TELKOMNIKA, Vol.15, No.3, September 2017, pp. 1317~1327
ISSN: 1693-6930, accredited **A** by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v15i3.5105

Binarization of Ancient Document Images based on Multipeak Histogram Assumption

Fitri Arnia*, Khairul Munadi
Department of Electrical and Computer Engineering, Syiah Kuala University,
Jl. Tgk. Syech Abdurrauf No. 7, Banda Aceh, Indonesia
*Corresponding author, e-mail: f.arnia@unsyiah.ac.id

Abstract

In document binarization, text is segmented from the background. This is an important step, since the binarization outcome determines the success rate of the optical character recognition (OCR). In ancient documents, that are commonly noisy, binarization becomes more difficult. The noise can reduce binarization performance, and thus the OCR rate. This paper proposes a new binarization approach based on an assumption that the histograms of noisy documents consist of multipeaks. The proposed method comprises three steps: histogram calculation, histogram smoothing, and the use of the histogram to track the first valley and determine the binarization threshold. In our simulations we used a set of Jawi ancient document images with natural noises. This set is composed of 24 document tiles containing two noise types: show-through and uneven background. To measure performance, we designed and implemented a point compilation scheme. On average, the proposed method performed better than the Otsu method, with the total point score obtained by the former being 7.5 and that of the latter 4.5. Our results show that as long as the histogram fulfills the multipeak assumption, the proposed method can perform satisfactorily.

Keywords*: Multipeak histogram, image binarization, global thresholding, OCR, noisy document*

1. Introduction

Ancient documents contain essential information that reveals mankind's past activities, letters, decisions, and habits to help us better understand our history. To make them more accessible, the content of historical documents must be disseminated and the most reliable approach is to distribute them using Internet technology. To do so, the contents of the documents or the documents themselves must be converted to digital versions that are searchable. Searchable digital documents can be obtained in three steps. First, digitalization can be accomplished either by scanning or photographing an original document. Once there is a digital version of the document, the second step is called binarization, which segments the text from its background. The outcome of binarization determines the success of the third step-optical character recognition (OCR).

Binarization of an ancient document is made more difficult due to the presence of noise. In noisy documents, many parts of the text are segmented as background and conversely [1]. The noise appears as various background illuminations and stains, which can reduce the performance of the binarization algorithm, and thus the OCR rate. Binarization can be accomplished by applying a threshold value either globally [2-4] or locally [5-11], or a combination of both [12-13]. In global thresholding, one threshold value is applied over all document images, and it is assumed that the images are bimodal, i.e., they consist of the object (text) and the background. This approach is an established concept that is the simplest yet useful approach to segmenting images [14-15]. Local thresholding applies a threshold value over a limited location in an image, and the value can be adaptively changed according to local illumination variations in the document [1, 7]. Combined methods take information from the entire image as well as neighborhood pixels to determine a threshold value [12-13]. There are a number of problems in current binarization techniques: (1) the global approach with its bimodal assumption is not suitable for documents with noise [10]; (2) the conventional local approach, introduces additional noise into binarized documents, especially in non-text regions [5], and fails to binarize faint characters [6]; (3) modified local [8-9] and combined approaches, intended to overcome limitations in the global and local methods, unfortunately require many procedures,

Received April 9, 2017; Revised June 22, 2017; Accepted July 9, 2017

most of which have been tested on a benchmark database with artificial noise or limited noise variations.

In this paper, we propose a new document image binarization technique based on an image histogram. This technique applies a global threshold, but it is based on the observation that histograms of noisy documents contain several peaks (i.e., multipeaks), of which the first relates to the text, while the others correspond to background areas of different illuminations/types of noise. Our method is as simpler as the method based on the global approach, yet it can detect the noises better due to there is no bimodal assumption. Rather than testing the proposed method with a standard database, in our simulations we used a set of Jawi ancient document images with natural noises, that are show-through noise and uneven background. The simulation results showed that the proposed method performed better than Otsu method [2] in binarizing documents with show-through noise, and can binarize documents containing uneven background moderately well.

The remainder of this paper is organized as follows: In section 2, we present our proposed binarization method. In section 3 and 4, we present the research method and analysis of the performance of the proposed method respectively. Finally, in section 5 we make our concluding remarks.

