Published in the Proceedings of the *ACM Symposium on Document Engineering (DocEng'23)*, 22-25 August 2023, Limerick, Ireland, which should be cited to refer to this work.

DOI: https://doi.org/10.1145/3573128.3609354

YinYang, a Fast and Robust Adaptive Document Image Binarization for Optical Character Recognition

Jean-Luc Bloechle
CoPeLab Group, Faculty of Science and Medicine, University of Fribourg
Fribourg, Switzerland
jean-luc.bloechle@unifr.ch

Jean Hennebert
iCoSys Institute, University of Applied Science Western Switzerland
Fribourg, Switzerland
jean.hennebert@hefr.ch

Christophe Gisler
iCoSys Institute, University of Applied Science Western Switzerland
Fribourg, Switzerland
christophe.gisler@hefr.ch

ABSTRACT

Optical Character Recognition (OCR) from document photos taken by cell phones is a challenging task. Most OCR methods require prior binarization of the image, which can be difficult to achieve when documents are captured with various mobile devices in unknown lighting conditions. For example, shadows cast by the camera or the camera holder on a hard copy can jeopardize the binarization process and hinder the next OCR step. In the case of highly uneven illumination, binarization methods using global thresholding simply fail, and state-of-the-art adaptive algorithms often deliver unsatisfactory results. In this paper, we present a new binarization algorithm using two complementary local adaptive passes and taking advantage of the color components to improve results over current image binarization methods. The proposed approach gave remarkable results at the DocEng'22 competition on the binarization of photographed documents.

CCS CONCEPTS

• **Computing methodologies → Image processing**.

KEYWORDS

binarization, image thresholding, image processing, OCR

ACM Reference Format:
Jean-Luc Bloechle, Jean Hennebert, and Christophe Gisler. 2023. YinYang, a Fast and Robust Adaptive Document Image Binarization for Optical Character Recognition. In *ACM Symposium on Document Engineering 2023 (DocEng '23), August 22–25, 2023, Limerick, Ireland*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3573128.3609354

1 INTRODUCTION

Image binarization is a low-level task commonly performed as a pre-processing step in document analysis systems. The aim is to convert a grayscale or color image into a monochrome form so that the foreground information is represented by black pixels and the background by white pixels, i.e., a bitonal image. This procedure is frequently used in systems that do not need to analyze color or texture, and instead use shape recognition techniques. Typical examples include robotic applications such as line followers and visual navigation in mazes and corridors, advanced driver assistance systems, autonomous vehicles with lane tracking, and widely-used optical character recognition techniques (for text, numbers, barcodes, QR-codes). Systems with limited memory and processing, such as embedded or mobile devices, can successfully benefit from a fast and robust binary image analysis technique.

This paper is structured as follows. Section 2 describes the so-called YinYang method and the technical optimizations applied in its implementation. Section 3 presents the experimental validations carried out. Section 4 concludes with a brief discussion of the proposed algorithm.

2 PROPOSED BINARIZATION METHOD

The aim of our method is to binarize images of written or printed text documents acquired by scanner or photography. The expected result of the binarization process is a bitonal image with black text over a white background.

The method presented in this paper is based on two fundamental assumptions: first, the foreground (i.e., the text) must be locally darker than the background, and second, there are locally more background pixels than foreground pixels. The locality condition is of critical importance, as images can be acquired in any environment with different devices and may therefore be subject to strong disparities in brightness. Documents where the text has a lighter color than the background (e.g., white text on a dark background) can be resolved by a pre-processing phase that detects the color inversion and, if necessary, produces the negative image.

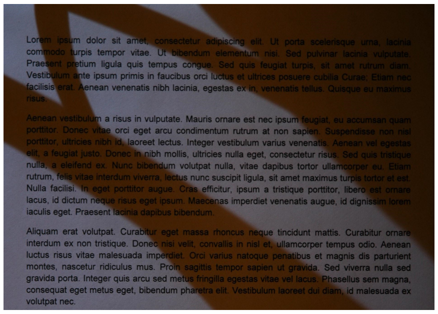
The YinYang algorithm performs two main complementary adaptive passes: (A) background estimation and subtraction, and (B) adaptive Otsu computation and thresholding. Fig. 1 shows the steps involved in the entire YinYang processing workflow. Various approaches to combining binarization methods have been proposed in the literature [12], [4], [1]. These approaches apply several binarization methods to a single image and either select the best result or merge the results together. They therefore do not propose a new binarization algorithm and are computationally expensive. In contrast to these combined approaches, YinYang integrates well-known concepts from the literature as well as original steps to propose an efficient and innovative binarization algorithm.

