Binarizing Photographed Document Images 2025 Quality, Time and Space Assessment

Gustavo P. Chaves
Instituto de Pesquisas Eldorado,
Manaus, AM, Brazil
gustavo.chaves@eldorado.org.br

Thaylor Vieira
Centro de Informática, UFPE,
Recife, PE, Brazil
tvm2@cin.ufpe.br

Gabriel de F. P e Silva
UFRPE,
Cabo de Sto. Agostinho, PE, Brazil
gabriel.psilva@ufrpe.br

Rafael Dueire Lins
UFRPE/UFPE,
Recife, PE, Brazil
rafael.dueirelins@ufrpe.br/rdl@cin.ufpe.br

Steven J. Simske
Colorado State University,
Fort Collins, CO, USA
Steve.Simske@colostate.edu

ABSTRACT

Image binarization is fundamental for document image processing. The performance of binarization algorithms depends on several factors that range from the quality of the digitalization devices to the intrinsic features of the document itself and the kind and intensity of the noises present in the image. This assessment on binarizing photographed documents evaluated the quality, time, space, and performance of five new algorithms and ninety-eight "classical" algorithms. The test data set is composed of laser and deskjet printed documents, photographed using six widely used mobile devices with the strobe flash *on*, *off*, and in *auto* modes under three different angles and places of capture.

CCS CONCEPTS

• **Computing methodologies** → **Image segmentation**.

KEYWORDS

Document Engineering, Binarization, Photographed Documents, Scanned Documents, Binarization algorithms, Quality evaluation, Space evaluation, Time evaluation

ACM Reference Format:
Gustavo P. Chaves, Thaylor Vieira, Gabriel de F. P e Silva, Rafael Dueire Lins, and Steven J. Simske. 2025. Binarizing Photographed Document Images 2025 Quality, Time and Space Assessment. In *ACM Symposium on Document Engineering 2025 (DocEng '25), September 02–05, 2025, Nottinham, UK*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3704268.3758318

1 INTRODUCTION

Smartphones have significantly changed photography by making it more accessible and ubiquitous. Rapid advances in smartphone camera technology now offer high-resolution sensors, advanced image processing, and various shooting modes, enabling users to capture professional-quality photos. In addition, the ease of sharing photos on social media has made photography a more engaging and social activity. Nearly three-quarters of the top 10 developed countries' populations own a smartphone [1]. The worldwide smartphone market is expected to expand by 3.09 % in 2025 compared to last year. Cameras became the biggest selling point on smartphones, thus manufacturers have been putting a lot of effort into improving their quality. Smartphone manufacturers have added more cameras to their phones to improve photo quality and optical zoom functionality, while keeping the device thin.

Although smartphone cameras were made to take "family" and landscape photos or making videos of such subjects, taking photos of documents became widespread, saving photocopying costs, and allowing the document image to be easily stored and shared using computer networks. Today, even governments accept photographed document images for all sorts of legal services, such as immigration visas, etc. However, photographed document images are, in general, far more difficult to process than scanned ones since the resolution is uneven, there is perspective distortion, and often interfering external light sources and shadowed areas. The only aspect that is much harder to address in scanned documents is whenever back-to-front interference[30] occurs, a phenomenon often present in historical documents, which in general is not present in photographed ones.

Digital cameras acquire color images, in which each pixel is represented by three bytes to store each of the Red, Green, and Blue components. The conversion of a color image into its black-and-white version is called *binarization*, or *thresholding*. Binarization is a key part in many image processing applications. The performance of the binarization schemes of photographed documents is completely different from that of scanned ones [14, 33, 39]. No binarization algorithm performs well for all kinds of documents [34]. Their performance depends on a wide number of factors from the digitalization device and set-up, image resolution, the kind of physical noises in the document [29], the way the document was printed, typed or handwritten, the age of the document, etc.