2. Proposed Method

The propose binarization approach is based on the assumption that the histogram of a document image consists of multipeaks, in which the first peak corresponds to the text, the last peak corresponds to the background and each peak in between corresponds to noises with various gray tones. Based on this assumption, we have a new hypothesis that the deepest valley between the first and the second peak is the threshold location that separates the text from its background and noise. The proposed method has three main steps: (1) histogram calculation, (2) histogram smoothing, and (3) using the histogram to track the first valley and determine the binarization threshold. The threshold value is the location of the first valley on the histogram. For a document with a high illumination variation (or noise characters), the image can be tiled to obtain a sufficient locality an area with a stable illumination before the threshold is determined. A smoothing procedure is carried out to minimize the number of local minima in the histogram.

2.1. Algorithm

The proposed binarization algorithm is as follows:

1. Calculating the histogram.

Let us consider a gray-tone image I(r) that is represented by an 8-bit integer, giving a possible intensity level L = 256 or 0 < r < 255. Here, r = 0 is represented as black, r = 255 is represented as white, and values between 0 and 255 are represented as a gradation of gray tones. Let z_k be the number of pixels associated with the kth intensity level in histogram h(k).

We then calculate histogram h (k) by Equation (1):

$$h(k) = z_k, \quad 0 < k < 255 \tag{1}$$

and calculate the normalized histogram by Equation (2):

$$h_n(k) = \frac{h(k)}{n} = \frac{z_k}{n} \tag{2}$$

where n represents the total number of pixels in image I(r).

2. Smoothing the histogram.

A histogram typically consists of many local minima. Histogram smoothing reduces the number of local minima. Here, we use the smoothing-by-averaging technique, as proposed in [14]. Averaging for a window size 5 is given by Equation (3) below:

$$h_{ns}(k) = \frac{\left(h(k-2)+h(k-1)+h(k)+h(k+1)+h(k+2)\right)}{5} \tag{3}$$

where $h_{ns}(k)$ represents the normalized and smoothed histogram.

3. Finding the threshold value.

On the smoothed histogram, we then identify the deepest valley between the first and the second peak, and determine the value *k* as the threshold value.

a. Finding the remaining local valleys v_l in the histogram is achieved by the following:

for k = *lo* to *hi*
if $h_{ns}(k) < h_{ns}(k-1)\ and\ h_{ns}(k) < h_{ns}(k+1)$
then $v_l = h_{ns}(k)$
end if
end for

where "*lo*" and "*hi*" are the lower and upper limits of the local valley searching area on the histogram.

b. Then,

for all v_l,
find $v_{ns}(k)$ with the smallest $h_{ns}(k)$; the position of the deepest valley between lo and hi;
threshold = $v_{ns}(k)$
end for

c. If the $v_{ns}(k)$ is not the single threshold value, repeat step (2) ad (3).

Note, Several cycles of the smoothing operation, as described in step 2, may be necessary to determine the single threshold $v_{ns}(k)$.

2.2. Illustration of Implementation Procedure

Figure 1 illustrates the algorithm implementation, where Figure 1(a) shows an example of a noisy ancient document, 1(b) is the normalized histogram of the image in 1(a), and 1(c) is the smoothed histogram produced by the smoothing operation given in Equation (3). Finally, Figure 1(d) is the threshold value calculated by the proposed algorithm. The smoothing process depends on the condition of the image noise types or digital resolution and may be applied for more than one cycle to obtain the necessary smoothness. Figure 2 shows the effect of the smoothing procedure on the histogram, in which the green line is the original histogram, which has many local minima and maxima. The first smoothed histogram is plotted in blue, and several fluctuations remain, producing local minima and maxima. The second smoothed histogram is plotted in red, and is sufficiently smoothed for the subsequent procedure.

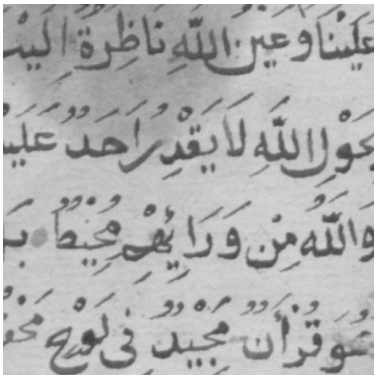
3. Research Method

The performance of the proposed method was evaluated using a set of Jawi ancient document images with various natural noises. We used the Otsu method for comparison, and assumed the comparison to be reasonable because (1) both methods apply the global thresholding approach, and (2) both methods were evaluated without additional pre-processing such as background estimation or contrast compensation.