2.1 Background Estimation and Subtraction

This first phase proceeds in three steps: (1) background estimation, (2) subtraction of the estimated background, and (3) normalization of the resulting foreground. The aim of the background estimation is to generate a new, artificial version of the image in which the

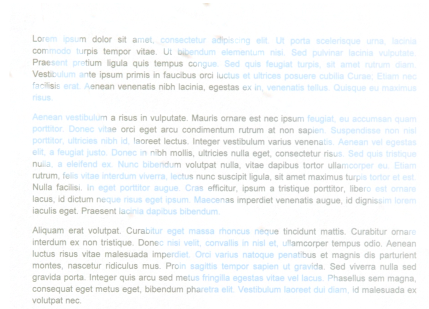
DocEng '23, August 22–25, 2023, Limerick, Ireland

ACM ISBN 979-8-4007-0027-9/23/08.
https://doi.org/10.1145/3573128.3609354

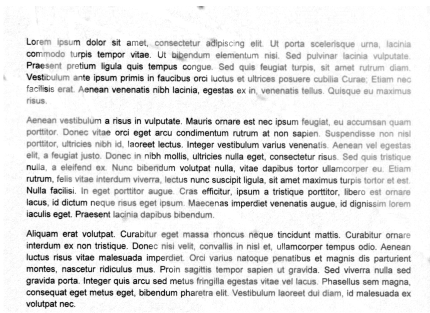


(a) Original image

(b) Background Estimation

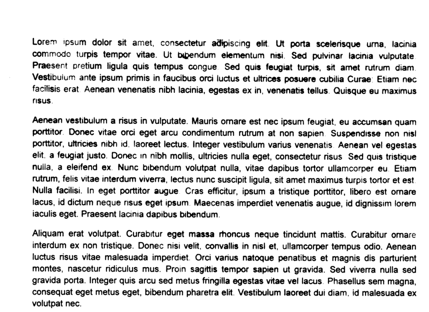


(c) Foreground Extraction



(d) Foreground Normalization

(e) Adaptive Otsu Thresholds



(f) Binarized image

Figure 1: YinYang image binarization processing workflow. The detected background (b) and adaptive Otsu thresholds (e) are artificially reconstructed by interpolating the values from an 8x8 pixel grid (hence the background jaggy effect).

textual content has been removed. To achieve this goal, the background color at each pixel $P(x, y)$ is estimated by detecting the most frequent color in a local window surrounding P (Fig. 1b). Given that the background and foreground of an image are mainly distinguished visually by pixel brightness, and for the sake of simplicity and computational efficiency, we propose an original solution. First, we compute the most frequent luminosity from the local window's luminosity histogram, using a gray level version of the image. Next we retrieve the color pixels corresponding to the most frequent gray level luminosity in the local window and average them to obtain the estimated background color of P.

The estimated background image is then subtracted from the original image, resulting in a first approximation of the foreground image (i.e., the textual part of the image). This foreground image is still encoded in the RGB color space because color components do carry very valuable information for effective binarization (Fig. 1c). The RGB foreground image is then converted to a gray level image by keeping the minimum color component value (R, G or B) for each pixel. This procedure increases the contrast of the foreground image, since only the darkest color components are used as gray intensities. The resulting gray level image is normalized by linearly stretching the contrast to its full range, i.e., from black to white, or 0 to 255 (Fig. 1d).

2.2 Adaptive Otsu and Thresholding

The second phase proceeds in three steps: (1) adaptive Otsu computation, (2) foreground upsampling, and (3) foreground thresholding.

Otsu's method [11] applied to image binarization returns a single intensity threshold that separates pixels into two classes: foreground and background. This threshold is determined by searching the intensity value that minimizes the intra-class variance. While AdOtsu [10] combines background estimation with an adaptive Otsu, our adaptive Otsu thresholds are computed from the normalized foreground image (Fig. 1e). Applying local Otsu's binarization thresholds on the estimated foreground significantly improves the binarization results, especially for images with large disparities in brightness.

