

Entropy-based Locally Adaptive Thresholding for Image Segmentation

Min Zhang^a, Shenghui Cheng^b, Xinhua Cao^c, Huai Chen^{d,*}, Xiaoyin Xu^{e,*}

^a*College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang 310027, China*

^b*Westlake University, Hangzhou, Zhejiang, China*

^c*Department of Radiology, Boston Children's Hospital, Boston, MA 02115, USA*

^d*Department of Radiology, First Affiliated Hospital of Guangzhou Medical University, Guangzhou, Guangdong, China*

^e*Department of Radiology, Brigham and Women's Hospital, Boston, MA 02115, USA*

Abstract

We presented a local thresholding method that is adaptively computed to segment images. The construction of the locally adaptive threshold mimics how human visual system interprets images, that is, identifying homogeneous regions in images in the presence of noise, clutters, and low contrast. Using a sliding window to raster scan an image, the method uses local statistics to measure homogeneity of pixel distribution in the window. The locally adaptive threshold is determined by evaluating a pixel against the local mean and an offset term, which is built by the local standard deviation and entropy. In implementation, except for the size of the sliding window, the method involves no parameters and, thus, requires no user input. For this reason, the method has a consistent and robust performance across users. Test results on different types of images demonstrated the robust performance of the method in separating fine details about an image and suppressing false positives. In practice, the method is extendable to parallel processing as the calculations are locally determined.

Keywords: segmentation, entropy, adaptive thresholding, standard deviation

*Corresponding author

Email address: huaichen1977@163.com (Huai Chen), xxu@bwh.harvard.edu (Xiaoyin Xu)

1. Introduction

Segmentation is often the first step in image processing and plays an important role in determining the performance of overall image quantification, classification, analysis, and interpretation. Segmentation techniques are generally of two types, global thresholding and local thresholding. Global thresholding typically finds one threshold to apply over the whole image. An example is the Otsu method [1], which searches for a threshold over the histogram of an image to maximize the inter-group variance of the foreground and background and is non-parametric. Based on the original Otsu method, many techniques have been proposed. Feng et al. developed a 3D multi-scale thresholding method [2]. In the method, an image is iteratively segmented by the 3D Otsu method and, after iteration, the result is filtered by a Laplacian operation to create a smoothed version of the image for the next iteration. The final result is a pooled summary of all the intermediate segmentation outcomes. As the search for a threshold in 3D space can be computationally slow, Bhandari et al. presented a 3D Otsu method that uses a stochastic meta-heuristic optimization algorithm to facilitate the search for the threshold vector in the 3D space for segmentation [3]. Global thresholding methods generally depend on analyzing the histogram of images for segmentation. For example, Kapur et al. developed a method to find the threshold that results in maximum entropy between the foreground and background distributions [4]. Rosenfeld and De La Torre designed a histogram concavity analysis method by identifying 'shoulders' and valleys of an histogram as the potential threshold candidates and used a convex hull to find the deepest valley as the final threshold [5]. Sezan designed an automatic peak detection method for histogram-based image segmentation [6]. In the method, a peak is either derived from the histogram or the cumulative distribution function. Recently, Wang et al. developed a flexible thresholding method using histogram to compute the threshold based on the slope difference distribution [7]. The method has two steps where the first step clusters pixels based on the peaks of slope difference distribution. The second step selects the threshold based

on the valleys of the slope difference distribution. Local thresholding usually uses a sliding window to raster scan an image and calculates some local statistics for assigning the central pixel to the foreground or background. Examples of local thresholding include Niblack method [8], Sauvola method [9], Bernsen method [10], Wolf method [11], and others. In the Niblack method, the local threshold is determined as the local mean minus a constant times the local standard deviation [8]. In the Sauvola method, the threshold is computed in a similar way to the Niblack method but uses the average of the highest and lowest gray value of an image to modify the local standard deviation to emphasize segmentation of very dim and very strong pixel [9]. In the Bernsen method, the threshold is determined as the mean of local maximum and minimum values in the sliding window [10]. In the Wolf method, the threshold is determined by normalizing the contrast and the mean gray value of the image [11]. One common characteristics of the above locally adaptive techniques is that, in addition to the window size, they require appropriate setting of parameters in threshold calculation, whereas the choice of parameters may have profound impact on the output. The need for setting parameters increases the risk for inconsistency in practical applications of the techniques as users may elect to use different parameters. From this perspective, it is beneficial if a method has as few parameters as possible. Another common characteristics about the above methods is that the thresholds are centered around the local mean, plus or minus some adjustment terms. Often, the adjustment terms are related to the local standard deviation as it measures how tightly or distantly the pixels are distributed around the mean. In addition to strictly global and local thresholding methods, there are techniques that incorporate both global and local information. For example, Cho et al. presented a non-iterative method that utilizes a range covariance matrix as the global attribute and sum of the variances of the chromatic components as the local attribute to construct a mean-shift-based segmentation approach [12]. Threshold can also be determined based on the assumed noise model of images. For example, Chien et al. proposed a method, assuming the background of an image is of multi-layers and the noise has a zero-mean Gaus-

