Robust Binarization of Stereo and Monocular Document Images Using Percentile Filter

Muhammad Zeshan Afzal[1], Martin Krämer[1], Syed Saqib Bukhari[1]
Mohammad Reza Yousefi[1], Faisal Shafait[2], Thomas M. Breuel[1]
[1]Technical University of Kaiserslautern, Germany.
{*afzal, kraemer, bukhari, yousefi, tmb*}*@iupr.com*
[2]The University of Western Australia, Australia.
faisal.shafait@uwa.edu.au

***Abstract*—Camera captured documents can be a difficult case for standard binarization algorithms. These algorithms are specifically tailored to the requirements of scanned documents which in general have uniform illumination and high resolution with negligible geometric artifacts. Contrary to this, camera captured images generally are low resolution, contain non-uniform illumination and also posses geometric artifacts. The most important artifact is the defocused or blurred text which is the result of the limited depth of field of the general purpose hand-held capturing devices. These artifacts could be reduced with controlled capture with a single camera but it is inevitable for the case of stereo document images even with the orthoparallel camera setup.**

Existing methods for binarization require tuning for the parameters separately both for the left and the right images of a stereo pair. In this paper, an approach for binarization based on the local adaptive background estimation using percentile filter has been presented. The presented approach works reasonably well under the same set of parameters for both left and right images. It also shows competitive results for monocular images in comparison with standard binarization methods.

I. Introduction

The extensive use of portable cameras for capturing documents is driving the current research in the area. It is due to the inexpensiveness and ease of use of such devices. Although camera captured documents offer many advantages, they also have inherited problems because of capturing procedure and the devices themselves, which are used for capturing. The document image processing pipeline both for the monocular and stereo images [1], [2], [3] in most of the cases starts with the binarization of the document images in order to extract bi-level features for further processing.

Off-the-shelf passive sensing devices, e.g. customer grade hand-held cameras, can only focus objects which are at a certain distance from the camera. This is known as depth of field. The objects or the parts of objects which are nearer or farther from that end up being not properly focused in the captured image. Although it is possible to have a setup with large enough depth of field, it is not possible with customer consumer grade cameras, because it requires the knowledge about the actual scene properties, e.g. distance to the objects, and the camera properties, e.g. lens aperture etc. Another way to tackle this problem is to correct depth of field after acquiring the images, but it would require multiple captures depending upon the scene.

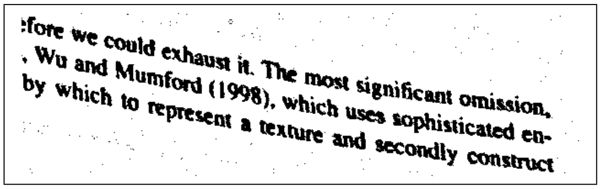
We show the effect of binarization for the stereo pairs using local adaptive binarization approach Savoula as it has been reported to perform relatively better than other local approaches [4].

We consider a stereo image pair where the effect of blurring in the left image is major. In Figure 1 the left column, i.e. Figures (1a, 1c, 1e and 1g) correspond to the left image of the stereo pair. In Figure 1 the right column, i.e. Figures (1b, 1d, 1f and 1h) correspond to the right image of the stereo pair. The left image is blurred and the right image is correctly focused. The binarization for both of the images has been carried out using Savoula with a window size of 30 as it has been reported by Bukhari et al. [5] as suitable window size. The values of k are varied for both of the images. As we lower the value of k, which is depicted in Figure 1a and Figure 1b, the results of binarization become noisy. It contains salt and pepper noise. On the other hand increasing the value of k acts differently on left and right image. While the binarization of the right image remains reasonable under the changing the value of k, the left image, which is blurred, produces degraded binarization results and the foreground is vanishes with the increasing value of k.

