



Article

Using Paper Texture for Choosing a Suitable Algorithm for Scanned Document Image Binarization

Rafael Dueire Lins^{1,2,3,*} , Rodrigo Bernardino^{1,3} , Ricardo da Silva Barboza³
and Raimundo Correa De Oliveira³

¹ Centro de Informática, Universidade Federal de Pernambuco, Recife 50670-901, PE, Brazil

² Departamento de Computação, Universidade Federal Rural de Pernambuco, Recife 55815-060, PE, Brazil

³ Coordenação de Engenharia da Computação, Escola Superior de Tecnologia, Universidade do Estado do Amazonas, Manaus 69410-000, AM, Brazil

* Correspondence: rdl@cin.ufpe.br; Tel.: +55-81-98896-0698

Abstract: The intrinsic features of documents, such as paper color, texture, aging, translucency, the kind of printing, typing or handwriting, etc., are important with regard to how to process and enhance their image. Image binarization is the process of producing a monochromatic image having its color version as input. It is a key step in the document processing pipeline. The recent Quality-Time Binarization Competitions for documents have shown that no binarization algorithm is good for any kind of document image. This paper uses a sample of the texture of the scanned historical documents as the main document feature to select which of the 63 widely used algorithms, using five different versions of the input images, totaling 315 document image-binarization schemes, provides a reasonable quality-time trade-off.

Keywords: document binarization; historical documents; DIB dataset; scanned documents; binarization competitions; binarization algorithms



Citation: Lins, R.D.; Bernardino, R.; Barboza, R.d.S.; De Oliveira, R.C.

Using Paper Texture for Choosing a Suitable Algorithm for Scanned Document Image Binarization. *J. Imaging* **2022**, *8*, 272. <https://doi.org/10.3390/jimaging8100272>

Academic Editors: Angelo Marcelli, Francesco Fontanella and Ioannis Pratikakis

Received: 12 July 2022

Accepted: 30 August 2022

Published: 5 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The process of converting a color image into its black-and-white (or monochromatic) version is called binarization or thresholding. The binary version of document images are, in general, more readable by humans, and save storage space [1,2] and communication bandwidth in networks, as the size of binary images is often orders of magnitudes smaller than the original gray or color images; they also use less toner for printing. Thresholding is a key preprocessing step for document transcription via OCR, which allows document classification and indexing.

No single binarization algorithm is good for all kinds of document images, as is demonstrated by the recent Quality-Time Binarization Competitions [3–7]. The quality of the resulting image depends on a wide variety of factors, from the digitalization device and its setup to the intrinsic features of the document, from the paper color and texture to the way the document was handwritten or printed. The time elapsed in binarization also depends on the document features and varies widely between algorithms. A fundamental question arises here: if the document features are deterministic for the quality output of the binary image and there is also a large time-performance variation, and there is a growing number of binarization algorithms, how does one choose an algorithm that provides the best quality-time trade-off? Most users tend to binarize a document image with one of the classical algorithms, such as Otsu [8] or Sauvola [9]. Often, the quality of the result is not satisfactory, forcing the user to enhance the image through filtering (salt-and-pepper, etc.) or to hand-correct the image.

The case of the binarization of photographed documents is even more complex than scanned ones, as the document image has uneven resolution and illumination. The ACM DocEng Quality-Time Binarization Competitions for Photographed Documents [5–7] have

shown that in addition to the physical document characteristics, the camera features and its setup (whether the in-built strobe flash is on or off) influence which binarization algorithm performs the best in quality and time performance. The recent paper [10] presents a new methodology to choose the “best” binarization algorithm in quality and time performance for documents photographed with portable digital cameras embedded in cell phones. It assesses 61 binarization algorithms to point out which binarization algorithm quality-time performs the best for OCR preprocessing or image visualization/printing/network transmission for each of the tested devices and setup. It also chooses the “overall winner”, and the binarization algorithms that would be the “first-choice” in the case of a general embedded application, for instance.

The binarization of scanned documents is also a challenging task. The quality of the resulting image varies not only with the set resolution of the scanner (today, the “standard” is either 200 or 300 dpi), but it also depends heavily on the features of each document, such as paper color and texture, how the document was handwritten or printed, the existence of physical noises [11], etc. Thus, it is important to have some criteria to point out which binarization algorithm, among the best algorithms today, provides the best quality-time trade-off for scanned documents.

Traditionally, binarization algorithms convert the color image into gray-scale before performing binarization. Reference [12] shows that the performance of binarization algorithms may differ if the algorithm is fed with the color image, its gray-scale converted image or one of its R, G, or B channels. Several authors [13,14] show that texture analysis plays an important role in document image processing. Two of the authors of this paper showed that the analysis of paper texture allows one to determine the age of documents for forensic purposes [15], avoiding document forgeries. This paper shows that by extracting a sample of the paper (background) texture of a scanned document, one can have a good indication of one of the 315 binarization schemes tested [12] that provides a suitable quality monochromatic image, with a reasonable processing time to be integrated into a document processing pipeline.

2. Materials and Methods

This work made use of the International Association for Pattern Recognition (IAPR) document image binarization (DIB) platform (<https://dib.cin.ufpe.br>, last accessed on 24 August 2022), which focuses on document binarization. It encompasses several datasets of document images of historical, bureaucratic, and ordinary documents, which were handwritten, machine-typed, offset, laser- and ink-jet printed, and both scanned and photographed; several documents had corresponding ground-truth images. In addition to being a document repository, the DIB platform encompasses a synthetic document image generator, which allows the user to create over 5.5 million documents with different features. As already mentioned, Ref. [12] shows that binarization algorithms, in general, yield different quality images whenever fed with the color, gray-scale-converted, and R, G, and B channels. Here, 63 classical and recently published binarization algorithms are fed with the five versions of the input image, totaling 315 different binarization schemes. The full list of the algorithms used is presented in Tables 1 and 2, along with a short description and the approach followed in each of them.

Table 1. Tested binarization algorithms—Part 1.

Method	Category	Description
Akbari_1 [16]	Deep Learning	Segnet network architecture fed by multichannel images (wavelet sub bands)
Akbari_2 [16]	Deep Learning	Variation of Akibari_1 with multiple networks
Akbari_3 [16]	Deep Learning	Variation of Akibari_1 where fewer channels are used
Bataineh [17]	Local threshold	based on local and global statistics

Table 1. *Cont.*

Method	Category	Description
Bernsen [18]	Local threshold	Uses local image contrast to choose threshold
Bradley [19]	Local threshold	Adaptive thresholding using the integral image of the input
Calvo-Zaragoza [20]	Deep learning	Fully convolutional Encoder–decoder FCN with residual blocks
CLD [21]	Local threshold	Contrast enhancement followed by adaptive thresholding and artifact removal
CNW [22]	Local threshold	Combination of Niblack and Wolf’s algorithm
dSLR [23]	Global threshold	Uses Shannon entropy to find a global threshold
DeepOtsu (SL) [24]	Deep Learning	Neural networks learn degradations and global Otsu generates binarization map
DE-GAN [25]	Deep Learning	Uses a conditional generative adversarial network
DiegoPavan (DP) [4]	Deep Learning	Downscale image to feed a DE-GAN network
DilatedUNet [5]	Deep Learning	Downsample to smooth image and use a dilated convolutional layer to correct the feature map spatial resolution
DocDLinkNet [26]	Deep Learning	D-LinkNet architecture with document image patches
DocUNet (WX) [3]	Deep Learning	A hybrid pyramid U-Net convolutional network is fed with morphological bottom-hat transform enhanced document images
ElisaTV [27]	Local threshold	Background estimation and subtraction
Ergina-Global [28]	Global threshold	Average color value and histogram equalization
Ergina-Local [29]	Local threshold	Detects where to apply local thresholding after a applying a global one
Gattal [30]	Clustering	Automatic Parameter Tuning of K-Means Algorithm
Gosh [31]	Clustering	Clustering applied to a superset of foreground estimated by Niblack’s algorithm
Howe [32]	CRF Laplacian	unary term and pairwise Canny-based term
Huang [33]	Global threshold	Minimizes the measures of fuzzines
HuangBCD (AH ₁) [4]	Deep Learning	BCD-Unet based model binarize and combine image patches
HuangUNet (AH ₂) [4]	Deep Learning	Unet based model binarize and combine image patches
iNICK [34]	Local threshold	Adaptively sets k in NICK based on global std. dev.
Intermodes [35]	Global threshold	Smooth histogram until only two local maxima
ISauvola [36]	Local threshold	Uses image contrast in combination with Sauvola’s binarization
IsoData [37]	Global threshold	IsoData clustering algorithm applied to image histogram
Jia-Shi [38]	Edge based	Detecting symmetry of stroke edges
Johannsen-Bille [39]	Global threshold	Minimizes formula based on the image entropy
Kapur-SW [40]	Global threshold	Maximizes formula based on the image entropy
Li-Tam [41]	Global threshold	Minimum cross entropy
Lu-Su [42]	Edge based	Local thresholding near edges after background removal
Mean [43]	Global threshold	Mean of the grayscale levels
Mello-Lins [44]	Global threshold	Uses Shannon Entropy to determine the global threshold. Possibly the first to properly handle back-to-front interference