sian distribution, to detect objects in the foreground in videos [13]. Other types of segmentation techniques include clustering [14], Markov process [15, 16], active contours [17], watershed [18, 19], graph-based methods [20, 21, 22], particle swam optimization [23, 24], Gaussian mixture model [25], and, in recent years, deep learning models [26, 27, 28]. For example, Sun et al. presented a method called Gaussian dynamic convolutions for interactive single image segmentation, in which the method randomly selects spatial sampling area according to the Gaussian distribution offsets to aggregate features in various scales and then the features are fed to a neural network as the input to complete the segmentation process [29]. In this work, we present a locally adaptive thresholding method that computes a threshold within a sliding window centered at the pixel for consideration. The method is based on three local statistics, namely, the local mean, which is used as the central tendency of pixel distribution, the local standard deviation, which measures the dispersion of pixel intensities, and local entropy, which measures the randomness of the pixels within the sliding window. The threshold is adaptively calculated by measuring how far a pixel deviates from the central tendency plus an offset measuring the diversion of the local neighborhood. Compared with existing techniques, the distinctive feature of the new method is two-fold. First, it has a non-parametric design as the method involves no parameters, except for the size of the sliding window. Thus, the method has a consistent and robust performance across users. Second, it does not make any assumptions about the statistical distribution of the foreground and background of images. Thus, the method has a wide applicability over different types of images. Our tests and comparisons show that the method can achieve good performance across a large variety of images. The paper is organized as follows. Section 2 describes the method. Section 3 presents the test results and comparisons with some existing methods. Section 4 gives discussion and conclusions.

2. Algorithm

Given an image I , our method uses a sliding window W to raster scan the whole image. At each pixel location (x, y) , we calculate the local mean $\mu(x, y)$, standard deviation $\sigma(x, y)$, and entropy defined $e(x, y)$ within W , such that

$$\mu(x, y) = \frac{\sum_i s(i)}{N} \quad (1)$$

$$\sigma(x, y) = \frac{1}{\sqrt{N}} \sqrt{\sum_i (s(i) - \mu(x, y))^2} \quad (2)$$

$$e(x, y) = -\sum_j p_j \log p_j \quad (3)$$

where N is the number of pixels in W and p_j is the probability that pixels within W taking value j . Assume the image has a dark foreground on a bright background, then we define a local threshold based on the local mean and an locally adaptive offset such that

$$T(x, y) = \mu(x, y) - \Delta(x, y) \quad (4)$$

where $\Delta(x, y)$ is the offset defined as

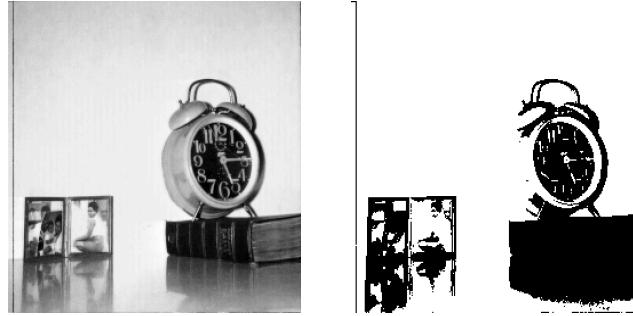
$$\Delta(x, y) = \frac{e^2(x, y)}{\sigma(x, y)}. \quad (5)$$

A pixel $s(x, y)$ is declared foreground if it is less than $T(x, y)$. For images that have a bright foreground on a dark background, we can revert the brightness of the images and they apply the threshold. From Eq. (4) and (5) we note that the threshold is adapted to the local mean $\mu(x, y)$, which is offset by $\Delta(x, y)$. The offset $\Delta(x, y)$ is a measurement about homogeneity of the pixels within W . By definition, the local entropy $e(x, y)$ is a measurement about randomness of pixels within W . A large $e(x, y)$ corresponds to a high degree of inhomogeneity of the pixels, while a small $e(x, y)$ means that pixels tend to repeat themselves in grayscale values. Standard deviation $\sigma(x, y)$, on the other hand, measures spread of pixel values whereas a large σ indicates that pixel values spread out about the mean. So we can see that, though entropy and standard deviation both measure the distribution of pixels in W , they obtain the measurement

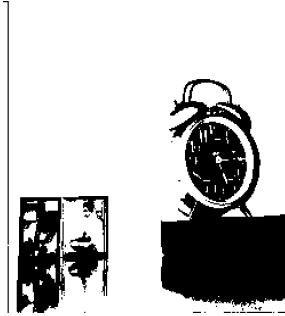
from different perspectives. Entropy focuses on the diversity of the pixels while standard deviation focuses on how spread the pixels are. Using entropy and standard deviation together we can assess homogeneity within the window.

Figure 1 shows the result of using the new method to segment an image. For this example, the original image has an uneven background and a foreground with some fine details, Figure 1(a). Though the example image has a seemingly simple composition of foreground and background, its segmentation is not easy. Figure 1(b) shows the result by the global thresholding Otsu method. The result by an existing local adaptive thresholding method, the Niblack method, is shown in Figure 1(c). We note that the Otsu method missed some details such as the full depiction of the digits on the clock. The Niblack method clearly generated much false positives in the background area. The segmentation by the new method is shown in Figure 1(d). For both the Niblack method and the new method we used a same window W of 20×20 . We can observe that the new method is able to correctly classify the foreground, though it created few scattered false positives to the top of the image. These false positives can then be removed by post-processing operations. To have an insight into how the threshold of new method is determined, Figure 2 shows the maps of local mean μ and the offset Δ of Figure 1. From the colorbars of the two maps, we note that the offset has a much smaller range than the local mean. However, this offset, though small, plays an important role in impacting the segmentation outcome. As a validation, we display the result of just using the local mean as the threshold to segment the image such that if a pixel $s(x, y)$ is less than $\mu(x, y)$, it is declared as the foreground. Otherwise, it is assigned to the background. The result is shown in Figure 3, from which we note that it has many false positives, proving that offset Δ can effectively reduce the amount of incorrect segmentation.