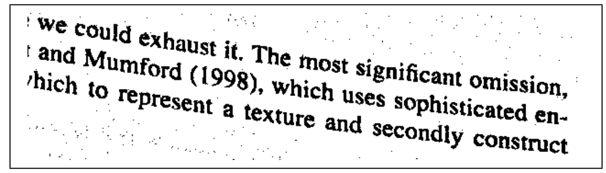
This paper proposes an approach based on background estimation using percentile filters which performs reasonable binarization for both left and right images of stereo pair under the same set of parameters. The rest of this paper is organized as follows: the next section describes the related work for the approaches for binarization, the percentile filter and the stereo approaches document image processing which uses binarization. Section 3 describes the percentile filter and the proposed binarization algorithm. Section 4 shows the experimental results with quantitative evaluation and the paper ends with the conclusion presented in Section 5.

II. Related Work

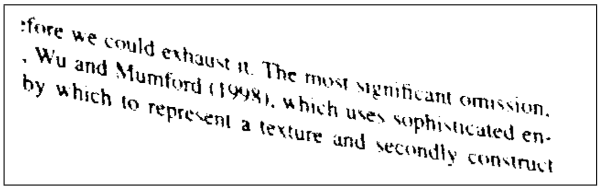
The diversity of document images have been driving the research in document image processing in various directions. The researchers are trying to come up with generalized methods to be able to process a wide variety of documents. The binarization methods have also been proposed keeping in mind certain types of document images. The binarization of documents is aimed at either color [6], [7], [8], [9] or gray [10], [4], [11], [12], [5], [13] level images, which leads to the different methods of binarization. In general, when the image is being thresholded it could be done by determining