Table 2. Tested binarization algorithms—Part 2.

Method	Category	Description
Michalak [45]	Image Processing	Downsample image to remove low-frequency information and apply Otsu
MO ₁ [45]	Image Processing	Downsample image to remove low-frequency information and apply Otsu
MO ₂ [46]	Image Processing	Equalize illumination and contrast, apply morphological dilatation and Bradley’s method
MO ₃ [47]	Local threshold	Average brightness corrected by two parameters to apply local threshold
MinError [48]	Global threshold	Minimum error threshold
Minimum [35]	Global threshold	Variation of Intermodes algorithm
Moments [49]	Global threshold	Aims to preserve the moment of the input picture
Niblack [50]	Local threshold	Based on window mean and std. dev.
Nick [51]	Local threshold	Adapts Niblack based on global mean
Otsu [8]	Global threshold	Maximize between-cluster variance of pixel intensity
Percentile [52]	Global threshold	Based on partial sums of the histogram levels
Pun [53]	Global threshold	Defines an anisotropy coefficient related to the asymmetry of the histogram
RenyEntropy [54]	Global threshold	Uses Renyi’s entropy similarly as Kapur’s method

Table 2. Cont.

Method	Category	Description
Sauvola [9]	Local threshold	Improvement on Niblack
Shanbhag [55]	Global threshold	Improves Kapur’s method; view the two pixel classes as fuzzy sets
Singh [56]	Global threshold	Uses integral sum image prior to local mean calculation
Su-Lu [57]	Edge based	Canny edges using local contrast
Triangle [58]	Global threshold	Based on most and least frequent gray level
Vahid (RNB) [4]	Deep Learning	A pixel-wise segmentation model based on Resnet50-Unet
WAN [59]	Global threshold	Improves Sauvola’s method by shifting up the threshold
Wolf [60]	Local threshold	Improvement on Sauvola with global normalization
Wu-Lu [61]	Global threshold	Minimizes the difference between the entropy of the object and the background
Yen [62]	Global threshold	Multilevel threshold based on maximum correlation criterion
YinYang [5]	Image Processing	Detect background with median of small overlapping windows, extract it and apply Otsu
YinYang21 (JB) [5]	Image Processing	A faster and more effective version of YinYang algorithm
Yuleny [3]	Shallow ML	A XGBoost classifier is trained with features generated from Otsu, Niblack, Sauvola, Su and Howe algorithms

Ref. [63] presents a machine learning approach for choosing among five binarization algorithms to binarize parts of a document image. Another interesting approach to enhance document image binarization is proposed in [64] and consists of analyzing the features of the original document to compose the result of the binarization of several algorithms to generate a better monochromatic image. Such a scheme was tested with 25 binarization algorithms, and it performed more than 3% better than the first rank in the H-DIBCO 2012 contest in terms of F-measure. The time-processing cost of such a scheme is prohibitive if one considers processing document batches, however.

One of the aims raised by the researchers in the DIB platform team is to develop an “image matcher” in such a way that given a real-world document, it looks for the synthetic document that better matches its features, as sketched in Figure 1. For each of the 5.5 million synthetic documents in the DIB platform, one would have the algorithms that would yield the best quality-time performance for document readability or OCR transcription. Thus, the match of the “real-world” document and the synthetic one would point out which binarization algorithm would yield the “best” quality-time performance for the real-world document. It is fundamental that the Image Matcher is a very lightweight process not to overload the binarization processing time. If one or a small set of document features provide enough information to make such a good choice, it is more likely that it will be for the image-matcher to be fast enough to be part of a document processing pipeline.



Figure 1. DIB image matcher.

In this paper, the image texture is taken as a key for selecting the real-world image that more closely resembles another real-world document for which one has a ground-truth monochromatic image of reference. Such images were carefully chosen from the set of historical documents in the DIB platform such as to match a large number of historical documents of interest from the late 19th century to today. To extract a sample of the texture, one manually selects a window of 120 × 60 pixels from the document to be binarized, as

shown in Figure 2. Only one window from each image was cropped in such a way that there was no presence of text from the front or any back-to-front interference. A vector of features is built, taking into account each RGB channel of the sample, the image average filtered $(R + G + B)/3$, and its gray-scale equivalent. Seven statistical measures are taken and placed in a vector: mean, standard deviation, mode, minimum value, maximum value, median, and kurtosis. This results in a vector containing 28 features, which describes the overall color and texture characteristics.

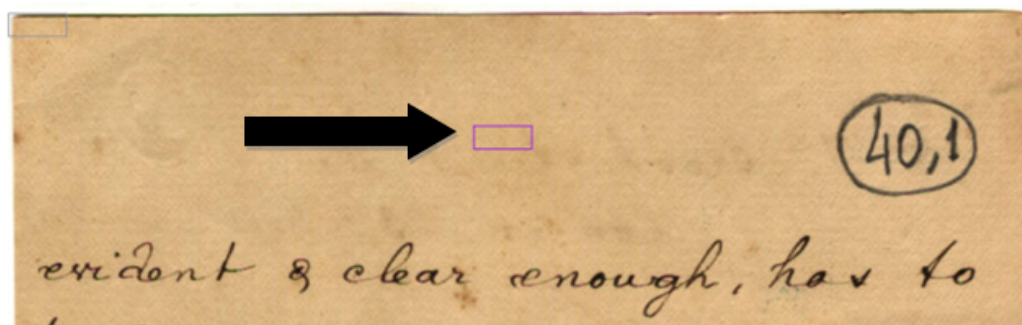


Figure 2. DIB—Choosing a texture pattern.

In this study, 40 real-world images are used, and the Euclidean distance between the texture vectors is used to find the 20 pairs of most similar documents. The texture with the smallest distance is chosen, and its source document image is used to determine the best binarization algorithm. Figure 3 illustrates how such a process is applied to a sample image and the chosen texture.