From the design of the new method, the only parameter that it involves is the size of sliding window. Figure 4 shows how the segmentation result changed for different window sizes. From what the figure shows and result of Figure 1(d), we see that, even as the window size changed from 11×11 to 51×51 , the



(a)



(b)

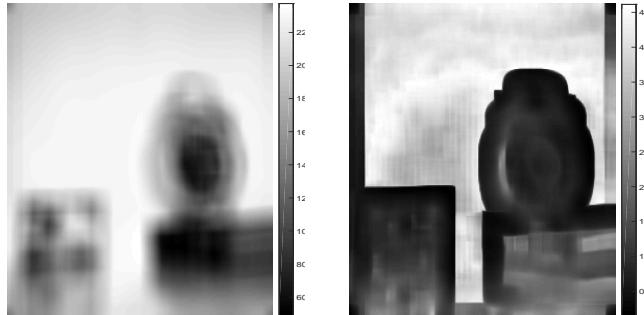


(c)



(d)

Figure 1: (a), Image "clock". (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 21 \times 21$ for the Niblack and new methods.



(a)



(b)

Figure 2: (a), Local mean map of image "clock". (b), Local offset map.

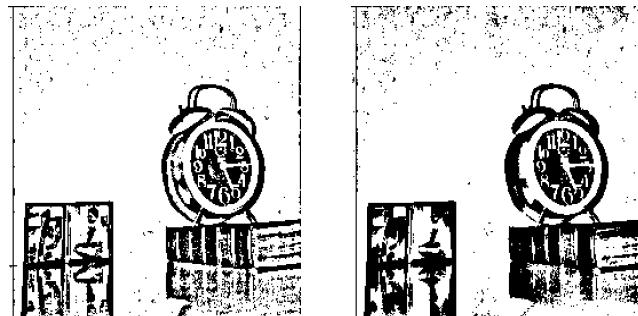


Figure 3: Segmentation result if we only used local mean as the threshold.

segmentation results largely remain unchanged, pointing to the robustness of the method.

3. Experiments

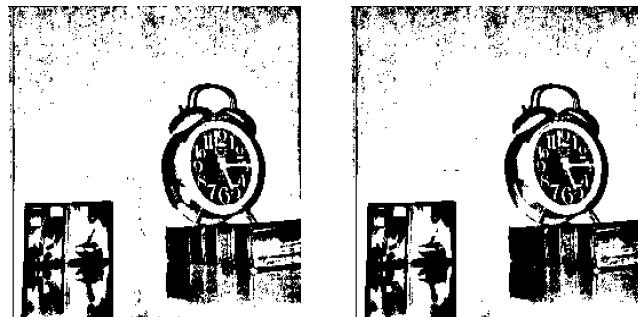
As an example, we applied the Otsu method, Niblack method, and the new method to the test image, Figure 5(a). This test image has a very rich texture and fine details. The three segmentation results are shown in Figure 5(b-d). It is seen that the Otsu method under-segmented the image, especially around the top half of the picture. The Niblack method created many false positives. The new method gave the best result among the three methods. For detailed comparison, we show a zoomed-in region of Figure 5(a) in Figure 6(a). From the results, we note that some fine details such as the scales on the ruler are missed in the Otsu result (Figure 6(b)), while the Niblack result has substantial over-segmentation (Figure 6(c)). The new method, on the other hand, created a well-balanced segmentation showing the details without generating over-segmentation, Figure 6(d). We next applied the three methods to a finger-print example. The original image, shown in Figure 7, has both well-separated ridges and adjacent ridges. The Otsu result of Figure 7(b) had good results on segmenting the well-separated regions but had trouble separating the central region where the ridges were close to each other. The Niblack result of Figure 7(c)



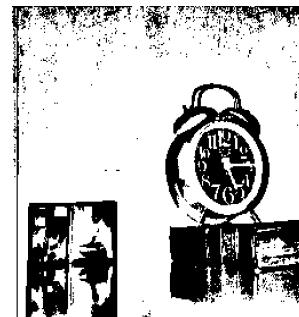
(a)



(b)



(c)



(d)

Figure 4: Using different window sizes to segment test image "clock". (a), $W = 11 \times 11$. (b), $W = 31 \times 31$. (c), $W = 41 \times 41$. (d), $W = 51 \times 51$.