Converting a 256 grayscale image into a bi-level image involves a significant loss of information. To mitigate data loss and further improve binarization results, the foreground image is first upsampled, then thresholded with the Otsu's adaptive thresholds. The use of bicubic interpolation [5], instead of a bilinear, slightly enhances the quality of upsampling and therefore the binarization results, particularly when processing low-resolution images.

2.3 Resolution Lowering for Image Background and Adaptive Otsu Computation

Extensively computing the background color for each pixel in the image would be highly inefficient. The same applies to adaptive Otsu thresholds. To speed up the process, the resolution of both the computed background image and the adaptive Otsu thresholds is reduced. The actual implementation of the algorithm computes only one pixel out of 64, using an 8x8 pixels grid. To counter the lower resolution effects, the non-computed background and Otsu missing values are generated on demand using a bilinear interpolation. Using a pixel grid, i.e., reducing the resolution, saves processing time and memory space. The choice of an 8 pixels grid proved to be a good balance between binarization quality and computation time in three binarization competitions [8], [9] and [6].

2.4 Local Windows Subsampling

Local windows dimensions of 64x64 pixels have also empirically demonstrated to be a good balance between binarization quality and computation time during binarization competitions. Each value on the 8x8 pixels grid (either a background value or an Otsu threshold) is computed from its local window containing 4096 pixels (64x64 pixels). Accessing and using all these 4096 pixels for each position on the pixels grid would be very greedy in computing time. As a solution, a subsampling mask is used to further reduce the number of analyzed pixels inside local windows to 156 out of 4096, which is less than 4%. The subsampling mask is predefined and has been designed to uniformly disperse the pixels on a representative non-aligned grid. Again, it has demonstrated to be an adequate balance between binarization quality and computation time.

2.5 Algorithm Parallelization

Image processing algorithms can greatly benefit from the relatively recent advent of multi-processors and multi-cores PC architectures. To take full advantage of these multithreaded architectures, YinYang was parallelized. The processing time for 188 images from the training dataset of the DocEng'22 binarization competition [6] was considerably reduced, from a mean of 5.68 seconds to 1.42 seconds per image, representing a fourfold time saving.

3 EXPERIMENTAL VALIDATION

Our experimental validation is based on independent results obtained during the "Quality, Space & Time Competition on Binarizing Photographed Documents" presented at the ACM Symposium on Document Engineering DocEng 2022 [6].

To be exact, YinYang has been evaluated and improved over the years in three binarization competitions: DocEng'2020, ICDAR'2021, and DocEng'2022. In DocEng'2022, YinYang competed against 68 binarization algorithms, some classical (Otsu, IsoData, Niblack, Bernsen, Sauvola, ISauvola, Bradley, ...) and others recent (iNICK, CNW, DocDLinkNet, DeepOtsu, DocUNet, Michalak21, DilatedUNet, DE-GAN, DPLinkNet, ...). Full evaluation and comparison of YinYang, as well as competing algorithms, is available in several recent competition papers such as "A Quality, Size and Time Assessment of the Binarization of Documents Photographed by Smartphones" by Bernardino et Al. (2023) [2].

3.1 Validation Corpus and Approach

To validate the approaches proposed at the DocEng'22 binarization competition and compare them with other popular and/or state of the art image binarization methods, Rafael Dueire Lins et al. [6] created several test sets of unevenly lit images. Specifically, their test sets included nine documents obtained from six different models of widely used cell phones (Samsung N10+, Samsung S21, Motorola G9 Plus, Samsung GalaxyA10S, Samsung Galaxy S20 and Apple Iphone SE 2), using their strobe flash in both on/off modes. As indicated in their paper [6], Rafael Dueire Lins et al. used quality, time, and space as evaluation criteria. With regard to quality, they state: "Two quality measures were used to evaluate the performance of the binarization algorithms. The first one, *Perr*, compares the proportion between the black-to-white pixels in the scanned and photographed binary documents. The second one made use of Google Vision to perform Optical Character Recognition (OCR) on the documents and applies the Levenshtein distance (*Ldist*) to the correct number of characters in the document transcription (*#char*). The error rate is calculated as: $[Ldist] = (\#char - Ldist)/\#char$. Measures were ranked in the same way as in [8]. First, the ranking for each measure is calculated for each document in a class. Then, the summation of the rank order for all documents in the class defines the final ranking. Visual inspection was applied to check the consistency of the results obtained."