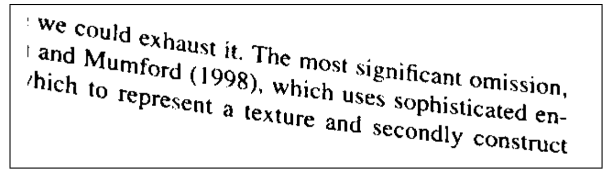
(a) w=30 , k=0.005



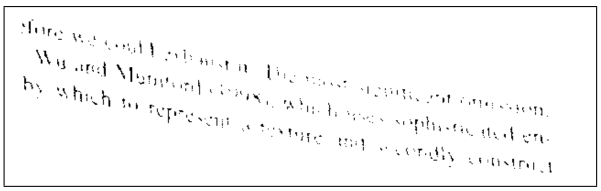
(b) w=30 , k=0.005



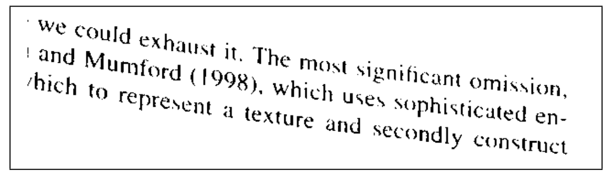
(c) w=30 , k=0.2



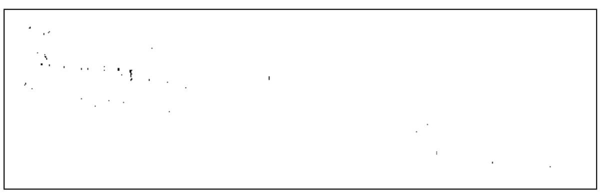
(d) w=30 , k=0.2



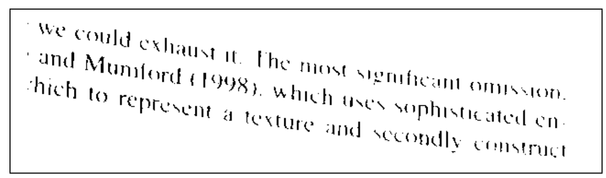
(e) w=30 , k=0.35



(f) w=30 , k=0.35



(g) w=30 , k=0.5



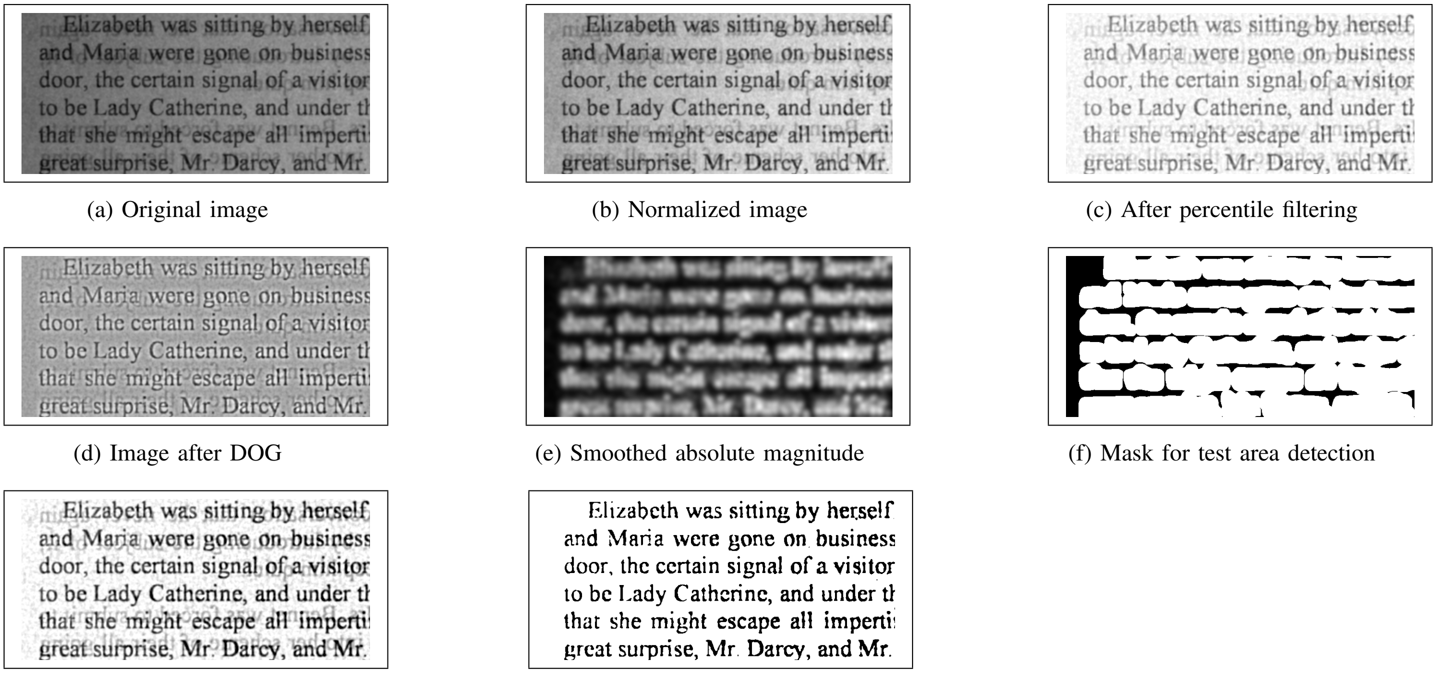
(h) w=30 , k=0.5

Fig. 1: The left column of the figure comprising of (a, c, e, g) shows the left image of the stereo pair with Savoula binarization evaluated using different k values with a fixed window size of 30. The right column of the figure comprising of (b, d, f, h) shows the right image of the stereo pair with Savoula binarization evaluated using different k values with fixed window size of 30.

a global threshold for whole of the page known as global binarization methods. On the other hand, the threshold can also be determined using only the statistics determined by a local window centered around the pixel which is being thresholded. A detailed discussion about the advantages and disadvantages of both local and global approaches is discussed in Bukhari et. al. [5] which concludes that global binarization approaches, e.g. Otsu [10] shows a suboptimal performance for camera captured documents. It is due to the fact that there are certain variations, which appear only in specific parts of the image, e.g. a page might contain a defocused region and another could be illuminated differently, whereas other areas might have other geometric distortions. In contrast local methods can adapt themselves, depending upon the image characteristic of the local region. These methods could further be divided into two categories. The first category of methods is pixel-based and the other category is content-based [5]. In pixel based methods text and non-text regions are treated equally for determining the threshold used for local binarization. In content based methods the text and non-text regions are identified and different thresholds are used for text and non-text regions. The threshold in such methods is fixed for all the text regions. These method improve the performance, but the threshold is not adapted in accordance with background properties.