(a)



(b)

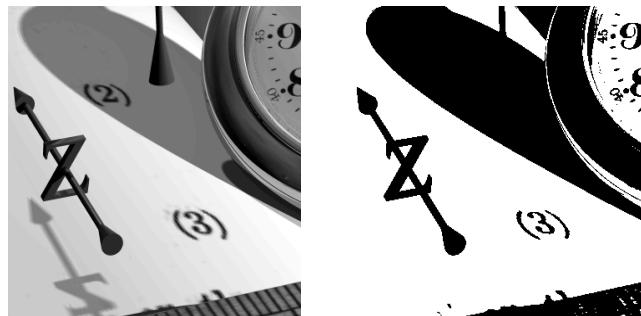


(c)



(d)

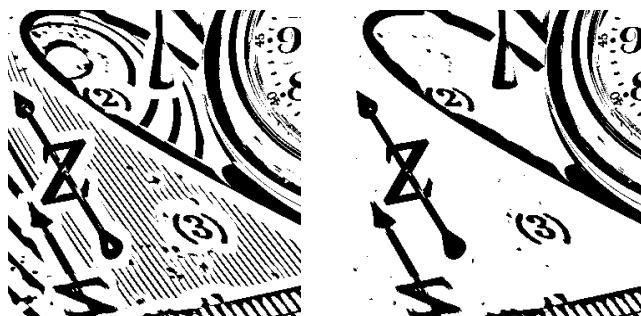
Figure 5: (a), Image "watch". (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 21 \times 21$ for the Niblack and new methods.



(a)



(b)



(c)



(d)

Figure 6: (a), A zoomed-in region of test image "watch". (b), Result of the Otsu method.
(c), Result of the Niblack method. (d), Result of the new method.

had some under-segmentation in the central region. Result by the new method, shown in Figure 7(d), had better performance in segmenting the ridges. For better visualization, a zoomed-in display of the original image is shown in Figure 8(a). The corresponding segmentation results are shown in Figure 8(b-d). We can better observe that the Otsu result could not separate some adjacent ridges while the Niblack result almost caused broken ridges, especially within the red rectangle box on the image. The new method was able to separate the ridges without causing incorrect broken parts. Another test image of fin-



Figure 7: (a), A fingerprint example. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 21 \times 21$ for the Niblack and new methods.

gerprint is shown in Figure 9(a). The Otsu result over-segmented some areas,

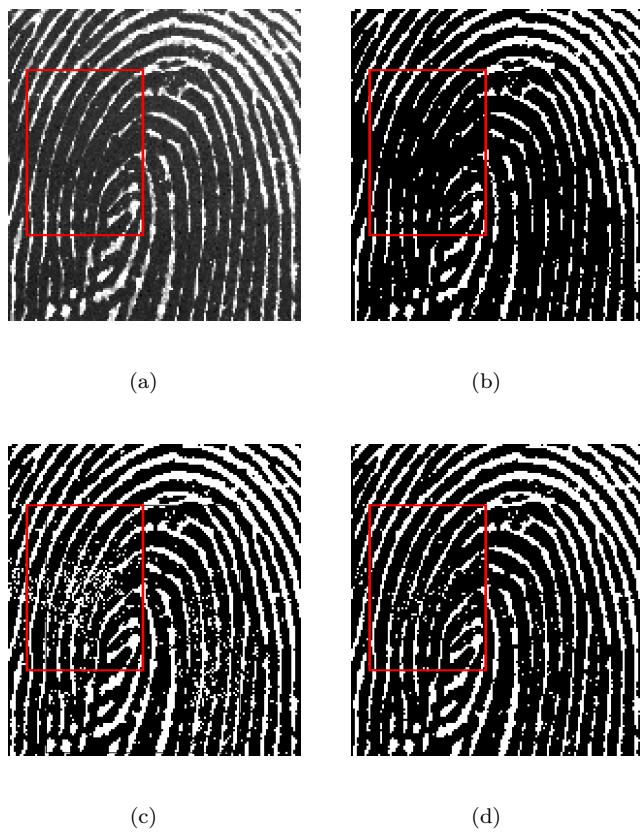


Figure 8: (a), Zoomed-in part of fingerprint example of Figure 7. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 21 \times 21$ for the Niblack and new methods. Rectangle box highlights an area of adjacent ridges.

such as the one enclosed in the rectangle box, Figure 9(b). The Niblack result under-segmented some ridges, including the area enclosed in the rectangle box, Figure 9(c). The result by the new method is shown in Figure 9(d), from which we note again that the new method generated the best result as it could segment closely adjacent ridges.

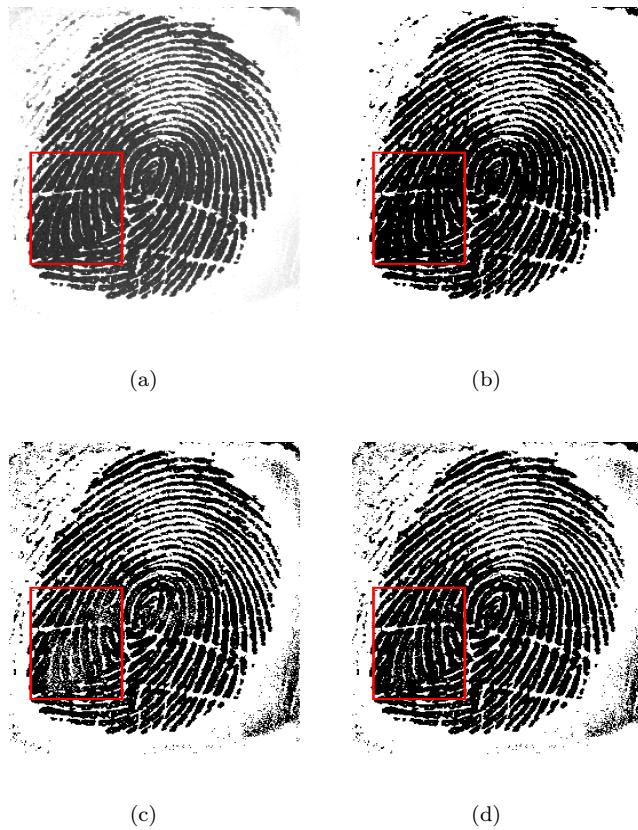


Figure 9: (a), A fingerprint example. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 31 \times 31$ for the Niblack and new methods. Rectangle box highlights an area of adjacent ridges.