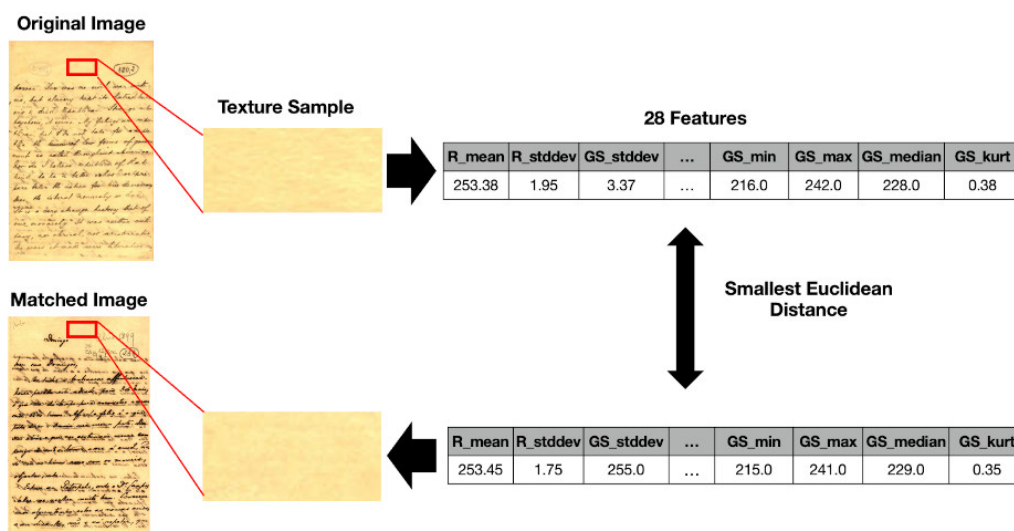


Figure 3. Example of matching real-world images by texture.

3. Binarization Algorithm Selection Based on the Paper Texture

In a real-world document, one expects to find three overlapping color distributions. This includes one that corresponds to the plain paper background, which becomes the paper texture, which should yield white pixels in the monochromatic image. The second distribution tends to be a much narrower Gaussian that corresponds to the printing or writing, which is mapped onto black pixels in the binary image. The third distribution, the back-to-front interference [11,65] overlaps the other two distributions, bringing one of the most important causes of binarization errors. Figure 4 presents a sample image with the corresponding color distributions.

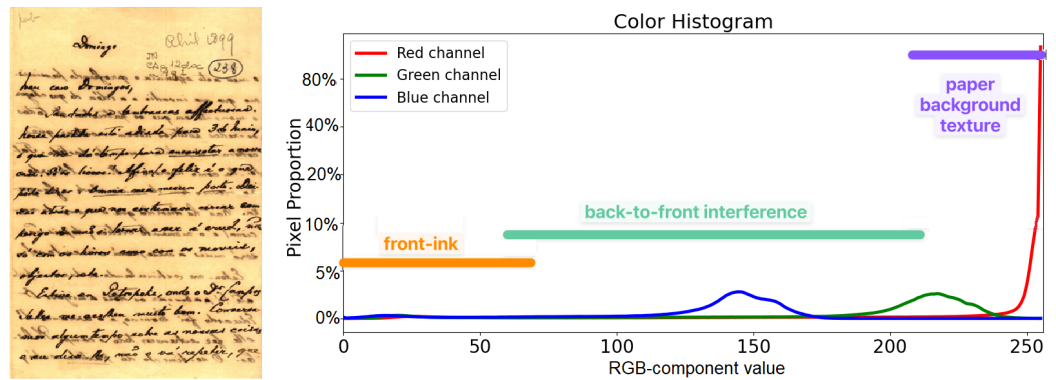


Figure 4. Pixel color distributions in a document image with strong back-to-front interference.

Deciding which binarization algorithm to use in a document tends to be a “wild guess”, a user-experience-based guess, or an a posteriori decision, which means one uses several binarization algorithms and chooses the image that “looks best” as a result. Binarization time is seldom considered. One must agree that the larger the number of binarization algorithms one has, the harder it is to guess the ones that will perform well for a given document. Ideally, the Image Matcher under development in the DIB-platform would estimate all the image parameters (texture type, kind of writing or printing, the color of ink, intensity of the back-to-front interference, etc.) to pinpoint which of the over 5.5 million synthetic images best matches the features of the “real world” document to be binarized. If that synthetic image is known, one would know which of the 315 binarization schemes assessed here would offer the best quality-time balance for that synthetic image.

This paper assumes that by comparing the paper texture between two real-world documents, one of which knows which binarization algorithm presents the best quality-time trade-off, one can use that algorithm on the other document, yielding acceptable quality results. Cohen’s Kappa [66,67] (denoted by k) is used here as a quality measure:

$$k = \frac{P_O - P_C}{1 - P_C}, \tag{1}$$

which compares the observed accuracy with an expected accuracy, assessing the classifier performance. P_O is the number of correctly mapped pixels (accuracy) and P_C is

$$P_C = \frac{n_{bf} \times n_{gf} + n_{bb} \times n_{gb}}{N^2}, \tag{2}$$

where n_{bf} and n_{bb} are the number of pixels mapped as foreground and background on the binary image, respectively, and n_{gf} and n_{gb} are the number of foreground and background pixels on the GT image, and N is the total number of pixels. The ranking for the pixels is defined by sorting the measured kappa in ascending order.

The peak signal-noise ratio (PSNR), distance reciprocal distortion (DRD) and F-Measure (FM) have been used for a long time to assess binarization results [68,69], becoming the chosen measures for nearly all studies in this area. Thus, they are also provided, even though the ranking process only takes Cohen’s Kappa into account. The PSNR for a $M \times N$ image is defined as the peak signal power to average noise power, which, for 8-bit images, is

$$PSNR = 10 \log_{10} \frac{255^2 \cdot MN}{\sum_i \sum_j (x(i,j) - y(i,j))^2}. \tag{3}$$

The DRD [70] correlates the human visual perception with the quality of the generated binary image. It is computed by

$$DRD = \frac{1}{NUBN(GT)} \sum_{k=1}^S DRD_{ij} |B(i, j) - GT(i, j)| \quad (4)$$

$$DRD_{ij} = \sum_{x=-2}^2 \sum_{y=-2}^2 W_{xy} |B(i+x, j+y) - G(i+x, j+y)|, \quad (5)$$

where $NUBN(GT)$ is the number of non-uniform 8×8 binary blocks in the ground-truth (GT) image, S is the flipped pixels and DRD_{ij} is the distortion of the pixel at position (i, j) in relation to the binary image (B), which is calculated by using a 5×5 normalized weight matrix W_{xy} as defined in [70]. DRD_{ij} equals to the weighted sum of the pixels in the 5×5 block of the GT that differ from the centered k th flipped pixel at (x, y) in the binarization result image B . The smaller the DRD, the better.

The F-Measure is computed as

$$FM = \frac{2 \times Recall \times Precision}{Recall + Precision}, \quad (6)$$

where $Recall = \frac{TP}{TP+FN}$, $Precision = \frac{TP}{TP+FP}$ and TP , FP , FN denote the true positive, false positive and false negative values, respectively.

Once the matching image (the most similar) is found, the best quality-time algorithm is used to binarize the original image. Algorithms with the same kappa are in the same ranking position. Several algorithms have a similar processing time. Among the top-10 in terms of quality, the fastest is chosen as the best quality-time binarization algorithm. This paper conjectures that considering two documents that were similarly printed (handwritten, offset printed, etc.) and have similar textures, if the best quality-time algorithm is known for one image, that same algorithm could be applied to the other image, yielding high-quality results. No doubt that if a larger number of document features besides the document texture, such as the strength of the back-to-front interference, the ink color and kind of pen, the printing method, etc. were used, the chances of selecting the best quality binarization scheme would be larger, but could imply in a prohibitive time overhead. It is also important to stress that the number of documents with back-to-front interference is small in most document files, and the ones with strong interference is even smaller. In the case of the bequest [71] of Joaquim Nabuco (1849/1910, Brazilian statesman and writer and the first Brazilian ambassador to the U.S.A.), for instance, the number of letters is approximately 6500, totaling about 22,000 pages. Only 180 documents were written on both sides in translucent paper, of which less than 10% of them exhibit strong back-to-front interference. Even in those documents, the paper texture plays an important role in the parameters of the binarization algorithms. Thus, in this paper, one assumes that the paper texture is the key information for choosing a suitable binarization scheme that has a large probability of being part of an automatic document processing pipeline. Evidence that such a hypothesis is valid is shown in the next section.

4. Results

In order to evaluate the automatic algorithm selection based on the texture, 26 handwritten and 14 typewritten documents were carefully selected from the DIB platform such that they are representative of a large number of real-world historical documents. Such documents belong to the Nabuco bequest [71] and were scanned in 200 dpi. Table 3 presents the full size of each document used in this study. All of them have a ground-truth binary image. The Euclidean distance between the feature vector of their paper textures was used to find the pairs of most similar documents. Five versions of the original and matched image were used in the final ranking.