3.2 Selection of an Evaluation Measure

The *Perr* measure used by Rafael Dueire Lins et al. is relatively subjective, as it is not a direct measure of binarization quality. Such a measure can be misleading and give acceptable results even with poor binarization, provided that the ratio between black and white pixels is respected. Given these considerations, and the fact that our algorithm is primarily designed to improve the OCR phase, we have chosen to focus our evaluation on the OCR results of the binarizing competition.

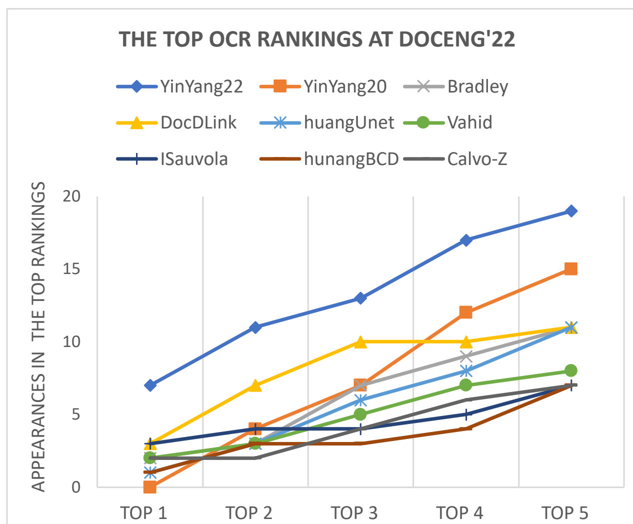


Figure 2: YinYang22 topped competing algorithms for OCR pre-processing at the DocEng'22 binarization competition

3.3 Best Ranked Algorithm for OCR

The evaluation was carried out on a dataset divided into 36 classes, according to cell phone models, flash on/off and document types. Fig. 2 shows that version 2022 of the YinYang algorithm outperformed all other binarization algorithms for OCR pre-processing. Fig. 2 is actually a comprehensive synthesis of the DocEng'22 competition evaluation (as far as OCR performance is concerned), all relevant evaluation information and details have already been exhaustively presented in the competition papers [2, 6].

Out of the 36 document class rankings, YinYang22 was ranked as the best algorithm 7 times. It appeared 11 times in the top 2, 13 times in the top 3, 17 times in the top 4 and 19 times in the top 5. The second most ranked algorithm in the top 5 was YinYang20, an earlier version of our algorithm, which appears 15 times. Next, three algorithms ranked 11 times in the top 5: DocDLink [13], Bradley [3], and huangUnet [7]. YinYang22 was consistently the most ranked algorithm in the tops 1 to 5 out of 69 binarization algorithms.

3.4 Best Ranking Score Algorithm for OCR

To quantify the performance of each binarization algorithm more precisely, we defined an overall OCR ranking score. For each document class, the top 10 algorithms received scores ranging from 10 to 1, i.e., 10 for the best algorithm, 9 for the second best, 8 for the third best, and so on. The score calculated for an algorithm is the sum of the scores for each document class, normalized by the number of document classes and multiplied by 10, to give a final score between 0 and 100. The Table 1 shows that YinYang22 is the best binarization algorithm for OCR with a score of 50, and Bradley [3], the second best with a score of 38. Interestingly, Bradley is a simple, very fast binarization algorithm that has aged well. The third best scoring algorithm is YinYang20, and the fourth is DocDLink.

Table 1: The top ten algorithm OCR ranking scores and their mean processing times in seconds (P-Time)

Rank	Algorithm	Score	P-Time
1	YinYang22	50	1.42
2	Bradley	38	0.33
3	YinYang20	36	1.58
4	DocDLink	30	7.65
5	Vahid	27	23.73
6	huangUnet	27	66.05
7	CNW	25	3.49
8	ISauvola	21	0.41
9	Akbari-3	19	32.28
10	huangBCD	19	282.66