We also compared the three methods on segmenting scanned handwriting and documents. An example is shown in Figure 10(a). We can observe that the original image has an even illumination with its center having a brighter background than the peripherals. Though the characters are clearly shown

in the scanned image, their segmentation by the Otsu method was affected by the uneven illumination, Figure 10(b). The Niblack method gave better segmentation, as the uneven background was largely eliminated, Figure 10(c). The new method gave the best result as it has fewest clutters in the result and extracted the complete characters, Figure 10(d). Figure 11(a) shows a scanned

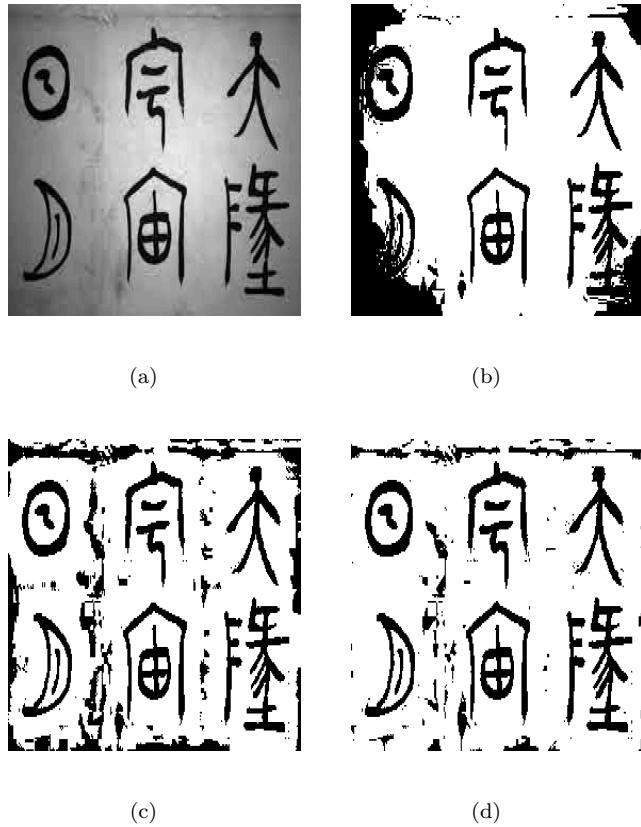


Figure 10: (a), A calligraphy example. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 31 \times 31$ for the Niblack and new methods.

document [30], which, though of uneven background, is highly discernible by human eyes. However, its segmentation is not easy. From Otsu's result of Figure 11(b), we note that some letters are missed. An example is the word "(2018 figure)" in the text, which was of a lighter gray scale than the other

text in the original image, was under-segmented. This pointed out a common limitation in global threshold, that is, a region of fairly recognizable foreground may be missed if it is of lower gray value than other parts of the foreground, even though it is of clearly higher gray value than the background. The result by the Niblack method, shown in Figure 11(c), though segmented the missed word of "(2018 figure)", had a significant amount of over-segmentation. We note that the new method gave a very robust and consistent segmentation of the document image, Figure 11(d), as it both segmented the weak foreground and suppressed the noisy background.

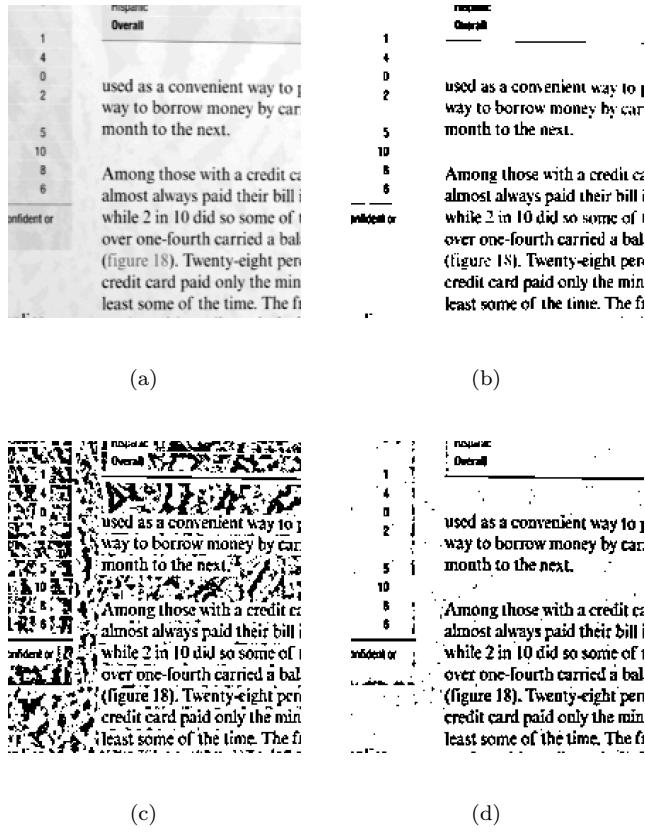


Figure 11: (a), A scanned document. (b), Result of the Otsu method. Note that the word "figure 18", because of its low gray value, was largely missed in the segmentation. (c), Result of the Niblack method. (d), Result of the new method. $W = 7 \times 7$ for the Niblack and new methods.