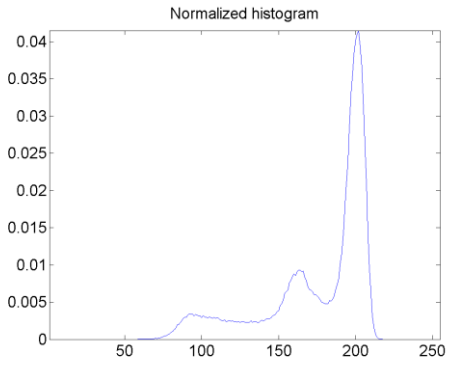
For each binary image, we measured the performance by roughly counting the number of broken and missed characters in the binarized documents. The method that produced binary documents with fewer broken or missed characters is rated as having higher performance, and vice versa. The term broken, with respect to a character, refers to one of two conditions; (1) the character stroke in the binary image has become too thin, and thus has broken, or (2) the character strokes in the binary image are too thick, and are thus combined.

To test the proposed method, in the simulations we used 24 segments of scanned document images obtained from the Aceh Museum [16]. These documents, written with Arabic letters that are pronounced either in Arabic or Jawi, contain additional black background and several attributes in addition to the document pages themselves, and also vary in size. Thus, cropping is necessary to obtain only the document pages. From the scanned documents, we localized two noise types-show-through effect and uneven background-then tiled the images into 512 × 512 or 256 × 256 pixels, and binarized them using the proposed and Otsu methods.

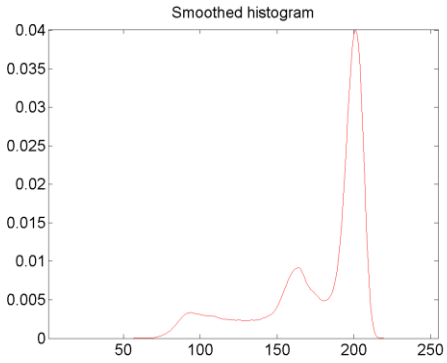
Figure 3 shows some examples of the noisy document tiles. Show-through is a text shadow that comes through from the other side of the paper. Uneven background refers to a condition in which a document has more than one type of noise, or has noises that cannot be easily classified.



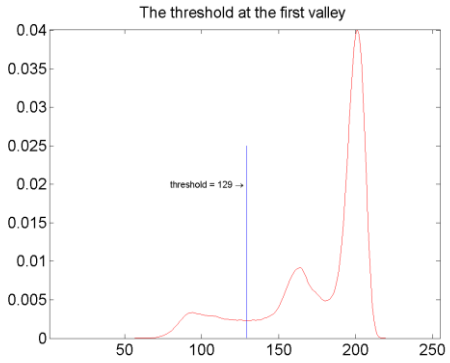
(a) Noisy ancient document



(b) Normalized histogram of image in (a)



(c) Smoothed version of histogram in (b)



(d) Threshold value determined by the algorithm

Figure 1. Illustration of algorithm implementation results

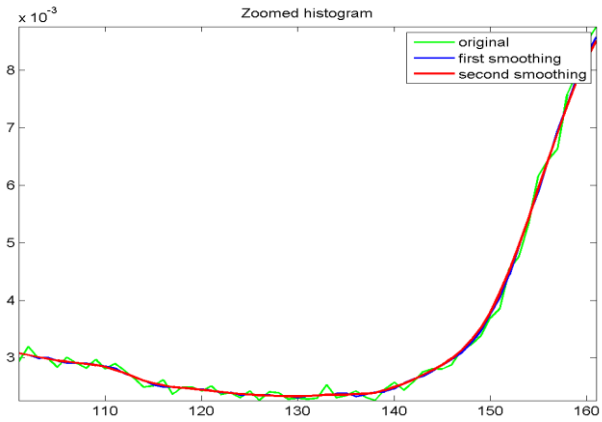


Figure 2. Original histogram and its smoothed versions

Figure 3. Samples of document tiles used in the simulations, consisting of two noise types. Top row: show through effects, bottom row: uneven background.

4. Results and Analysis

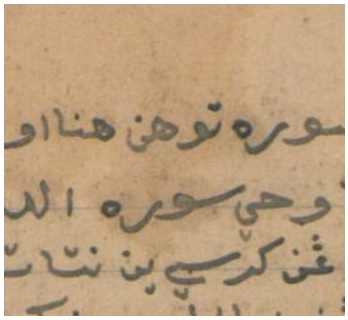
4.1. Smoothing Cycles

Before determining the threshold value on the histogram, it must be sufficiently smoothed. We found that in some cases, more than one smoothing cycles is required. In the simulations, for most of the document tiles the threshold could be determined after one smoothing cycle, but approximately 3% required two or three smoothing cycles. Figure 4 shows two examples of thresholds determined after applying different numbers of smoothing cycles. Figure 4(a) and 4(b) show a noisy document its corresponding histogram, respectively, in which the threshold was able to be determined after two smoothing cycle. Figures. 4(c) and 4(d) show another noisy document and its corresponding histogram, respectively, for which three smoothing cycles were needed to determine the threshold. We found that the number of required cycles is not dependent on the noise type.

