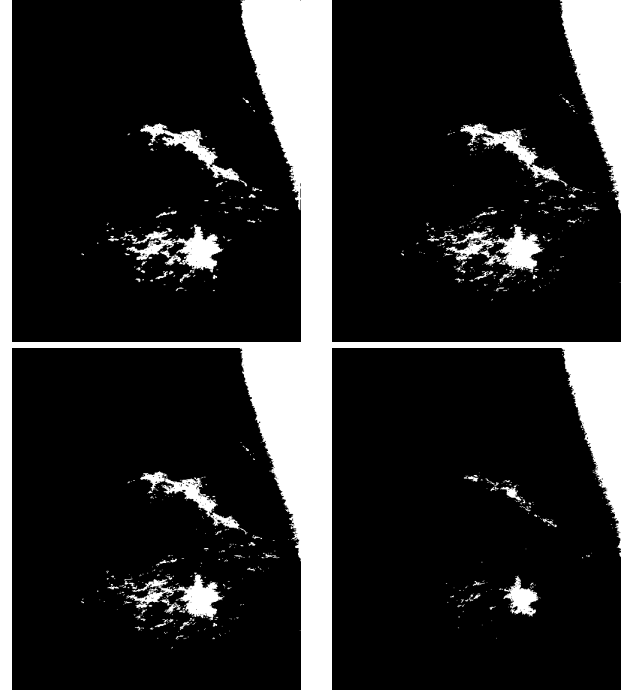TABLE V: Top left: 1-D Histogram, Top right: Lloyd-Max (k=124), Bottom left: Otsu's (k=124), Bottom right: Pun's (k=57)



TABLE VI: Top left: Abutaleb's (l,k=108,107), Top right: Chang's (k=110), Bottom left: Pal's Local Entropy (k=111), Bottom right: Pal's Joint Entropy (k=136)

### E. Application of Thresholding to Gamma Ray Imagery

The next data set considered is an image created via gamma ray detection from the Crab pulsar in space. The image is pictured in figure 6. This analysis will help determine which

thresholding algorithm can discriminate the large emitting objects from the rest of the pulsar in the noisy image.
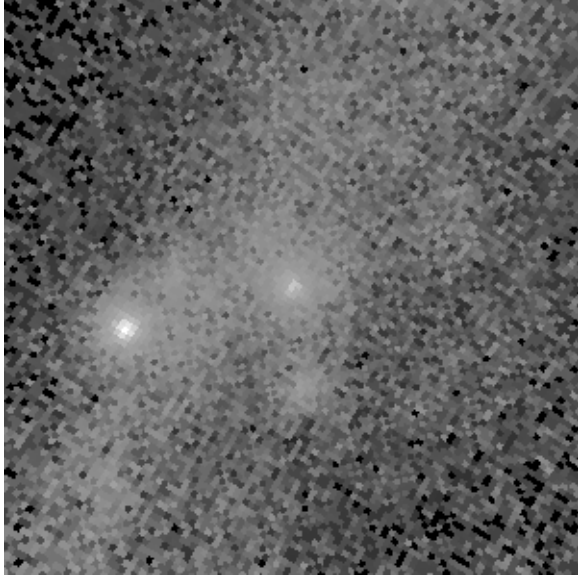


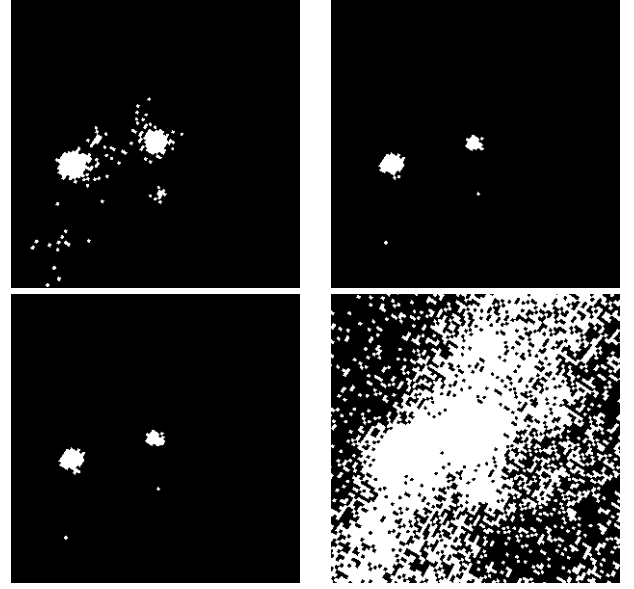Fig. 6: Original Crab Pulsar Gamma Image



TABLE VIII: Top left: Abutaleb's (l,k=148,95), Top right: Chang's (k=158), Bottom left: Pal's Local Entropy (k=156), Bottom right: Pal's Joint Entropy (k=99)