The proposed method in this paper takes into consideration the background statistics based on percentile filters [14]. So, this method can be categorized as a pixel based binarization method according to earlier classification. One very widely used example of percentile filter is median filter [14] which equals fifty (50) percentile. The percentile filters are generally categorized as ordered rank filters [15]. Rank order filter are very general and can be used to approximate other filters, e.g. median filter, as mentioned above, or morphological filters[16]. There are many ways how a percentile filter can be calculated. There are some methods which are specifically tailored for the images. One such comparison has been given by Duin et. al. [17].

This paper describes work based on the percentile filter for



(a) Original image (b) Normalized image (c) After percentile filtering

(d) Image after DOG (e) Smoothed absolute magnitude (f) Mask for test area detection

(g) After black and white clipping (h) Final binarized image

Fig. 2: The steps of binarization are illustrated in the figures above. **(a)** Original image **(b)**Normalized version of the image shown in (a) according to the equation (10) **(c)** Resultant image after applying percentile filter to the normalized image shown in (b) according to the equation (11) **(d)** Resultant image after applying difference of Gaussian (DOG) to the image shown in (c) according to the equation (12) **(e)** The absolute values of the image shown in (d) with the smoothed with a Gaussian filter according to equation (13) **(f)** Binary mask depicting the text area from which the percentile values are calculated according to the equation (14) **(g)** Resultant image after applying the lo and hi percentile score to the image shown in (c) according to the equation (16) **(h)** Final image after the thresholding according to the equation (17)

binarization which works well both on focused and defocused images. This method also works well on monocular images with defocused parts.

III. Binarization Using Percentile Filter

Binarization using percentile filters starts with estimating the background at each location in the image. In a sense we are calculating a whole new image which is the background of the image based on percentile. First we define the percentile filter and after that the details and fast implementation are discussed and this section concludes with binarization details using percentile filters.

A. Percentile Filter

This algorithm has originally been proposed by [14]. We select a window of a certain size, defined by the user, and we calculate the histogram of the window. The window is defined as follows:

$$w(x,y) = (I_{ij})_{x-dx \le i \le x+dx, y-dy \le j \le y+dy} \quad (1)$$

where x and y denotes the location of the pixel at the center of the window and, dx and dy denote the size of the window both in x and y directions respectively.

Let us define the bounds of our window with the following sets

$$s_1 = \{x - dx \le i \le x + dx\} \quad (2)$$

$$s_2 = \{y - dy \le j \le y + dy\} \quad (3)$$

Now let us rewrite equation (1) with a single index.

$$w(x,y) = \{I_q \mid q = (i,j) \in s_1 \times s_2\} \quad (4)$$

To be able to calculate the percentile, let us sort the values in the window represented in equation (4) by defining an ordering function:

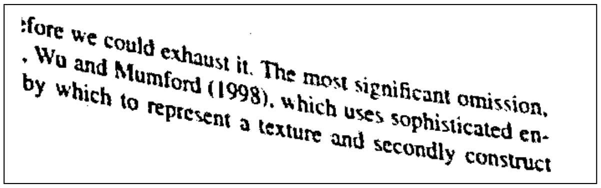
$$ord(a,b) = \begin{cases} 1, & \text{if } a > b \\ \text{-1}, & \text{else} \end{cases} \quad (5)$$

Let the number of pixels in the window be n and we define the following sequence:

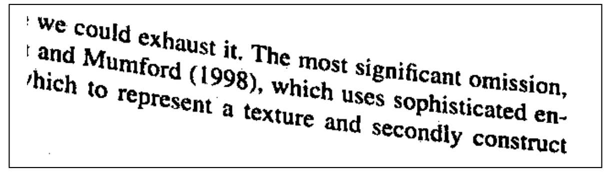
$$ws(x,y) = (I_k)_{0 \le k < n} \text{ such that } \forall k : ord(I_k, I_{k+1}) < 0 \quad (6)$$

The index of the percentile is given as follows:

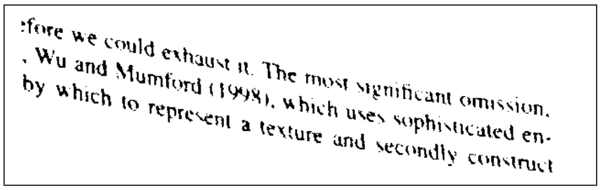
$$i_p = p \times n/100 \text{ where } (0 \le p \le 100) \quad (7)$$



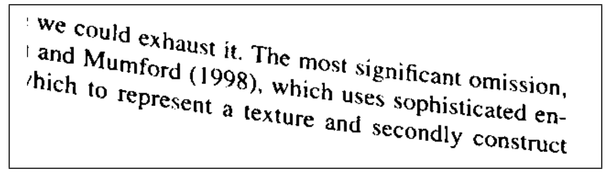
(a) p=70, w=11 , t=0.88



(b) p=70, w=11 , t=0.88



(c) w=30 , k=0.2



(d) w=30 , k=0.2

Fig. 3: The upper row (a, b) shows the left and right image of the stereo pair binarized using percentile filter with the same set of parameters, i.e. (p=70, w=11 , t=0.85). The lower row (c, d) shows the left and right image of the stereo pair binarized using Savoula with the same set of parameters, i.e. (w=30, k=0.2). The percentile filter performs better on both left and the right images of the stereo pair with same set of parameters.

where i_p denotes the index of the value selected as a percentile, p is the required percentile and n is denoting the total number of elements.
Combining equation (6) and equation (7) we define the value percentile for our window

$$ws_p(x, y) = I_{i_p} \tag{8}$$

where ws_p denotes the value of the p^{th} percentile of the window centered around (x, y). It is important to note that all the things have been shown above for one window centered at (x, y). This procedure will be repeated for whole of the image. For determining the percentile of the pixels near the boundary, reflecting boundary conditions are used, i.e. the image has been mirrored to handle indices lying outside the image. An efficient implementation of the percentile filter based on histograms has been discussed in [17].

B. Binarization

A simple method of binarization using percentile filter is as follows: Let f be our original image and the domain of the image is all gray level values, i.e. $f(x, y) \in [0, 255]$ Let g be the background image estimated for each value based on percentile filters at every location (x, y) and the domain of the image corresponds to only two levels, i.e. $g(x, y) \in \{0, 255\}$. The background image is computed according to the procedure that has already been described. The thresholding has been done as follows:

$$o(x, y) = \begin{cases} 255, & \text{if } f(x, y) < t * g(x, y) \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where t is the parameter, which is used to determine that whether a pixel is foreground or background, depending on the similarity of the pixel, and the background, which has been estimated using percentile filter.

A more complicated version of the binarization procedure is described below step by step and is illustrated with Figure. 2 The image I is normalized in the range between 0 and 1 as follows

$$I_n = (I - I_{min})/I_{max} \tag{10}$$

A test image and its normalization have been shown in the Figure. 2a and Figure. 2b. We apply percentile filter with value of percentile p and window of size w

$$I_{p_w} = (P_{p_w}) * I_n \tag{11}$$

Applying percentile filter on the normalized image shown in 2b results in the image shown in Figure. 2c. The value used of percentile filter is 80.

For selecting the low and high thresholds, the image is then enhanced using difference of Gaussian followed by a thresholding with a fixed value producing a binary image which is dilated. The first step is the image enhancement. Let σ_1 and σ_2 be the standard deviation of the successive levels which have been selected heuristically

$$I_g = (I_{p_w})_{\sigma_1} - (I_{p_w})_{\sigma_2} \tag{12}$$

The result of applying difference of Gaussian to Figure. 2c is shown in Figure. 2d. The values 0 and 5 are used for the parameters σ_1 and σ_2 respectively.

The magnitude of the resultant image I_g is further smoothed with σ_2

$$I_s = (((I_g)^2)_{\sigma_2})^{0.5} \tag{13}$$

The Figure. 2e shows the result of applying the above mentioned smoothing to the image shown in Figure. 2d.