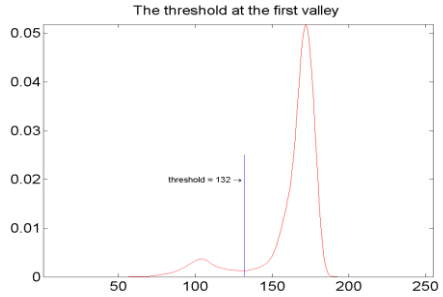
4.2. Performance of the Proposed Method

Here, we present and discuss the performance of the proposed method with respect to the type of noise: show-through and uneven background. In Table 1 and 2, when the binary image of the proposed/Otsu method had fewer broken characters, it was given a point, denoted by the sign '√', and the sign '-' denotes the opposite case. The sign '=' indicates that the same number of broken characters was produced by both methods, which is given 0.5 points.

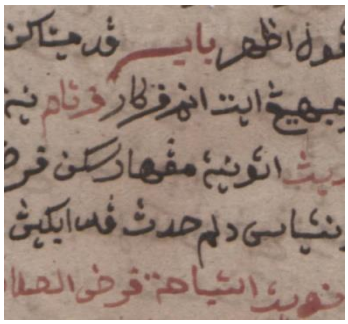
Table 1 shows the threshold value and performance of the proposed method for a document with the show-through effect. All the thresholds determined by the proposed method were lower than those determined by Otsu. In most cases, there were fewer broken characters in documents prepared by the proposed method than those prepared by the Otsu method. However, in three documents show-through-6, show-through-9, and show-through-10 there were more broken characters in those prepared by the proposed method than those prepared by the Otsu. Thus, for the show-through effect, the total point value of the proposed method was 9 and that of Otsu was one third of that total.



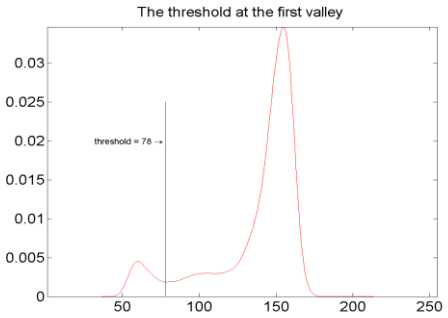
(a) noisy document 1



(b) histogram of (a)



(c) noisy document 2



(d) histogram of (c)

Figure 4. Samples of document tiles with noises and their histograms, (a) original noisy document 1, for which two smoothing cycles were applied to its histogram in (b); (c) original noisy document 2, for which its histogram in (d) was smoothed by three smoothing cycles

Table 1. Threshold values and number of point in binarization results of the proposed and Otsu methods in show-through documents

No	Document Tiles	Threshold Value		Number of ticks	
		Proposed	Otsu	Proposed	Otsu
1	Show-through-1	126	158	√	-
2	Show-through-2	135	153	√	-
3	Show-through-3	124	154	√	-
4	Show-through-4	131	151	√	-
5	Show-through-5	132	149	√	-
6	Show-through-6	126	153	-	√
7	Show-through-7	140	151	√	-
8	Show-through-8	134	154	√	-
9	Show-through-9	97	123	-	√
10	Show-through-10	103	132	-	√
11	Show-through-11	127	152	√	-
12	Show-through-12	133	150	√	-
			Total point	9	3

Figures 5(a) and 5(b) show the original show-through-1 and its gray-scale histogram, respectively. The show-through effect in the document produced three shades in the gray-scale; the darkest being the text, the next darkest shade being the show-through component, and the lighter shade being the background. This result corresponds to its histogram, where the first peak corresponds to the text, the last peak corresponds to the background, and the smaller peaks in between correspond to the show-through noise. The threshold of the proposed method was 126, and the three shades mentioned above can be separated nicely, resulting in the binary document shown in Figure 5(c). The Otsu threshold was 158, which was too high, and produced a binary document with roughly 50 broken characters. In this case, the show-through noise had been segmented as text, causing some characters to be combined (see Figure 5(d)).