the results are in table VIII. Abutaleb's, Chang's and Pal's local entropy algorithms were all able to discriminate the bright objects in the scene. Abutaleb's algorithm did slightly better at classifying more pixels from the bright objects when compared to Change and Pal's local entropy method. However, Abutaleb's algorithm does have more false alarms when compared to Chang and Pal's local entropy method. Pal's joint entropy method performed similarly to the one dimensional algorithms. Clearly, when taking into account the spatial information, i.e. texture, it is possible to determine a good threshold which isolates the bright objects of the scene.

### F. Application of Thresholding to Xray Imagery

The last data set considered in this analysis is an image created via xray detection from the Crab pulsar in space. The image is pictured in figure 7. This image will give insight as to which thresholding algorithm can discriminate the large emitting objects from the rest of the pulsar.

Each of the one dimensional histogram thresholding algorithms was applied to the Crab pulsar xray image and the results are depicted in table IX. Again, the Lloyd-Max algorithm performed similarly to the Otsu's method. Also, Pun's method did better than Lloyd-Max and Otsu's method in discriminating the bright object at the center of the image. None of these method's did well in suppressing false alarms. Upon examining the histogram, it is not bi-modal and there does not appear to be an obvious separation of the pixels into clear object classes.

Next, each of the two dimensional histogram thresholding algorithms was applied to the Crab pulsar xray image and the results are in table X. Abutaleb's, and both of Pal's methods performed well on this image in that they were able to isolate the bright object from the rest of the image. Chang's algorithm did not perform well on this particular data set.
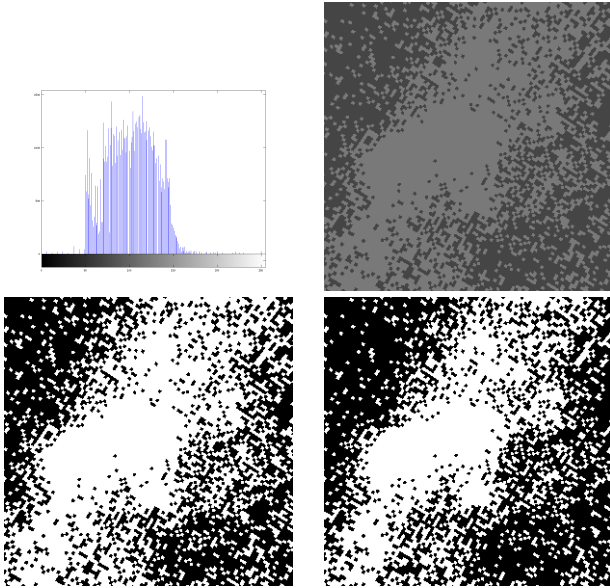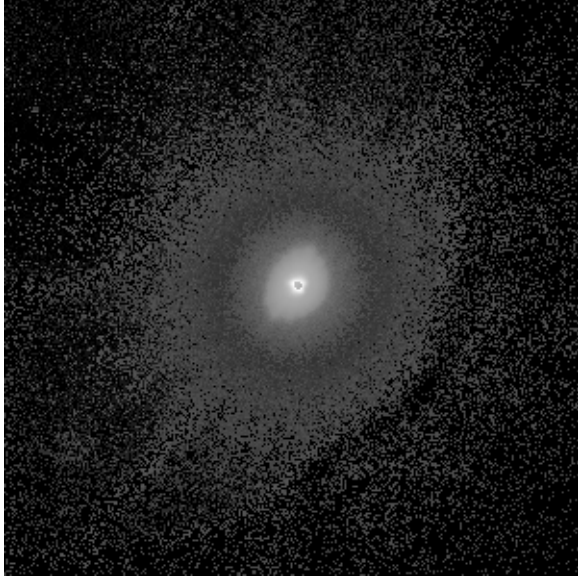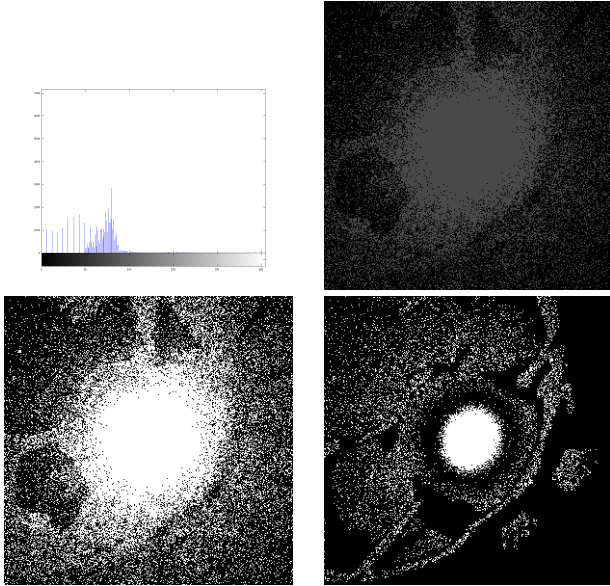