For estimating the white and black clipping percentile a mask over the text area has been generated. For this purpose the resultant image from the step above is first thresholded with a constant value 0.3 to produced a binary image and the in the next step the binary image is dilated with a structuring element B

$$I_d = I_s \oplus B \tag{14}$$

The mask results in applying the above operations on 2e is shown in Figure. 2f. The length used for the structuring element is 10 both in the x and y directions.

We use image I_d to mask I_{p_w} i.e

$$m(x,y) = \begin{cases} I_{p_w}(x,y), & \text{if } I_d(x,y) = 1 \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

This means that the values, which are masked out, are not used for the calculation of the percentile score.

Let p_b and p_w be the black and white clipping percentile which are heuristically selected. Let (lo) and (hi) be the percentile score calculated based on the black and white clipping percentile respectively calculated from the masked image. We use the calculated percentile scores on the image I_{p_w} as follows

$$I_f(x,y) = \frac{I_{p_w}(x,y) - lo}{hi - lo} \quad (16)$$

For calculating the percentile scores for our image shown in Figure. 2e we used he values 5 and 90 for p_b and p_w respectively. The resultant image is shown in Figure: 2g after applying it to the image shown in Figure. 2c.

The binary image is produced by thresholding as follows

$$I_b(x,y) = \begin{cases} 0, & \text{if } I_f(x,y) > t \\ 255, & \text{otherwise} \end{cases} \quad (17)$$

The resultant binary image is shown in the Figure. 2h for $t = 0.55$.

IV. Experiments and Results

First we consider the stereo images for comparing binarization results. Figure 3 shows the result of the percentile filter in the first row (Figures 3a and 3b) for the stereo image pair considered in Figure 1. The second row contains the result for Sauvola binarization of the same image pair. Figures in the second row, i.e. (3c and 3d) are the same as (1c and 1d) and shown here for comparison purposes. The results show that the percentile filter for both images perform better than Sauvola because the binarization is almost the same. This is essential for stereo matching. The Sauvola binarization performs well for the focused image as can be seen in Figure 3d, but for the defocused image the quality is degraded and it might not help stereo matching for finding reliable matches.

In order to observe the effect of percentile filter for the blurred image restoration, we took a monocular image and its ground truth. The image has been convolved with an isotropic Gaussian for several values of standard deviation ranging from 0 to 5. Then the image is binarized using the percentile filter and the results are shown in Figure. 5 for the visual inspection. The full effect of the measures [18] has been shown in the Fig. 4. It can be observed that the percentile filter is robust against the blurring effect which could either be caused by the stereo cameras or in general by a single camera.

Furthermore, the proposed method has also been evaluated on monocular document images. The complicated version of binarization which helps us preserving the character shapes better. The dataset consists of 25 degraded images each of size 2000×500. We compare the results of our approach with standard Savoula binarization. For finding the best parameters for both of the methods we used FMeasure for error evaluation. The method of Savoula binarization has two parameters k and w. A grid search over an interval $[0.1, 0.35]$ with step size of 0.1 and $[3, 53]$ with step size of 2 respectively for k and w has been performed. The best values are 0.17 and 15 for k and w respectively. As we can see in the Table I that the performance of both methods for FMeasure is comparable and in fact Savoula performs a bit better than our method. On the other side, for OCR error measure, our method performs much better than Savoula because it is able to preserve the character shapes better which is very useful for practical applications. We used ocropus[19] for the OCR error measures. A sample binarized image is shown in Fig. 6.

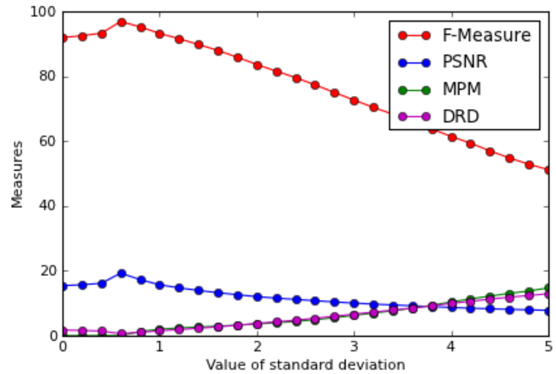


Fig. 4: Measures for the blurred image

Measure	Savoula	Percentile
FMeasure	92.07	89.39
OCR error (edit distance)	69.29	31.40