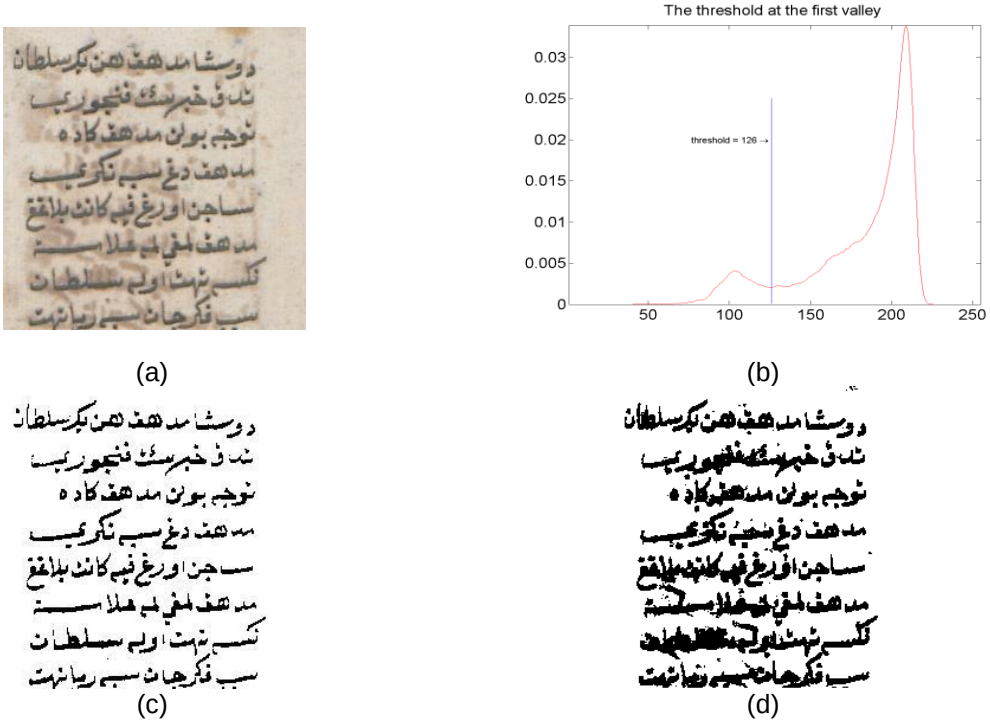


Figure 5. Top row: (a) show-through-1, (b) gray-scale histogram of (a) indicating the threshold value; bottom row: binarized documents of the (c) proposed method, and (d) Otsu method

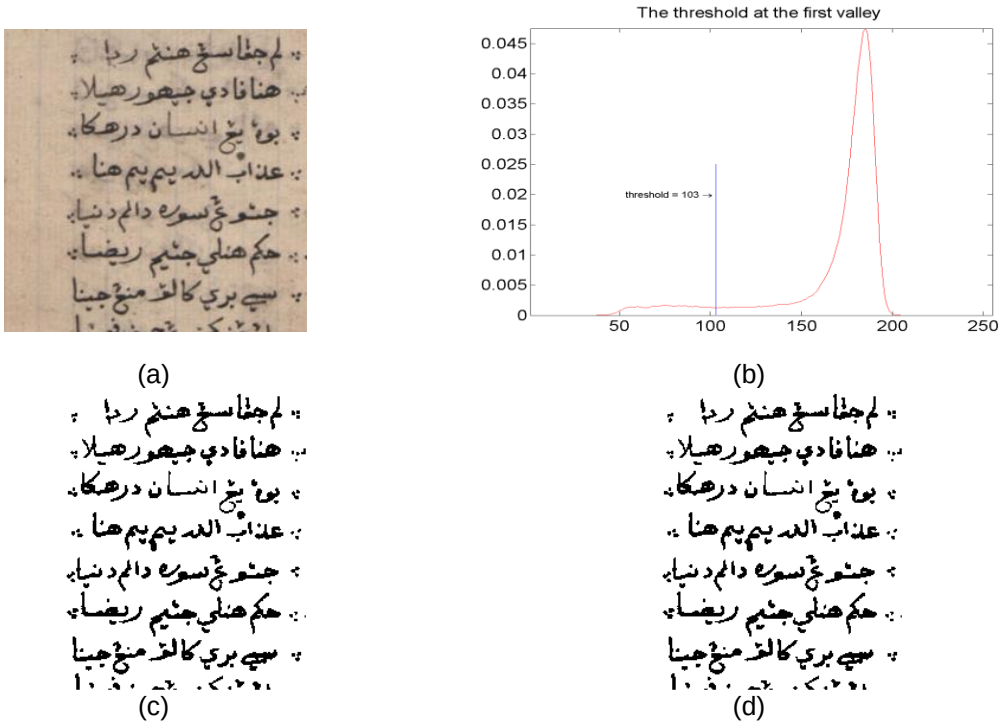


Figure 6. Top row: (a) show-through-10, (b) gray-scale histogram of (a) indicating the threshold value; bottom row: binarized documents of the (c) proposed method, and (d) Otsu method

Figure 6 shows the show-through-10 document, in which the Otsu method produced fewer broken characters compared to the proposed method. The original document in Figure

6(a) has a slight show-through effect; that is, there was no significant noise. Its grayscale histogram in Figure 6(b) does not have any clear valley, and there is a long flat region in the histogram. In this case, the multipeak assumption cannot be efficiently and correctly implemented to determine the threshold value.