3.5 Mean Image Processing Time

To be usable in production mode, image binarization algorithms need to be fast enough. Waiting more than five seconds to binarize an image captured from a cell phone is not acceptable to an end-user. The quality of the image binarization algorithm cannot be achieved at any cost. The mean image processing time in seconds (P-Time) is shown in Table 1. The difference between the fastest and slowest algorithms is quite significant: 0.33 second for Bradley versus 282.66 seconds for huangBCD. The second-best algorithm, Bradley, is the fastest of the top ten, with a mean processing time of 0.33 second per image. The multithreaded version of YinYang22 stands up well, with an estimated mean processing time of 1.42 seconds per image. Logically, the significant gain in quality between YinYang22 and Bradley comes at the price of slower processing times. However, YinYang22 remains efficient and ready to be used in production mode, unlike most of the other algorithms in the top 10, which are too slow to be responsive in real-life applications.

4 CONCLUSION

This paper presented YinYang, a fast and robust adaptive document image binarization algorithm for OCR. Submitted at the DocEng'22 binarization competition, YinYang appeared more frequently in the top rankings than other competing algorithms. The competition tests were carried out on various and heterogenous datasets, underlining YinYang's ability to be generic and robust under a wide range of conditions. The strength and novelty of the proposed method comes from the use of two complementary local adaptive passes: first, background estimation and subtraction, then adaptive Otsu thresholding. The first pass leverages on the image's color information, while the second uses a bicubic upsampling to transfer as much grayscale information as possible to the final bi-level image.

In addition to providing good binarization quality, the evaluation also showed that YinYang is quite efficient in terms of processing time and is therefore a good candidate for limited resources or near real-time applications. This processing performance was achieved thanks to substantial optimization strategies described in the paper: use of pixel grids, sampling masks and parallel processing.

REFERENCES

[1] Euthimios Badekas and Nikos Papamarkos. 2007. Optimal combination of document binarization techniques using a self-organizing map neural network. *Engineering Applications of Artificial Intelligence* 20, 1 (2007), 11–24.

[2] Rodrigo Bernardino, Rafael Dueire Lins, and Ricardo da Silva Barboza. 2023. A Quality, Size and Time Assessment of the Binarization of Documents Photographed by Smartphones. *Journal of Imaging* 9, 2 (2023), 41.

[3] Derek Bradley and Gerhard Roth. 2007. Adaptive thresholding using the integral image. *Journal of graphics tools* 12, 2 (2007), 13–21.

[4] David Hebert, Stephane Nicolas, and Thierry Paquet. 2013. Discrete CRF based combination framework for document image binarization. In *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 1165–1169.

[5] Robert Keys. 1981. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing* 29, 6 (1981), 1153–1160.

[6] Rafael Dueire Lins, Rodrigo Barros Bernardino, Ricardo Barboza, and Raimundo Oliveira. 2022. The Winner Takes It All: Choosing the "best" Binarization Algorithm for Photographed Documents. In *International Workshop on Document Analysis Systems*. Springer, 48–64.

[7] Rafael Dueire Lins, Rodrigo Barros Bernardino, Elisa Barney Smith, and Ergina Kavallieratou. 2021. ICDAR 2021 competition on time-quality document image binarization. In *International Conference on Document Analysis and Recognition*. Springer, 708–722.

[8] Rafael Dueire Lins, Ergina Kavallieratou, Elisa Barney Smith, Rodrigo Barros Bernardino, and Darlisson Marinho de Jesus. 2019. ICDAR 2019 time-quality binarization competition. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 1539–1546.

[9] Rafael Dueire Lins, Steven J Simske, and Rodrigo Barros Bernardino. 2021. Binarisation of photographed documents image quality and processing time assessment. In *Proceedings of the 21st ACM Symposium on Document Engineering*. 1–6.

[10] Reza Farrahi Moghaddam and Mohamed Cheriet. 2012. AdOtsu: An adaptive and parameterless generalization of Otsu's method for document image binarization. *Pattern Recognition* 45, 6 (2012), 2419–2431.

[11] Nobuyuki Otsu. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 1 (1979), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

[12] Bolan Su, Shijian Lu, and Chew Lim Tan. 2011. Combination of document image binarization techniques. In *2011 International Conference on Document Analysis and Recognition*. IEEE, 22–26.

[13] Lichen Zhou and Chuang Zhang. 2018. Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction. 2018 IEEE. In *CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 192–1924.