We also tested the three methods on medical images. Figure 12(a) shows an X-ray image, which has a bright foreground on a dark background. So, for processing the image, we inverted the image by using 255, the maximum value of an 8-bit image, to subtract the original image. The Otsu result, while segmenting the hands and arms area, missed the individual bones, Figure 12(b). The Niblack result generated a large amount of over-segmentation, Figure 12(c). The result by the new method accurately depicts the bone structure of the arm and hand, Figure 12(d). Comparing Figure 12(c) and (d), we note that the new method better segmented the eight bones in the wrists and suppressing incorrect segmentation in the space between the two bones in the forearm, the radius and ulna bones. Figure 13(a) displays a digital subtraction angiogram in the sagittal view of the brain. The global thresholding result by the Otsu method is shown in Figure 13(b), which clearly under-segmented many blood vessels. The Niblack method, on the other hand, caused a large amount incorrect over-segmented noise from the background, Figure 13(c). In comparison, the new method gave the best result as it preserved the blood vessels and had a small amount of over-segmented noisy parts, Figure 13(d). Figure 14 shows another example of segmenting DSA image. From the results, again we note that the new method achieved best performance in assigning blood vessels to the foreground while causing a small number of false-positives.

In supplementary materials, we included additional test results and comparisons with more existing techniques, including global thresholding techniques like maximum entropy method [31] and minimum error method [32] and local thresholding techniques like the Bernsen method [10], Phansalkar method [33], and Sauvola method [9].

4. Discussion and Conclusions

We designed a locally adaptive thresholding method that utilizes local mean, local standard deviation, and local entropy to extract foreground from an image while avoiding over-segmentation. Because the method uses local statistics only,

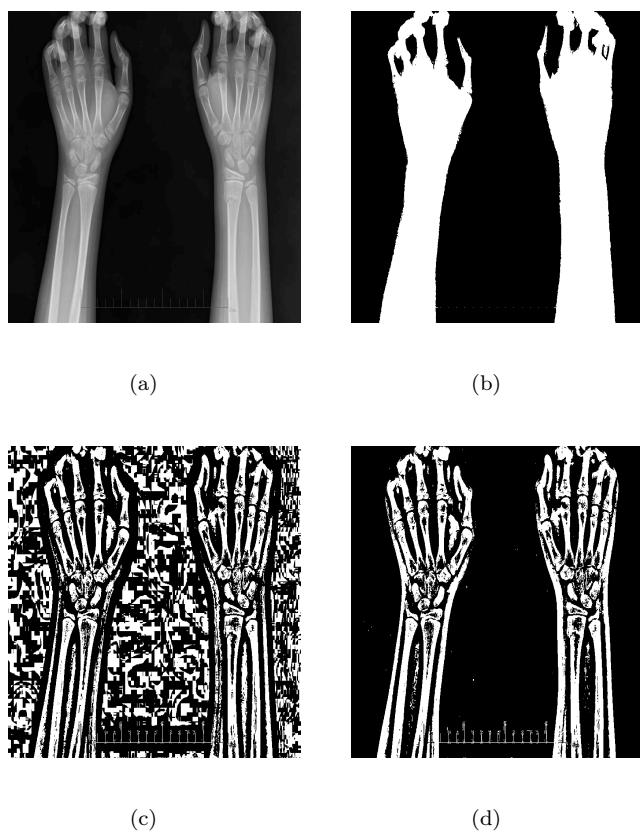


Figure 12: (a), An X-ray image of the forearms, wrists, and hands. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 31 \times 31$ for the Niblack and new methods.

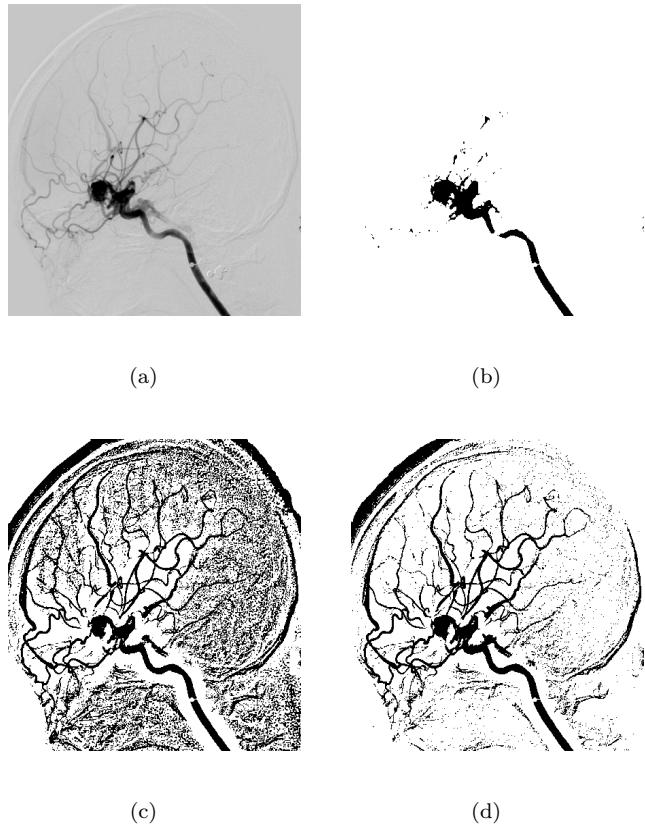


Figure 13: (a), An original digital subtraction angiogram of the brain, showing major and minor arteries. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 31 \times 31$ for the Niblack and new methods.