Table 3. Size of the test images in pixels.

Image	Size	Image	Size	Image	Size	Image	Size
HW01	888 × 1361	HW11	907 × 1383	HW21	1077 × 1345	TW05	1602 × 2035
HW02	915 × 1358	HW12	937 × 1372	HW22	894 × 1387	TW06	1551 × 1947
HW03	920 × 1374	HW13	924 × 1381	HW23	925 × 1376	TW07	1212 × 1692
HW04	911 × 1426	HW14	895 × 1373	HW24	992 × 1552	TW07	1212 × 1692
HW05	1021 × 1586	HW15	999 × 1557	HW25	912 × 1375	TW09	1619 × 1961
HW06	1024 × 1550	HW16	890 × 1380	HW26	891 × 1381	TW10	1599 × 2067
HW07	898 × 1389	HW17	954 × 1401	TW01	1645 × 2140	TW11	1701 × 1957
HW08	1016 × 1570	HW18	1049 × 1670	TW02	1660 × 2186	TW12	1677 × 2179
HW09	866 × 1354	HW19	917 × 1372	TW03	1581 × 2119	TW13	1692 × 2193
HW10	1021 × 1579	HW20	1050 × 1326	TW04	1575 × 1989	TW14	1671 × 2165

The results and the images are described in Tables 4–12. The letter that follows the algorithm name indicates the version of the input document image used, that as shown in [12] yields monochromatic images of different quality with different processing times:

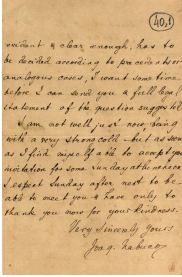
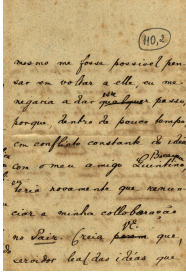
- **C:** all RGB channels (color)
- **R:** the red channel
- **G:** the green channel
- **B:** the blue channel
- **L:** luminance image, calculated as $0.299 * R + 0.587 * G + 0.114 * B$

The other parts stand for:

1. **Original Image:** the image one wants to binarize.
2. **Matched Image:** the image which one already has the algorithm that yields the best quality-time trade-off amongst all the 315 binarization schemes.
3. **Textures samples:** sample of the paper background of the original image (left) used to select the texture matched image (right), whose sample is presented below each document
4. **Results Table:** the best 10 algorithms for the original image
5. **Direct Binarization:** the best quality-time algorithm and corresponding binary image according to the ranking of all 315 binarization schemes. The choice is made by directly looking at the results of all algorithms.
6. **Texture-based Binarization:** the best quality-time algorithm of the matched image and the corresponding monochromatic version of the original image binarized with the chosen algorithm.

The algorithm choice was appropriate for all the presented images, as can be noted by visually inspecting the binary images, their quality ranking, and the kappa, PSNR, DRD, and F-Measure values. For Tables 4, 6, 7, and 9, the selected algorithm was at rank 5 or more and did not yield a significantly worse image in those cases. The difference in kappa was smaller than 10%, except in the case of the image shown in Table 8, in which the kappa reached 12%. It is interesting to observe that for the image shown in Table 8, an image with strong back-to-front interference, although the value of kappa has the highest percent difference of all the tested images, the monochromatic image produced by using the texture binarization scheme proposed here is visually more pleasant and readable for humans than the scheme that yields the best kappa, as may be observed in the zoomed image shown in Figure 5. One may see that the texture-based choice of the binarization scheme leaves some noise in areas that correspond to the back-to-front interference, most of which could be removed with a salt-and-pepper filter. As previously remarked here, images with strong back-to-front interference tend to be rare in any historical document file.

Table 4. Results for image matching with image HW 01.

		Binarization Results for the Original Image					Original Image HW 01	Matched Image HW 12
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	IsoData-C	0.92	20.07	1.50	92.11	0.01		
1	IsoData-L	0.92	20.06	1.50	92.06	0.01		
1	Otsu-C	0.92	20.10	1.48	92.15	0.00		
1	Otsu-L	0.92	20.06	1.50	92.06	0.00		
1	Gattal-C	0.92	20.12	1.46	92.13	45.59		
1	Gattal-L	0.92	20.06	1.50	92.06	45.87		
2	dSLR-C	0.91	19.67	1.69	91.54	0.02		
2	dSLR-G	0.91	19.82	1.62	91.69	0.02		
...		
2	MO ₁ -R	0.91	19.81	1.55	91.58	0.14		



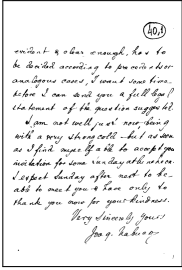
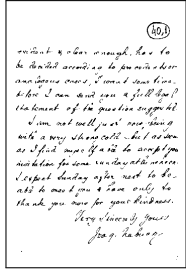
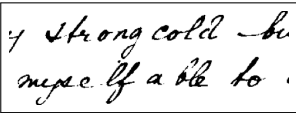
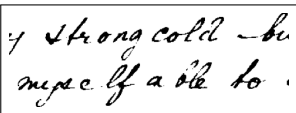
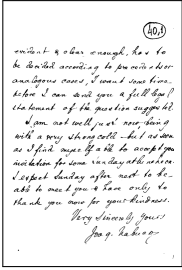
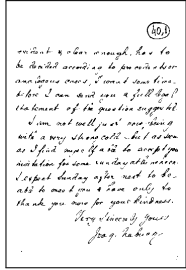
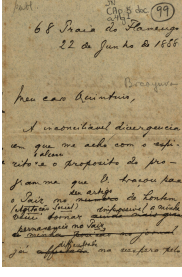
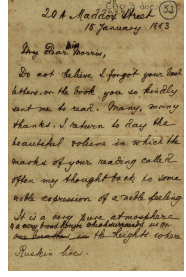
Original Texture	Matched Texture	Direct Binarization	Texture-based
			
<p>Direct Binarization</p>  <p>Texture-based binarization</p> 		<p>Otsu-C</p> 	<p>MO₁-R</p> 

Table 5. Results for image matching with image HW 02.

		Binarization Results for the Original Image					Original Image HW 02	Matched Image HW 16
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	Li-Tam-C	1.00	34.52	0.11	99.73	0.01		
2	dSLR-G	0.99	28.01	0.31	98.80	0.01		
2	dSLR-L	0.99	28.39	0.29	98.90	0.01		
2	Intermodes-G	0.99	29.07	0.27	99.07	0.01		
2	Intermodes-L	0.99	27.78	0.34	98.76	0.01		
2	Li-Tam-G	0.99	29.07	0.27	99.07	0.01		
2	Li-Tam-L	0.99	29.77	0.24	99.21	0.01		
3	dSLR-R	0.98	26.90	0.38	98.45	0.01		
...		
8	dSLR-C	0.93	20.56	1.73	93.77	0.01		

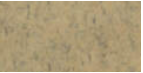

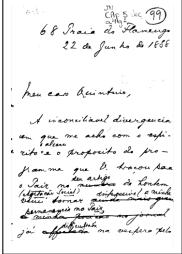
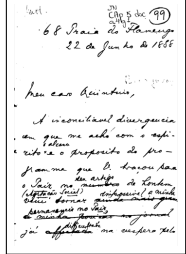
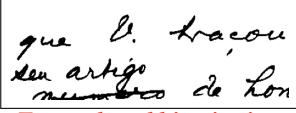
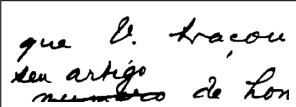
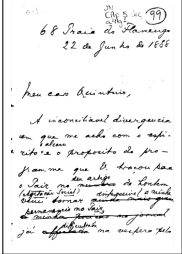
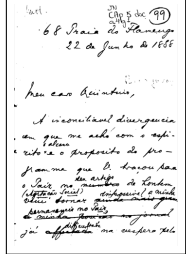
Original Texture	Matched Texture	Direct Binarization	Texture-based
			
<p>Direct Binarization</p>  <p>Texture-based binarization</p> 		<p>Li-Tam-C</p> 	<p>dSLR-C</p> 

Table 6. Results for image matching with image HW 03.