TABLE VII: Top left: 1-D Histogram, Top right: Lloyd-Max (k=95), Bottom left: Otsu's (k=94), Bottom right: Pun's (k=104)

Each of the one dimensional histogram thresholding algorithms was applied to the Crab pulsar gamma image and the results are depicted in table VII. The Lloyd-Max and Otsu's algorithm performed identically while Pun's algorithm did slightly better. However, none of the algorithms were able to isolate the bright objects in the scene. Examining the one dimensional histogram does not give any insight as to where a good threshold would exist.

Next, each of the two dimensional histogram thresholding algorithms was applied to the Crab pulsar gamma image and

Fig. 7: Original Crab Pulsar Xray Image



TABLE X: Top left: Abutaleb's (l,k=85,72), Top right: Chang's (k=2), Bottom left: Pal's Local Entropy (k=90), Bottom right: Pal's Joint Entropy (k=89)

The choice of a thresholding algorithm needs to be informed by the data on which it is operating. Inclusion of spatial information can degrade performance in cases of low texture, or can provide very powerful discrimination in cases of high texture.



TABLE IX: Top left: 1-D Histogram, Top right: Lloyd-Max (k=38), Bottom left: Otsu's (k=40), Bottom right: Pun's (k=80)

## III. Conclusion

This paper examined six different thresholding algorithms commonly used to threshold grayscale images. None of the algorithms performed well in all cases. Otsu's method and the Lloyd-Max algorithm performed almost identically on all images. In addition, using the co-occurrence matrix was not advantageous in all cases. In cases there were no dominant textures, leveraging the co-occurrence matrix degraded performance as seen in the Lena and Cameraman images. However, in cases were spatial interactions between pixels present high levels of texture, the co-occurrence matrix provided necessary information to the thresholding algorithm which allowed for better discrimination of objects in the scene. This was exemplified in the mammography image and space-borne images.

## References

[1] Ahmed S. Abutaleb. Automatic thresholding of gray-level pictures using two-dimensional entropy. *IEEE Transactions on Computer Vision, Graphics, and Image Processing*, 47(1):22–32, 1989.

[2] C.-I Chang, K. Chen, J. Wang, and M. Althouse. A relative entropy-based approach to image thresholding. *IEEE Transactions on Pattern Recognition*, 27(9):1275–1289, 1994.

[3] Rafael C. Gonzalez. Standard image processing test images. http://web.eecs.utk.edu/~gonzalez/ipweb2e/downloads/standard_test_images/, 2012. [Online; accessed 12-October-2012].

[4] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.

[5] Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

[6] Joel Max. Quantizing for minimum distortion. *IEEE Transactions on Information Theory*, 6:7–12, 1960.

[7] N. Otsu. A threshold selection method from gray level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9:62–66, 1979.

[8] N.R. Pal and S.K. Pal. Entropic thresholding. *IEEE Transactions on Signal Processing*, 16:97–108, 1989.

[9] T. Pun. Entropic thresholding: A new approach. *Computer Graphics and Image Processing*, 16(3):210–239, 1981.

**Bernard Lampe** (M'09) became an IEEE Member (M) in 2009 and received his bachelors of science degree from The University of Michigan in Ann Arbor, Michigan, USA in 2009.

Mr. Lampe is also a member of the American Society for Computing Machines (ACM) since 2009.