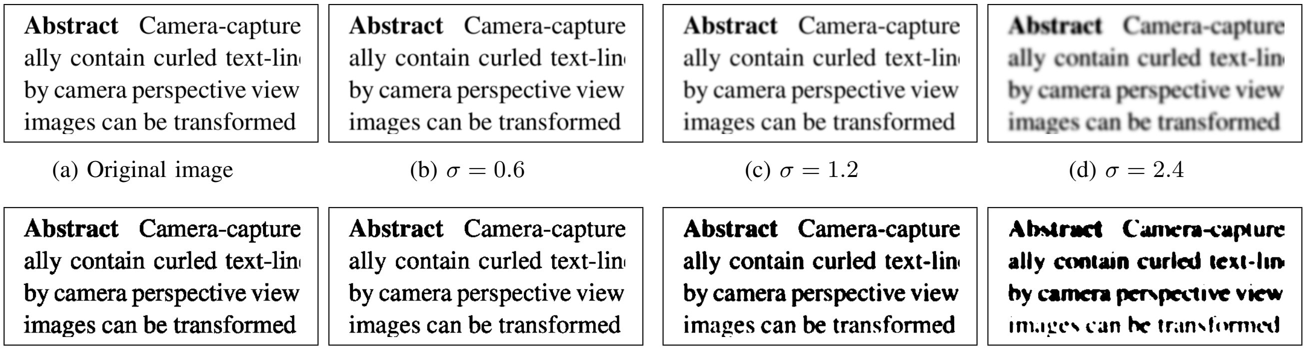
TABLE I: Comparison of proposed method with Savoula Binarization. While Savoula performs a bit better for FMeasure evaluation, our method performs far better for OCR-Based evaluation

V. Conclusion

A simple local binarization method is presented. We have shown applicability of the proposed binarization method on stereo document images. Compared to conventional binarization approaches, the main benefit is that the same parameters can be used for both images of the stereo image pair and still produce good binarization results. We have also shown that performance is comparable to standard methods for Fmeasure and our methods outperforms standard Savoula method by a big margin for OCR-Based evaluation.

References

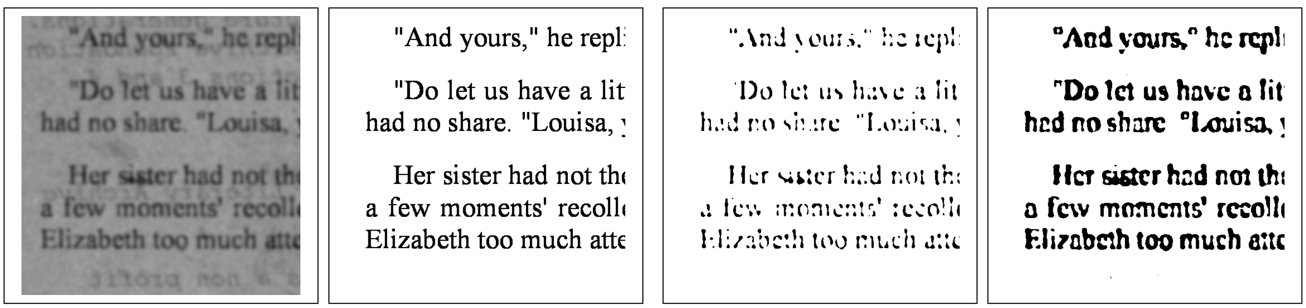
[1] M. Afzal, M. Krämer, S. Bukhari, F. Shafait, and T. Breuel, “Improvements to Uncalibrated Feature-based Stereo Matching for Document Images by using Text-Line Segmentation ,” in *Proceedings of the 10th IAPR International Workshop on Document Analysis Systems*, 2012.



(a) Original image (b) $\sigma = 0.6$ (c) $\sigma = 1.2$ (d) $\sigma = 2.4$

(e) Ground Truth (f) Binarization of (b) (g) Binarization of (c) (h) Binarization of (d)

Fig. 5: The upper row (a, b, c, d) shows the original image and the blurred image with standard deviation of 0.6, 1.2 and 1.4 respectively. The lower row (e, f, g, h) shows the ground truth and the restored images corresponding to their smoothed counter parts. The values of the parameters used for the binarization are (p = 33, w = 75, t = 0.78).