		Binarization Results for the Original Image					Original Image HW 03	Matched Image HW 12
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	ElisaTV-G	0.96	22.52	0.91	95.83	1.60		
1	ElisaTV-L	0.96	22.52	0.91	95.85	1.59		
1	MO ₁ -C	0.96	23.24	0.82	96.51	0.01		
1	MO ₁ -G	0.96	22.91	0.92	96.24	0.01		
1	MO ₁ -L	0.96	23.15	0.85	96.43	0.01		
1	MO ₁ -R	0.96	22.87	0.86	96.19	0.01		
2	dSLR-G	0.95	22.00	1.07	95.21	0.01		
2	dSLR-L	0.95	22.10	1.03	95.31	0.01		
2	dSLR-R	0.95	22.00	1.02	95.30	0.01		
2	Huang-R	0.95	21.96	1.03	95.29	0.01		

Original Texture	Matched Texture	Direct Binarization	Texture-based
<p>Direct Binarization</p>		<p>Texture-based binarization</p>	

Table 7. Results for image matching with image HW 04.

		Binarization Results for the Original Image					Original Image HW 04	Matched Image HW 09
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	dSLR-C	0.96	22.26	4.15	96.70	0.01		
1	Intermodes-C	0.96	22.08	4.28	96.53	0.01		
1	Intermodes-G	0.96	22.01	4.36	96.48	0.01		
1	Intermodes-L	0.96	22.15	4.22	96.60	0.01		
1	Intermodes-R	0.96	21.62	4.68	96.17	0.01		
1	Li-Tam-L	0.96	21.68	4.64	96.17	0.01		
1	Sauvola-C	0.96	21.69	4.68	96.13	0.03		
1	Sauvola-G	0.96	21.87	4.52	96.30	0.03		
...		
8	Howe-C	0.89	17.33	13.41	90.49	6.83		

Original Texture	Matched Texture	Direct Binarization	Texture-based
<p>Direct Binarization</p>		<p>Texture-based binarization</p>	

Table 8. Results for image matching with image HW 05.

Binarization Results for the Original Image							Original Image HW 05	Matched Image HW 06
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	Jia-Shi-L	0.92	18.73	25.58	93.23	4.61		
1	Jia-Shi-R	0.92	18.57	25.51	92.91	4.61		
2	DocDLink-C	0.91	18.03	27.56	91.85	4.08		
3	Jia-Shi-B	0.90	17.53	32.94	91.23	4.50		
4	DocDLink-L	0.89	17.18	35.56	90.31	4.01		
5	DocDLink-B	0.88	16.58	41.27	88.96	3.98		
6	Lu-Su-B	0.86	15.85	48.36	87.59	14.78		
6	Lu-Su-C	0.86	15.71	51.69	87.25	14.14		
...		
11	Wolf-B	0.81	14.83	62.79	83.15	0.05		

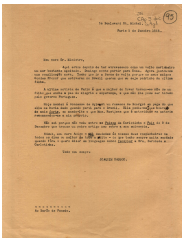
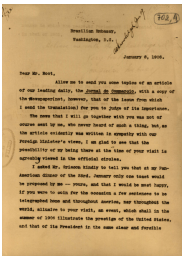
Original Texture	Matched Texture	Direct Binarization	Texture-based
		Jia-Shi-L	Wolf-B
		Wolf-B	Jia-Shi-L

Table 9. Results for image matching with image HW 06.

Binarization Results for the Original Image							Original Image HW 06	Matched Image HW 05
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	Wolf-B	0.93	20.74	8.98	93.32	0.05		
2	dSLR-B	0.92	20.24	10.66	92.72	0.01		
2	dSLR-G	0.92	20.01	11.42	92.44	0.01		
2	dSLR-L	0.92	19.92	11.15	92.10	0.01		
2	Intermodes-B	0.92	20.10	11.55	92.68	0.01		
2	Intermodes-C	0.92	19.78	12.28	92.09	0.01		
2	Intermodes-L	0.92	19.79	12.30	92.13	0.01		
2	Li-Tam-B	0.92	20.24	10.66	92.72	0.01		
...		
4	Jia-Shi-L	0.90	18.73	15.06	90.60	4.43		

Original Texture	Matched Texture	Direct Binarization	Texture-based
		Wolf-B	Jia-Shi-L
		Jia-Shi-L	Wolf-B

Table 10. Results for image matching with image TW 01.

Binarization Results for the Original Image							Original Image TW 01	Matched Image TW 07
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	Li-Tam-C	0.94	23.84	4.20	94.18	0.02		
1	Li-Tam-G	0.94	23.94	4.16	94.30	0.02		
1	Li-Tam-L	0.94	23.53	4.56	93.81	0.01		
1	MO ₁ -C	0.94	24.00	4.11	94.40	0.02		
1	MO ₁ -G	0.94	24.09	4.09	94.50	0.02		
1	MO ₁ -L	0.94	23.98	4.15	94.38	0.02		
2	Intermodes-B	0.93	23.29	5.30	93.35	0.02		
2	IsoData-B	0.93	23.29	5.30	93.35	0.02		
...		
6	Su-Lu-L	0.89	21.69	6.83	89.26	0.59		

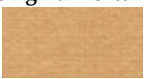



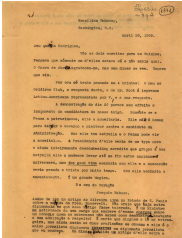
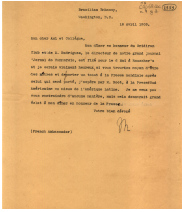
Original Texture	Matched Texture	Direct Binarization	Texture-based
			
<p>Direct Binarization</p> <p>ei á Condessa de Byleadt u dado quando parti para o Br eu mando-lhe o que Mme. R a mim próprio.</p>		<p>Li-Tam-L</p> <p>Su-Lu-L</p>	
<p>Texture-based binarization</p> <p>ei á Condessa de Byleadt u dado quando parti para o Br eu mando-lhe o que Mme. R a mim próprio.</p>			

Table 11. Results for image matching with image TW 02.

Binarization Results for the Original Image							Original Image TW 02	Matched Image TW 11
#	Algorithm	Kappa	PSNR	DRD	FM	Time		
1	dSLR-C	0.96	23.20	3.95	96.28	0.02		
1	dSLR-G	0.96	23.07	4.06	96.09	0.02		
1	dSLR-L	0.96	23.18	3.89	96.19	0.02		
1	Intermodes-C	0.96	22.78	4.39	95.95	0.02		
1	Intermodes-G	0.96	22.77	4.46	95.93	0.02		
1	Intermodes-L	0.96	22.72	4.47	95.90	0.02		
1	Li-Tam-G	0.96	22.77	4.46	95.93	0.02		
1	Li-Tam-L	0.96	22.72	4.47	95.90	0.02		
...		
5	Otsu-G	0.92	19.78	9.04	92.31	0.01		



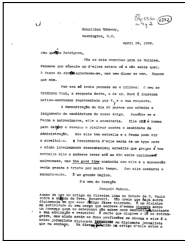
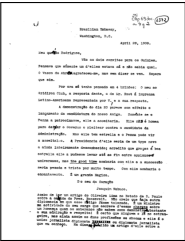
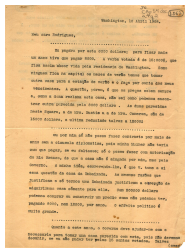
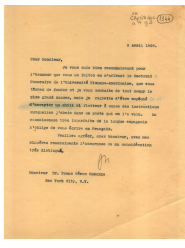

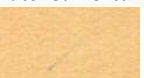
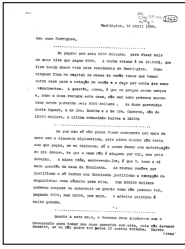
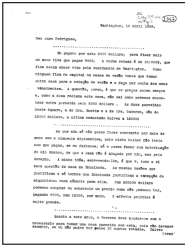
Original Texture	Matched Texture	Direct Binarization	Texture-based
			
<p>Direct Binarization</p> <p>A demonstração do dia 20 parece da candidatura do nosso Amigo. trocinasse, elle a acceitaria.</p>		<p>dSLR-C</p> <p>Otsu-G</p>	
<p>Texture-based binarization</p> <p>A demonstração do dia 20 parece da candidatura do nosso Amigo. trocinasse, elle a acceitaria.</p>			

Table 12. Results for image matching with image TW 03.