Table 2 lists the threshold values and points obtained by both methods for documents with uneven background. For 50% of the documents, namely uneven background-2, -5, -7, -9, -10, and uneven background-12, the proposed method performed as well as Otsu. The proposed method performed better in 25% of the documents uneven background-1, -4, and -6. The Otsu method performed better in the remaining 25% uneven background-3, -8, and -11. The end result was that each method scored a total of six points. All the thresholds of the proposed method documents were lower than those of the Otsu. Next, we present and discuss two contrary results. The first is uneven background-1, in which the proposed method resulted in fewer broken characters, and the second is uneven background-11, in which the proposed method produced numerous broken characters.

Table 2. Threshold values and number of point in binarization results of the proposed and Otsu methods of documents with uneven background

No	Document Tiles	Threshold Value		Number of Ticks	
		Proposed	Otsu	Proposed	Otsu
1	Uneven background-1	114	135	√	=
2	Uneven background-2	124	137	=	=
3	Uneven background-3	130	141	-	√
4	Uneven background-4	122	136	√	-
5	Uneven background-5	93	116	=	=
6	Uneven background-6	94	111	√	-
7	Uneven background-7	102	114	=	=
8	Uneven background-8	140	154	-	√
9	Uneven background-9	127	136	=	=
10	Uneven background-10	133	136	=	=
11	Uneven background-11	142	165	-	√
12	Uneven background-12	155	163	=	=
			Total point	6	6

Figure 7(a) shows the original image of uneven background-1, and Figure 7(b) shows its histogram. The binary images of the proposed and Otsu methods are shown in Figures 7(c) and 7(d). The noise in the document background includes spots and show-throughs. Its histogram follows the assumption of the proposed method, i.e., it has a deep first valley. The threshold of the proposed method was 114, and that of Otsu was 135. If we compare the binary documents of the two methods, we can see that the spots in the binary image of the first are thinner than those in the latter. In this case, the spots of the show-through effect in the original document were more significant in the Otsu binary image. Furthermore, some characters in the Otsu image were combined, as denoted by the red circles.

Figure 8(a) shows the original document tile of uneven background-11, 8(b) is its gray-scale histogram, and 8(c) and 8(d) are the binary images produced by the proposed and Otsu methods, respectively. The histogram of document image was in accordance with the assumption of the proposed method. However, the detected threshold of 142 was located at a local minimum, the characters in the binary image were too thin, and many characters were broken. On the other hand, the Otsu threshold was 165, and the characters in the binary image based on the threshold were sufficiently thick, as we can see in Figure 8(d). Unfortunately, some stain due to water spilling remained.