(a) Original Image (b) Ground Truth (c) Savoula Binarization (d) Percentile Filter

Fig. 6: The comparison of the Savoula with percentile filter. The characters shapes are better using percentile filter in comparison to Savoula.

[2] M. Afzal, S. Bukhari, M. Krämer, F. Shafait, and T. Breuel, "Robust Stereo Matching for Document Images Using Parameter Selection of Text-Line Extraction," in *21st Int. Conf. on Pattern Recognition, ICPR12 Tsukuba, Japan,*, Nov. 2012.

[3] M. Krämer, M. Afzal, S. Bukhari, F. Shafait, and T. Breuel, "Robust Stereo Correspondence for Documents by Matching Connected Components of Text-Lines with Dynamic Programming," in *21st Int. Conf. on Pattern Recognition, ICPR12 Tsukuba, Japan*, Nov. 2012.

[4] J. Sauvola and M. Pietikäinen, "Adaptive Document Image Binarization," *Pattern Recognition*, vol. 33, pp. 225–236, 2000.

[5] S. S. Bukhari, F. Shafait, and T. Breuel, "Adaptive binarization of unconstrained hand-held camera-captured document images," *Journal of Universal Computer Science*, vol. 15, no. 18, pp. 3343–3363, 12 2009.

[6] K. Sobottka, H. Kronenberg, T. Perroud, and H. Bunke, "Text extraction from colored book and journal covers." *IJDAR*, no. 4, pp. 163–176, 2000.

[7] C.-M. Tsai and H.-J. Lee, "Binarization of color document images via luminance and saturation color features," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 434–451, 2002.

[8] E. Badekas, N. A. Nikolaou, and N. Papamarkos, "Text localization and binarization in complex color documents," in *MLDM Posters*, 2007, pp. 1–15.

[9] H. Orii, H. Kawano, H. Maeda, and N. Ikoma, "Text-color-independent binarization for degraded document image based on map-mrf approach," *IEICE Transactions*, vol. 94-A, no. 11, pp. 2342–2349, 2011.

[10] N. Otsu, "A threshold selection method from gray-level histograms," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, no. 1, pp. 62 –66, jan. 1979.

[11] B. Gatos, I. Pratikakis, and S. J. Perantonis, "Adaptive degraded document image binarization," *Pattern Recogn.*, vol. 39, no. 3, pp. 317–327, Mar. 2006.

[12] F. Shafait, D. Keysers, and T. Breuel, "Efficient implementation of local adaptive thresholding techniques using integral images," in *Proceedings of the 15th Document Recognition and Retrieval Conference , Part of the IST/SPIE International Symposium on Electronic Imaging, January 26-31, San Jose, CA, USA*, vol. 6815. SPIE, 1 2008.

[13] D. Rivest-Hénault, R. F. Moghaddam, and M. Cheriet, "A local linear level set method for the binarization of degraded historical document images," *IJDAR*, vol. 15, no. 2, pp. 101–124, 2012.

[14] B. Justusson, "Median filtering: Statistical properties," in *Two-Dimensional Digital Signal Prcessing II*, ser. Topics in Applied Physics. Springer Berlin / Heidelberg, 1981, vol. 43, pp. 161–196.

[15] G. Heygster.

[16] P. Soille, "On morphological operators based on rank filters," *Pattern Recognition*, vol. 35, no. 2, pp. 527 – 535, 2002.

[17] R. Duin, H. Haringa, and R. Zeelen, "Fast percentile filtering," *Pattern Recognition Letters*, vol. 4, no. 4, pp. 269 – 272, 1986.

[18] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "Icdar 2011 document image binarization contest (dibco 2011)," in *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, sept. 2011, pp. 1506 –1510.

[19] T. M. Breuel, "The OCRopus Open Source OCR System," http://code.google.com/p/ocropus/.