#	Algorithm	Binarization Results for the Original Image					Original Image TW 03	Matched Image TW 10
		Kappa	PSNR	DRD	FM	Time		
1	Minimum-C	0.97	24.80	4.82	96.99	0.02		
1	Nick-C	0.97	25.07	5.01	97.14	0.08		
1	Nick-G	0.97	24.62	5.53	96.81	0.07		
1	Nick-L	0.97	25.03	5.07	97.11	0.07		
1	Singh-C	0.97	25.33	4.87	97.33	0.12		
1	Singh-G	0.97	24.58	5.65	96.81	0.11		
1	Singh-L	0.97	25.06	5.13	97.16	0.12		
2	MinError-C	0.96	24.45	4.90	96.67	0.02		
2	MinError-L	0.96	24.19	5.14	96.43	0.02		
2	Nick-R	0.96	23.44	6.77	95.86	0.08		

Original Texture	Matched Texture	Direct Binarization	Texture-based
			
<p>Direct Binarization</p> <p>dona reclama esta ce parecida pelo 8000</p> <p>Texture-based binarization</p> <p>dona reclama esta ce parecida pelo 8000</p>		<p>Minimum-C</p> <p>Minimum-C</p>	

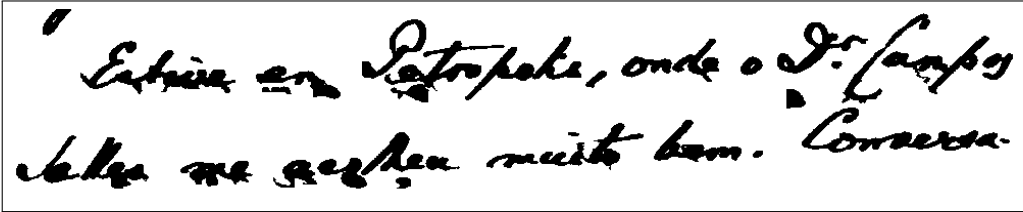
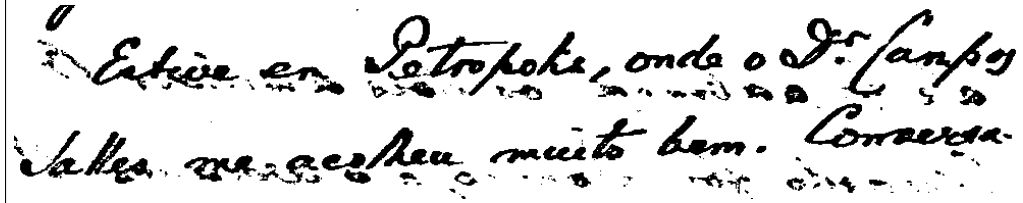
Direct Binarization	Texture-based Binarization
	

Figure 5. Zoom in a part of a document image HW 05 with strong back-to-front interference binarized using the direct (Jia-Shi [38]) and texture-based (Wolf [72]) methods.

In the case of the document image HW 04, presented in Table 7, although the difference in kappa is 7.3%, visually inspecting the resulting binary image, it is really close in quality to the actual best in terms of quality, which implies the choice based on texture does indicate a good option of binarization algorithm even with a relatively lower rank, although the Howe algorithm [32] used to binarize the matched image HW 09, has a much higher processing time than the da Silva–Lins–Rocha algorithm [2] (dSLR-C), the top quality algorithm using direct binarization.

It is also relevant to say that there is a small degree of subjectivity in the whole process as the ground-truth images of historic documents are hand-processed. If one looks at Table 10, one may also find some differences in the produced images that illustrate such subjectivity. The result of the direct binarization using Li–Tam algorithm [41] yields an

image with a high kappa of 0.94 with much thicker strokes than the one chosen by the texture-based method, the Su–Lu algorithm [57], both of which were fed with the gray-scale image obtained by using the conventional luminance equation. Although the kappa of the Su–Lu binarized image is 0.89, the resulting image is as readable as the Li–Tam one, a phenomenon which is somehow similar to the one presented in Figure 5. The main idea of the proposed methodology is not to find exactly the same best quality-time algorithm as directly binarizing, but one algorithm that yields satisfactory results.

5. Conclusions

Document binarization is a key step in many document processing pipelines; thus it is important to be performed quickly and with high quality. Depending on the intrinsic features of the scanned document image, the quality-time performance of the binarization algorithms known today varies widely. The search for a document feature that is possible to be extracted automatically with a low time complexity that may provide an indication of which binarization algorithm provides the best quality-time trade-off is thus of strategic importance. This paper takes the document texture as such a feature.

The results presented have shown that the document texture information may be satisfactorily used as a way to choose which binarization algorithm to apply to scanned historical documents, and how the input image should be if the original color image, its gray-scale conversion or one of its RGB channels is to be successfully scanned. The choice of the algorithms is based on the use of real images that “resemble” the paper background of the document to be binarized. A sample of the texture of the document is collected and compared with the remaining 39 different paper textures used for handwritten or machine typed documents, each of which points to an algorithm that provides the best quality-time trade-off for the synthetic document. The use of that algorithm in the real-world document to be binarized was assessed here and yielded results that may be considered of good quality and quickly produced, both for image readability by humans or automatic OCR transcription.

This paper presents evidences that by matching the textures of scanned documents, one can find suitable binarization algorithms for a given new image. The methodology presented may be enhanced further by including new textures and binarization schemes. The inclusion of new textures may narrow the euclidean distance between the image to be binarized document and the existing textures in the dataset. The choice of the most suitable binarization scheme for the document with the new texture may be done by the visual inspection of the result of the top-ranked binarization algorithms of the document image with the closest Euclidean distance of the images already in the reference dataset.

A number of issues remain open for further work, however. The first one is automating the process of texture sampling and matching in such a way as not to be a high overload on the binarization process as a whole. This may also involve the collection of texture samples in different parts of the document to avoid collecting parts either printed with back-to-front interference or other physical noises, such as stains or holes. The second point is trying to minimize the number of features in the vector-of-features to be matched with the vector-of-features of the synthetic textures. The third point is attempting to find a better matching strategy than simply calculating the Euclidean distance between the vectors, as done here, perhaps by using some kind of clustering.

Author Contributions: Conceptualization: R.D.L.; methodology: R.D.L.; data curation: The DIB Team (<https://dib.cin.ufpe.br>, last accessed on 24 August 2022); writing—original draft preparation: R.D.L., R.B., R.d.S.B. and R.C.D.O.; writing—review and editing: R.D.L., R.B., R.d.S.B. and R.C.D.O.; funding acquisition: R.d.S.B., R.C.D.O. and R.D.L. All authors have read and agreed to the published version of the manuscript.

Funding: The research reported in this paper was mainly sponsored by the RD&I project Callidus Academy signed between the Universidade do Estado do Amazonas (UEA) and Callidus Indústria

through the Lei de Informática/SUFRAMA. Rafael Dueire Lins was also partly sponsored by CNPq—Brazil.

Data Availability Statement: The results presented here made use of the IAPR (International Association on Pattern Recognition) DIB—Document Image Binarization dataset, available at: <https://dib.cin.ufpe.br>, last accessed on 24 August 2022.