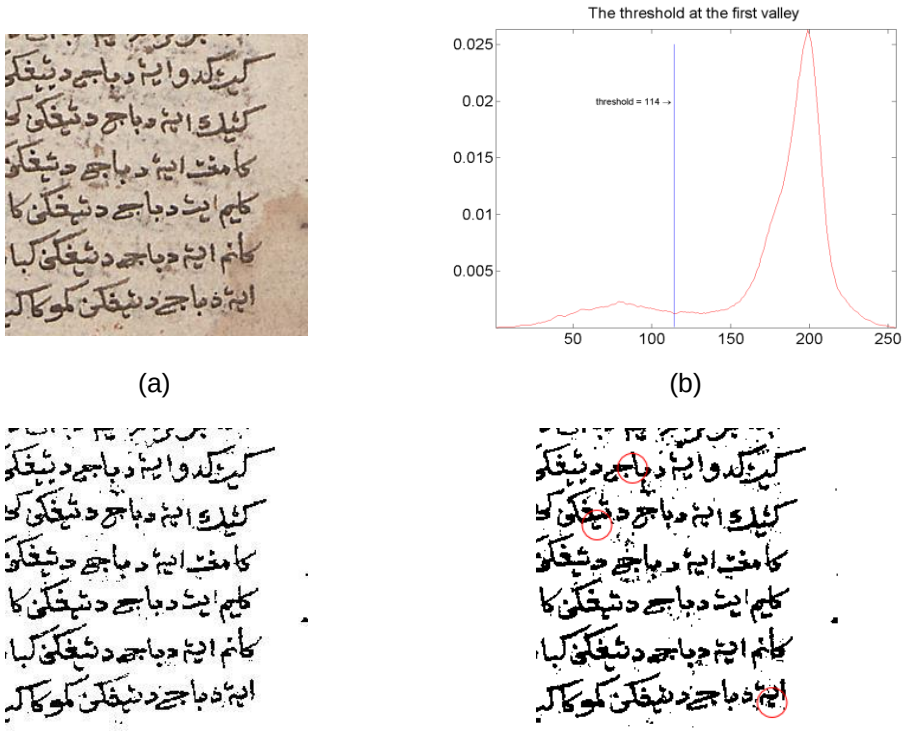


Figure 7. Top row: (a) uneven background-1 and (b) its gray-scale histogram indicating the threshold value; bottom row: binarized documents of the (c) proposed and (d) Otsu methods

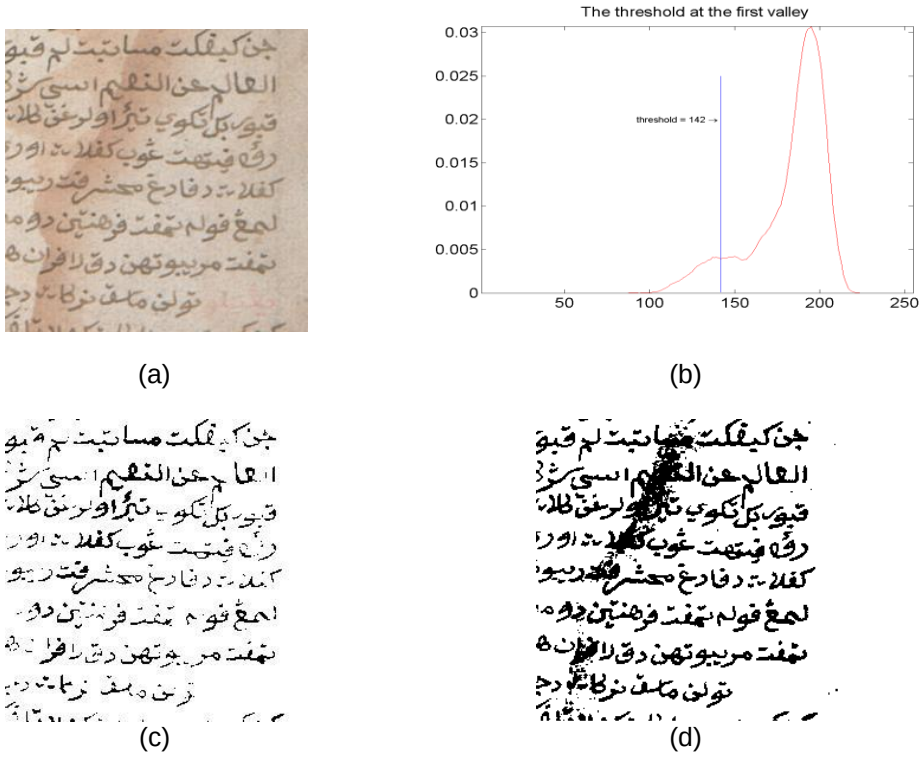


Figure 8. Top row: (a) uneven bacground-11, (b) gray-scale histogram of (a) indicating the threshold value; bottom row: binarized document of the (c) proposed method, and (d) Otsu method

4.3. Discussion and Future Work

The average total point of the proposed method across the two noise types was 7.5, which was higher than that of Otsu, which was 4.5. These results show that the proposed method performed best in documents with show-through effects. The data also showed that the proposed method may perform well in documents with uneven background. Overall, we suggest that the proposed method can be appropriately used to binarize noisy documents containing various noise types, as long as their histograms are in accordance with the multipeak assumption.

The proposed method performed poorer than Otsu for a certain condition, specifically when the shape of the histogram does not follow the mulitpeak assumption. In this case, the histogram has no first peak, and the first valley on the histogram is long and flat. The proposed algorithm can fail to or falsely determine the threshold location, as shown in Figure 6. However, in many cases, this histogram sequence-no-peak, followed by the long flat valley, and ending with a peak-indicates that the original document does not have significant noise. Thus, the Otsu method is more suitable for binarizing documents with this histogram type.

Another problem occurs when the detected threshold value is a local threshold, as discussed regarding Figure 8. The binary image of the proposed method in Figure 8 was determined after applying one smoothing cycle to the histogram. Unfortunately, the first valley was a local threshold position. In this case, the histogram must be smoothed more than once to obtain the necessary smoothness and correctly locate the threshold. However, the proposed algorithm is not yet so equipped.