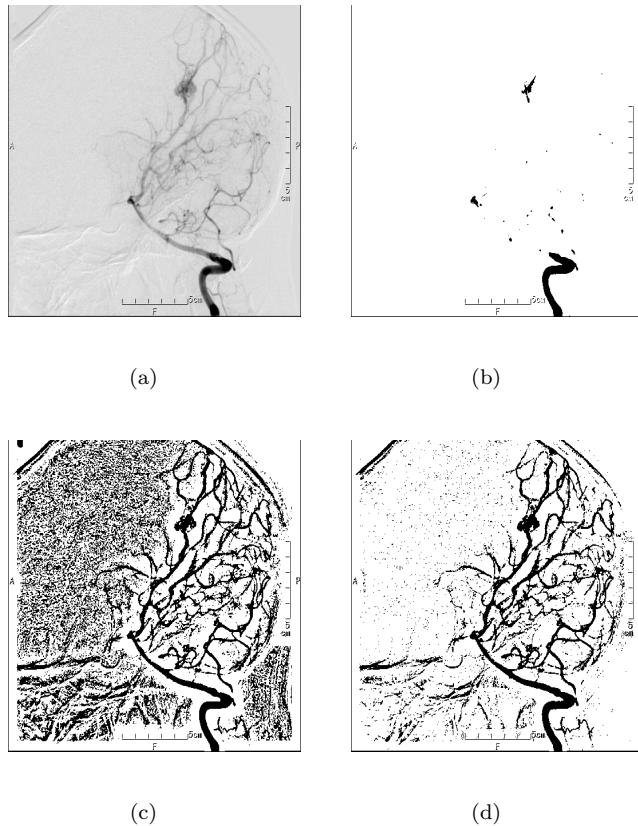


Figure 14: (a), An original digital subtraction angiogram of the brain. (b), Result of the Otsu method. (c), Result of the Niblack method. (d), Result of the new method. $W = 31 \times 31$ for the Niblack and new methods.

the method can be implemented in parallel computing for fast speed. Tests on various types of images and comparisons with existing techniques demonstrated the robustness and accuracy of the new method.

Homogeneity can be perceived by HVS in more than one way and there is not generally accepted definition of homogeneity in image computations. We constructed a measure of homogeneity using local standard deviation and entropy to capture the notion of homogeneity from the perspectives of measuring both the variations in pixel values and randomness in pixel repetitiveness. We note that there could be other metrics for evaluating homogeneity in images and more effort could be dedicated to designing methods mimicking HVS in perceiving images.

A novel contribution of our work is to use standard deviation and entropy to measure homogeneity of the pixels contained in the sliding window. We recognize that standard deviation and entropy are two statistical metrics emphasizing different aspect of a stochastic distribution. Standard deviation emphasizes how far pixels are located in gray values from their mean, while entropy emphasizes how repetitive the pixels are. The offset term in the new method utilizes the ratio of local entropy over standard deviation to adaptively adjust the local mean as the threshold in the sliding window. This adjustment, as shown in our tests, is very powerful in extracting the correct foreground while reducing false positives.

The new method has the following strengths. First, it is virtually parameter-free as the calculation for the locally adaptive threshold involves no parameter while a user only needs to select the size of the sliding window. Second, it can successfully process images with an uneven background and illumination. Third, it has a good noise suppressing capability.

Though the new method has been shown to possess a good capability in suppressing noise during segmentation, for some types of images, some noise is inevitably assigned to the foreground, as can be seen in Figures 13 and 14. In these cases, post-processing such as morphological cleaning can remove the scattered noisy segmentation. For example, in their work, Gupta et al. used

pre-filtering and post-binari-zation denoising as part of the processing steps to segment document images [34]. As the focus of this paper is about the new thresholding method, we did not include pre- and post-processing to assist the segmentation process or further improve the result of segmentation.

References

- [1] N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. Sys., Man. Cyber.* 9 (1) (1979) 62–66.
- [2] Y. Feng, H. Zhao, X. Li, X. Zhang, H. Li, A multi-scale 3D Otsu thresholding algorithm for medical image segmentation, *Digital Signal Processing* 60 (1) (2017) 186–199.
- [3] A. K. Bhandari, I. V. Kumar, K. Srinivas, Cuttlefish algorithm-based multilevel 3-D Otsu function for color image segmentation, *IEEE Transactions on Instrumentation and Measurement* 69 (5) (2019) 1871–1880.
- [4] J. N. Kapur, P. K. Sahoo, A. K. C. Wong, A new method for gray-level picture thresholding using the entropy of the histogram, *Comp Vision Graph. and Image Process.* 29 (3) (1985) 273–285.
- [5] A. Rosenfeld, P. De La Torre, Histogram concavity analysis as an aid in threshold selection, *IEEE Trans. Syst. Man Cybern. SMC-13* (2) (1983) 231–235.
- [6] M. I. Sezan, A peak detection algorithm and its application to histogram-based image data reduction, *Comp Vision Graph. and Image Process.* 49 (1) (1990) 36–51.
- [7] Z. Wang, J. Xiong, Y. Yang, H. Li, A flexible and robust threshold selection method, *IEEE Transactions on Circuits and Systems for Video Technology* 28 (9) (2018) 2220–2232.
- [8] W. Niblack, *An Introduction to Digital Image Processing*, Prentice Hall, 1986.