Acknowledgments: The authors are grateful to the referees of this paper that raised several important points that improved its presentation and to all researchers who made the code for their binarization algorithms available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mello, C.A.; Lins, R.D. Generation of images of historical documents by composition. In Proceedings of the 2002 ACM Symposium on Document Engineering (DocEng'02), McLean, VA, USA, 8–9 November 2002; pp. 127–133. [CrossRef]
2. Da Silva, J.M.M.; Lins, R.D. Color document synthesis as a compression strategy. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil, 23–26 September 2007; pp. 466–470. [CrossRef]
3. Lins, R.D.; Kavallieratou, E.; Barney Smith, E.; Bernardino, R.B.; de Jesus, D.M. ICDAR 2019 time-quality binarization competition. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Sydney, NSW, Australia, 20–25 September 2019; pp. 1539–1546. [CrossRef]
4. Lins, R.D.; Bernardino, R.B.; Barney Smith, E.; Kavallieratou, E. ICDAR 2021 Competition on Time-Quality Document Image Binarization. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Lausanne, Switzerland, 5–10 September 2021; pp. 1539–1546. [CrossRef]
5. Lins, R.D.; Simske, S.J.; Bernardino, R.B. DocEng'2020 time-quality competition on binarizing photographed documents. In Proceedings of the DocEng'20: ACM Symposium on Document Engineering 2020, Online, 29 September–1 October 1 2020. [CrossRef]
6. Lins, R.D.; Bernardino, R.B.; Simske, S.J. DocEng'2021 time-quality competition on binarizing photographed documents. In Proceedings of the ACM Symposium on Document Engineering (DocEng'21), Limerick, Ireland, 24–27 August 2021; pp. 1–4.
7. Lins, R.D.; Bernardino, R.B.; Barboza, R.; Simske, S.J. DocEng'2022 Quality, Space, and Time Competition on Binarizing Photographed Documents. In Proceedings of the DocEng'22. ACM, San Jose, CA, USA, 20–23 September 2022; pp. 1–4.
8. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [CrossRef]
9. Sauvola, J.; Seppanen, T.; Haapakoski, S.; Pietikainen, M. Adaptive document binarization. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Ulm, Germany, 18–20 August 1997; Volume 1, pp. 147–152.
10. Lins, R.D.; Bernardino, R.B.; Barboza, R.; Oliveira, R. The Winner Takes It All: Choosing the “best” Binarization Algorithm for Photographed Documents. In Proceedings of the Document Analysis Systems, La Rochelle, France, 22–25 May 2022; pp. 48–64. [CrossRef]
11. Lins, R.D. A Taxonomy for Noise in Images of Paper Documents—The Physical Noises. In Proceedings of the Lecture Notes in Computer Science, Hanoi, Vietnam, 23–27 November 2009; Volume 5627, pp. 844–854. [CrossRef]
12. Lins, R.D.; Bernardino, R.B.; da Silva Barboza, R.; Lins, Z.D. Direct Binarization a Quality-and-Time Efficient Binarization Strategy. In Proceedings of the 21st ACM Symposium on Document Engineering (DocEng'21), Limerick, Ireland, 24–17 August 2021. [CrossRef]
13. Mehri, M.; Héroux, P.; Gomez-Krämer, P.; Mullot, R. Texture feature benchmarking and evaluation for historical document image analysis. *Int. J. Doc. Anal. Recognit. (IJ DAR)* **2017**, *20*, 1–35. [CrossRef]
14. Beyerer, J.; Leon, F.; Frese, C. Texture analysis. In *Machine Vision*; Springer: Berlin/Heidelberg, Germany, 2016.
15. Barboza, R.d.S.; Lins, R.D.; Jesus, D.M.d. A Color-Based Model to Determine the Age of Documents for Forensic Purposes. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 25–28 August 2013; pp. 1350–1354. [CrossRef]
16. Akbari, Y.; Britto, A.S.; Al-Maadeed, S.; Oliveira, L.S. Binarization of Degraded Document Images using Convolutional Neural Networks based on predicted Two-Channel Images. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Sydney, NSW, Australia, 20–25 September 2019.
17. Bataineh, B.; Abdullah, S.N.H.S.; Omar, K. An adaptive local bin. method for doc. images based on a novel thresh. method and dynamic windows. *Pattern Recog. Lett.* **2011**, *32*, 1805–1813. [CrossRef]
18. Bernsen, J. Dynamic thresholding of gray-level images. In Proceedings of the International Conference on Pattern Recognition, Paris, France, 27–31 October 1986; pp. 1251–1255.
19. Bradley, D.; Roth, G. Adaptive Thresholding using the Integral Image. *J. Graph. Tools* **2007**, *12*, 13–21. [CrossRef]
20. Calvo-Zaragoza, J.; Gallego, A.J. A selectional auto-encoder approach for document image binarization. *Pattern Recognit.* **2019**, *86*, 37–47. [CrossRef]
21. Saddami, K.; Munadi, K.; Away, Y.; Arnia, F. Effective and fast binarization method for combined degradation on ancient documents. *Heliyon* **2019**, *5*, e02613. [CrossRef]