In the future, the proposed method will be enhanced by (1) developing an algorithm to detect the condition of the valley, whereby if the valley is flat and sufficiently long, it is highly probable that the Otsu method will perform better, and (2) developing an algorithm for optimizing the number of smoothing cycles needed to produce binary images with as few broken characters as possible.

5. Conclusion

In this paper, we proposed a novel binarization method for ancient noisy document images, based on the assumption that histograms of the document images consist of multipeaks. The proposed method comprises three steps: (1) histogram calculation, (2) histogram smoothing, and (3) using the histogram for first valley tracking and threshold determination. In the simulations, we used a set of Jawi ancient document images comprising 24 document tiles with natural noises. We considered two noise types: show-through and uneven background. The simulation results suggest that as long as the histogram fulfills the multipeak assumption, the proposed method performs satisfactorily. Based on the tested noise types, the results also show that the proposed method performed best when binarizing documents with show-through noise, and can binarize documents containing uneven background moderately well. On average, the proposed method performed better than the Otsu method, with a total obtained point value of 7.5 compared to 4.5, respectively. In future work, we plan to enhance the performance of the proposed method by (1) developing an algorithm to detect the condition of the valley, and (2) developing an algorithm for optimizing the number of smoothing cycles required to produce binary images with as few broken characters as possible.

References

[1] B Gatos, I Pratikakis, SJ Perantonis. Adaptive Degraded Document Image Binarization. *Pattern Recognition*. 2006; 39(3): 317-327.
[2] N Otsu. A Threshold Selection Method from Gray-level Histograms. *Automatica*. 1975; 20(1): 62-66.
[3] E Kavallieratou, H Antonopoulou. Cleaning and Enhancing Historical Document Images. *Advanced Concepts for Intelligent Vision Systems of the series Lecture Notes in Computer Science*. 2005; 3708: 681-688.
[4] Anny Yuniarti, Anindhita Sigit Nugroho, Bilqis Amaliah, Agus Zainal Arifin. Classification and Numbering of Dental Radiographs for an Automated Human Identification System. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10(1): 137-146.
[5] W Niblack. An Introduction to Digital Image Processing. UK: Prentice Hall. 1986: 114-115.
[6] J Sauvola, M Pietikainen. Adaptive Document Image Binarization. *Pattern Recognition*. 2000; 33(2): 225-236.

[7] K Khurshid, I Siddiqi, C Faure, N Vincent. *Comparison of Niblack Inspired Binarization Methods for Ancient Documents*. Proceedings of SPIE 7247, Document Recognition and Retrieval XVI. 2009: 7247.
[8] S Lu, B Su, CL Tan. Document Image Binarization Using Background Estimation and Stroke Edges. *International Journal on Document Analysis and Recognition (IJDAR)*. 2010; 13(4): 303-314.
[9] B Su, S Lu, CL Tan. *Binarization of Historical Document Images Using the Local Maximum and Minimum*. International Workshop on Document Analysis Systems (DAS). Boston, MA, USA. 2010: 159-166.
[10] LTK Van, G Lee. Stroke Width-Based Contrast Feature for Document Image Binarization. *Journal of Information Processing Systems*. 2014; 10(1): 55-68.
[11] Jie Zhang, Chengjun Xie, Liangtu Song, Rui Li, Hongbo Chen. Robust Image Segmentation Using LBP Embedded Region Merging. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2017; 14(1): 368-377.
[12] K Ntirogiannis, B Gatos, I Pratikakis. A Combined Approach for the Binarization of Handwritten Document Images. *Pattern recognition letters.* 2014*;* 35: 3-15.
[13] S Saha, S Basu, M Nasipuri. *Binarization of Document Images Using Hierarchical Histogram Equalization Technique with Linearly Merged Membership Function*. Proceedings of the International Conference on Information Systems Design and Intelligent Applications. Visakhapatnam, India. 2012: 639-647.
[14] KS Tan, NAM Isa. Color Image Segmentation Using Histogram Thresholding–Fuzzy C-means Hybrid Approach. *Pattern Recognition.* 2011; 44(1): 1-15.
[15] R Hedjam, HZ Nafchi, M Kalacska, M Cheriet. Influence of Color-to-Gray Conversion on the Performance of Document Image Binarization: Toward a Novel Optimization Problem. *IEEE Transactions on Image Processing*. 2015; 24(11): 3637-3651.
[16] Manuscript of Museum Aceh. Available: http://www.islamic-manuscripts.net.