- [9] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, *Pattern Recognition* 33 (2) (2000) 225–236.
- [10] J. Bernsen, Dynamic thresholding of gray-level images, in: Proc. Eighth Int'l Conf. Pattern Recognition, Paris, 1986.
- [11] C. Wolf, J.-M. Jolion, Extraction and recognition of artificial text in multimedia documents, *Formal Pattern Analysis & Applications* 6 (4) (2004) 309–326.
- [12] H. Cho, S.-J. Kang, Y. H. Kim, Image segmentation using linked mean-shift vectors and global/local attributes, *IEEE Transactions on Circuits and Systems for Video Technology* 27 (10) (2017) 2132–2140. doi:10.1109/TCSVT.2016.2576918.
- [13] S.-Y. Chien, W.-K. Chan, Y.-H. Tseng, H.-Y. Chen, Video object segmentation and tracking framework with improved threshold decision and diffusion distance, *IEEE Transactions on Circuits and Systems for Video Technology* 23 (6) (2013) 921–934.
- [14] H. Huang, F. Meng, S. Zhou, F. Jiang, G. Manogaran, Brain image segmentation based on FCM clustering algorithm and rough set, *IEEE Access* 7 (2019) 12386–12396.
- [15] E. Erdil, S. Yildirim, T. Tasdizen, M. Cetin, Pseudo-marginal MCMC sampling for image segmentation using nonparametric shape priors, *IEEE Transactions on Image Processing* 28 (11) (2019) 5702–5715.
- [16] S. Yousefi, R. Azmi, M. Zahedi, Brain tissue segmentation in MR images based on a hybrid of MRF and social algorithms, *Medical Image Analysis* 16 (4) (2012) 840–848.
- [17] T. F. Chan, L. A. Vese, Active contours without edges, *IEEE Transactions on Image Processing* 10 (2) (2001) 266–277.

- [18] L. Shafarenko, M. Petrou, J. Kittler, Automatic watershed segmentation of randomly textured color images, *IEEE Transactions on Image Processing* 6 (11) (1997) 1530–1544.
- [19] J. M. Gauch, Image segmentation and analysis via multiscale gradient watershed hierarchies, *IEEE Transactions on Image Processing* 8 (1) (1999) 69–79.
- [20] Y. Boykov, G. Funka-Lea, Graph cuts and efficient ND image segmentation, *International Journal of Computer Vision* 70 (2) (2006) 109–131.
- [21] S. Vicente, V. Kolmogorov, C. Rother, Graph cut based image segmentation with connectivity priors, in: *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.
- [22] B. Peng, L. Zhang, D. Zhang, A survey of graph theoretical approaches to image segmentation, *Pattern Recognition* 46 (3) (2013) 1020–1038.
- [23] M. G. Omran, A. Salman, A. P. Engelbrecht, Dynamic clustering using particle swarm optimization with application in image segmentation, *Pattern Analysis and Applications* 8 (4) (2006) 332–344.
- [24] T. R. Farshi, J. H. Drake, E. Özcan, A multimodal particle swarm optimization-based approach for image segmentation, *Expert Systems with Applications* 149 (2020) 113233.
- [25] T. M. Nguyen, Q. M. J. Wu, Fast and robust spatially constrained Gaussian mixture model for image segmentation, *IEEE Transactions on Circuits and Systems for Video Technology* 23 (4) (2013) 621–635.
doi:10.1109/TCSVT.2012.2211176.
- [26] G. Wang, W. Li, M. A. Zuluaga, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, et al., Interactive medical image segmentation using deep learning with image-specific fine tuning, *IEEE Transactions on Medical Imaging* 37 (7) (2018) 1562–1573.

- [27] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (12) (2017) 2481–2495.
- [28] J. Dong, Y. Cong, G. Sun, Y. Yang, X. Xu, Z. Ding, Weakly-supervised cross-domain adaptation for endoscopic lesions segmentation, *IEEE Transactions on Circuits and Systems for Video Technology* 31 (5) (2021) 2020–2033.
- [29] X. Sun, X. Sun, C. Chen, X. Wang, J. Dong, H. Zhou, S. Chen, Gaussian dynamic convolution for efficient single-image segmentation, *IEEE Transactions on Circuits and Systems for Video Technology* In press (2021) 1–1.
- [30] I. Zharikov, P. Nikitin, I. Vasiliev, V. Dokholyan, DDI-100: Dataset for Text Detection and Recognition, in: Proceedings of the 2020 4th International Symposium on Computer Science and Intelligent Control, 2020, pp. 1–5.
- [31] P. K. Sahoo, S. Soltani, A. K. Wong, A survey of thresholding techniques, *Computer Vision, Graphics, and Image Processing* 41 (2) (1988) 233–260.
- [32] J. Kittler, J. Illingworth, Minimum error thresholding, *Pattern Recognition* 19 (1) (1986) 41–47.
- [33] N. Phansalkar, S. More, A. Sabale, M. Joshi, Adaptive local thresholding for detection of nuclei in diversity stained cytology images, in: 2011 International Conference on Communications and Signal Processing, IEEE, 2011, pp. 218–220.
- [34] M. R. Gupta, N. P. Jacobson, E. K. Garcia, OCR binarization and image pre-processing for searching historical documents, *Pattern Recognition* 40 (2) (2007) 389–397.