22. Saddami, K.; Afrah, P.; Mutiawani, V.; Arnia, F. A New Adaptive Thresholding Technique for Binarizing Ancient Document. In Proceedings of the INAPR, Jakarta, Indonesia, 7–8 September 2018; pp. 57–61.
23. Silva, J.M.M.; Lins, R.D.; Rocha, V.C. Binarizing and Filtering Historical Documents with Back-to-Front Interference. In Proceedings of the ACM SAC, Dijon, France, 23–27 April 2006; pp. 853–858. [\[CrossRef\]](#)
24. He, S.; Schomaker, L. DeepOtsu: Document Enhancement and Binarization using Iterative Deep Learning. *Pattern Recognit.* **2019**, *91*, 379–390. [\[CrossRef\]](#)
25. Souibgui, M.A.; Kessentini, Y. DE-GAN: A Conditional Generative Adversarial Network for Document Enhancement. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1180–1191. [\[CrossRef\]](#)
26. Zhou, L.; Zhang, C.; Wu, M. D-linknet: Linknet with pretrained encoder and dilated convolution for satellite imagery road extraction. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
27. Barney Smith, E.H.; Likforman-Sulem, L.; Darbon, J. Effect of Pre-processing on Binarization. In Proceedings of the Document Recognition and Retrieval XVII, San Jose, CA, USA, 19–21 January 2010; p. 75340H.
28. Kavallieratou, E. A binarization algorithm specialized on document images and photos. *ICDAR 2005, 2005*, 463–467.
29. Kavallieratou, E.; Stathis, S. Adaptive binarization of historical document images. In Proceedings of the International Conference on Pattern Recognition, Hong Kong, China, 20–24 August 2006; Volume 3, pp. 742–745.
30. Gattal, A.; Abbas, F.; Laouar, M.R. Automatic Parameter Tuning of K-Means Algorithm for Document Binarization. In Proceedings of the 7th International Conference on Software Engineering and New Technologies (ICSSENT), Hammamet, Tunisia, 26–28 December 2018; pp. 1–4.
31. Bera, S.K.; Ghosh, S.; Bhowmik, S.; Sarkar, R.; Nasipuri, M. A non-parametric binarization method based on ensemble of clustering algorithms. *Multimed. Tools Appl.* **2021**, *80*, 7653–7673. [\[CrossRef\]](#)
32. Howe, N.R. Doc. binarization with automatic parameter tuning. *Int. J. Doc. Anal. Recognit.* **2013**, *16*, 247–258. [\[CrossRef\]](#)
33. Huang, L.K.; Wang, M.J.J. Image thresholding by minimizing the measures of fuzziness. *Pattern Recognit.* **1995**, *28*, 41–51. [\[CrossRef\]](#)
34. Saddami, K.; Munadi, K.; Muchallil, S.; Arnia, F. Improved Thresholding Method for Enhancing Jawi Binarization Performance. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1.
35. Prewitt, J.M.S.; Mendelsohn, M.L. The Analysis of Cell Images. *Ann. N. Y. Acad. Sci.* **2006**, *128*, 1035–1053. [\[CrossRef\]](#) [\[PubMed\]](#)
36. Hadjadj, Z.; Meziane, A.; Cherfa, Y.; Cheriet, M.; Setitra, I. *ISauvola: Improved Sauvola's Algorithm for Document Image Binarization*; Springer: Cham, Switzerland, 2004; pp. 737–745.
37. Velasco, F.R. *Thresholding Using the Isodata Clustering Algorithm*; Technical Report; Office of the Secretary of Defense: Washington, DC, USA, 1979.
38. Jia, F.; Shi, C.; He, K.; Wang, C.; Xiao, B. Degraded document image binarization using structural symmetry of strokes. *Pattern Recognit.* **2018**, *74*, 225–240. [\[CrossRef\]](#)
39. Johannsen, G.; Bille, J. A threshold selection method using information measures. In Proceedings of the International Conference on Pattern Recognition, London, UK, 27–29 January 1982; pp. 140–143.
40. Kapur, J.; Sahoo, P.; Wong, A. A new method for gray-level picture thresholding using the entropy of the histogram. *Comput. Vision, Graph. Image Process.* **1985**, *29*, 140. [\[CrossRef\]](#)
41. Li, C.; Tam, P. An iterative algorithm for minimum cross entropy thresholding. *Pattern Recognit. Lett.* **1998**, *19*, 771–776. [\[CrossRef\]](#)
42. Lu, S.; Su, B.; Tan, C.L. Document image binarization using background estimation and stroke edges. *Int. J. Doc. Anal. Recognit.* **2010**, *13*, 303–314. [\[CrossRef\]](#)
43. Glasbey, C. An Analysis of Histogram-Based Thresholding Algorithms. *Graph. Model. Image Process.* **1993**, *55*, 532–537. [\[CrossRef\]](#)
44. Mello, C.A.B.; Lins, R.D. Image segmentation of historical documents. *Visual2000* **2000**, *30*, 88–96.
45. Michalak, H.; Okarma, K. Fast Binarization of Unevenly Illuminated Document Images Based on Background Estimation for Optical Character Recognition Purposes. *J. Univers. Comput. Sci.* **2019**, *25*, 627–646.
46. Michalak, H.; Okarma, K. Improvement of image binarization methods using image preprocessing with local entropy filtering for alphanumeric character recognition purposes. *Entropy* **2019**, *21*, 562. [\[CrossRef\]](#) [\[PubMed\]](#)
47. Michalak, H.; Okarma, K. Adaptive image binarization based on multi-layered stack of regions. In Proceedings of the International Conference on Computer Analysis of Images and Patterns, Salerno, Italy, 3–5 September 2019; pp. 281–293.
48. Kittler, J. Minimum error thresholding. *Pattern. Recog.* **1986**, *19*, 41–47. [\[CrossRef\]](#)
49. Tsai, W.H. Moment-preserving thresholding: A new approach. *Comput. Vision, Graph. Image Process.* **1985**, *29*, 377–393. [\[CrossRef\]](#)
50. Niblack, W. *An Introduction to Digital Image Processing*; Strandberg: København, Denmark, 1985.
51. Khurshid, K.; Siddiqi, I.; Faure, C.; Vincent, N. Comparison of Niblack inspired binarization methods for ancient documents. In Proceedings of the SPIE, Orlando, FL, USA, 14–15 April 2009; p. 72470U.
52. Doyle, W. Operations Useful for Similarity-Invariant Pattern Recognition. *J. ACM* **1962**, *9*, 259–267. [\[CrossRef\]](#)
53. Pun, T. Entropic thresholding, a new approach. *Comput. Graph. Image Process.* **1981**, *16*, 210–239. [\[CrossRef\]](#)
54. Sahoo, P.; Wilkins, C.; Yeager, J. Threshold selection using Renyi's entropy. *Pattern Recognit.* **1997**, *30*, 71–84. [\[CrossRef\]](#)
55. Shanbhag, A.G. Utilization of Information Measure as a Means of Image Thresholding. *CVGIP Graph. Model. Image Process.* **1994**, *56*, 414–419. [\[CrossRef\]](#)

56. Singh, T.R.; Roy, S.; Singh, O.I.; Sinam, T.; Singh, K.M. A New Local Adaptive Thresholding Technique in Binarization. *arXiv* **2012**, arXiv:1201.5227.
57. Bolan, S.; Shijian, L.; Chew Lim, T. Robust Document Image Binarization Technique for Degraded Document Images. *IEEE Trans. Image Process.* **2013**, *22*, 1408–1417. [[CrossRef](#)]
58. Zack, G.W.; Rogers, W.E.; Latt, S.A. Automatic measurement of sister chromatid exchange frequency. *J. Histochem. Cytochem.* **1977**, *25*, 741–753. [[CrossRef](#)]
59. Mustafa, W.A.; Abdul Kader, M.M.M. Binarization of Document Image Using Optimum Threshold Modification. *J. Phys. C Ser.* **2018**, *1019*, 012022. [[CrossRef](#)]
60. Wolf, C.; Doermann, D. Binarization of low quality text using a Markov random field model. In Proceedings of the Object Recognition Supported by User Interaction for Service Robots, Quebec City, QC, Canada, 11–15 August 2002; Volume 3, pp. 160–163.
61. Lu, W.; Songde, M.; Lu, H. An effective entropic thresholding for ultrasonic images. In Proceedings of the 14th International Conference on Pattern Recognition, Brisbane, QLD, Australia, 17–20 August 1998; Volume 2, pp. 1552–1554.
62. Yen, J.C.; Chang, F.J.C.S.; Yen, J.C.; Chang, F.J.; Chang, S. A New Criterion for Automatic Multilevel Thresholding. *IEEE Trans. Image Process.* **1995**, *4*, 370–378. [[PubMed](#)]
63. Chattopadhyay, T.; Reddy, V.R.; Garain, U. Automatic Selection of Binarization Method for Robust OCR. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Washington, DC, USA, 25–28 August 2013; pp. 1170–1174.
64. Reza Farrahi, M.; Fereydoun Farrahi, M.; Mohamed, C. Unsupervised ensemble of experts (EoE) framework for automatic binarization of document images. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Washington, DC, USA, 25–28 August 2013; pp. 703–707.
65. Lins, R.D.; Guimarães Neto, M.; França Neto, L.; Galdino Rosa, L. An environment for processing images of historical documents. *Microprocess. Microprogramm.* **1994**, *40*, 939–942. [[CrossRef](#)]
66. Congalton, R.G. A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sens. Environ.* **1991**, *37*, 35–46. [[CrossRef](#)]
67. Bernardino, R.; Lins, R.D.; Jesus, D.M. A Quality and Time Assessment of Binarization Algorithms. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Sydney, NSW, Australia, 20–25 September 2019; pp. 1444–1450. [[CrossRef](#)]
68. Ntirogiannis, K.; Gatos, B.; Pratikakis, I. Performance Eval. Methodology for Historical Doc. Image Binarization. *IEEE Trans. Image Process.* **2013**, *22*, 595–609. [[CrossRef](#)]
69. Tensmeyer, C.; Martinez, T. Historical document image binarization: A review. *SN Comput. Sci.* **2020**, *1*, 173. [[CrossRef](#)]
70. Lu, H.; Kot, A.; Shi, Y. Distance-reciprocal distortion measure for binary document images. *IEEE Signal Process. Lett.* **2004**, *11*, 228–231. [[CrossRef](#)]
71. Lins, R.D. Nabuco Two Decades of Document Processing in Latin America. *J. Univers. Comput. Sci.* **2011**, *17*, 151–161.
72. Wolf, C.; Jolion, J.M.; Chassaing, F. Text localization, enhancement and binarization in multimedia documents. In Proceedings of the 2002 International Conference on Pattern Recognition, Quebec City, QC, Canada, 11–15 August 2002; Volume 2, pp. 1037–1040. [[CrossRef](#)]