This is the sixth venue [32, 35–37, 41] of the ACM DocEng competition focusing on the binarization of smartphone camera-acquired text documents, the type of document that is most often photographed. This paper presents the results of the assessment in quality, size, and time performance of five new algorithms and ninety-eight existing algorithms, including "classical" ones. The

DocEng'25, September 02–05, 2025, Nottinham, UK

ACM ISBN 979-8-4007-1169-5/24/08.
https://doi.org/10.1145/3704268.3758318

[1] https://explodingtopics.com/blog/smartphone-stats, last visited on 22nd July 2025

test set used encompasses laser and deskjet printed text documents photographed in three different places, with six different models of smartphones widely used today, with their in-built strobe flash *on* *off*, and in *auto* modes.

2 PARTICIPANTS

Five new algorithms were included in this assessment. The algorithms and are listed in enrollment order with the affiliation is of their first member:

(A) Scanbot SDK, Germany *(Stefan Dragnev)*: This algorithm is an extension of the Nick binarization scheme. The main feature that distinguishes this algorithm from existing ones is in the lighting normalization step. Algorithms like Sauvola or Nick have no lighting normalization step, which causes lighting irregularities like strong shadows to pollute the statistics inside the filter window. Subtracting low-frequency noise from the image before binarization produces improved results, especially in the presence of strong shadows. Using a median filter, instead of a Gaussian filter further improves results in the presence of strong shadows. Only the executable code that takes an input directory in and produces another directory of binary images was submitted to this competition.

(B) A New Hybrid Binarization Algorithm for Manuscripts, Egypt, *(Ashraf Nijim, Muhammad AboKresha, Ayman El Shenawy , and Reda Abo Alez)*: The paper proposes a new hybrid binarization algorithm, designed to process ancient manuscripts and historical documents with back-to-front interference (also known as bleed-through). The algorithm is divided into three main stages:

Classification: The grayscale image is classified into three classes: foreground pixels (text, black), background pixels (white), and confused pixels (uncertain areas). A threshold is calculated based on the global mean and standard deviation to define these classes.

Segmentation: The confused pixels class is analyzed using vertical and horizontal histograms, detecting local minima to define segmentation lines. Thus, the image is divided into rectangles (sub-images), each containing regions with similar characteristics. The sub-images are determined automatically by the segmentation process, which uses the vertical and horizontal histograms of the confused pixels. An additional algorithm merges adjacent rectangles that have similar concentrations of confused pixels, forming larger sub-images when needed.

Voting: For each sub-image, seven different global thresholding techniques are applied (such as Otsu, Fuzzy C-Means, Moment Preserving, Entropy, MET, Range-Constrained Otsu, among others). From the seven calculated thresholds, the two closest values are chosen, and the final threshold for the sub-image is the average of these two values. Each sub-image is then binarized separately using that threshold. The final result is composed by combining all the binarized sub-images.

(C) Degraded Historical Documents Images Binarization Using a Combination of Enhanced Techniques, Algeria *(Omar Boudraa, Walid Khaled Hidouci, and Dominique Michelucci)*: The paper proposes a multi-phase hybrid system for binarizing degraded historical document images, combining preprocessing, hybrid binarization, and post-processing to enhance the quality of the extracted text.

Preprocessing: Converts color documents to grayscale. Applies CLAHE (Contrast Limited Adaptive Histogram Equalization), which enhances local contrast without excessively amplifying noise. CLAHE is only applied if the contrast is considered weak (evaluated using Michelson's local contrast).

Hybrid Binarization: Combines three well-known binarization techniques: Otsu's Method (global), Multilevel Otsu (TSMO), Nick's Method (adaptive) The choice of method depends on the local contrast level, divided into four categories: Low contrast: TSMO. Medium contrast: Otsu. High contrast: The lowest TSMO threshold. Ambiguous: Additional checks decide between TSMO and Otsu. Moreover, large clusters of suspicious pixels are processed through connected components analysis, and Nick's Method is applied locally to these areas.

Post-processing: Final refinement of the binarized text: Filling single-pixel gaps. Removing small isolated components and noise. Correcting convexity/concavity defects on text edges.

(D) Three Stage Binarization, Taiwan *(RuiYang Ju)*: This algoritm available at GitHub (https://github.com/RuiyangJu) is a hybrid approach for document image binarization, designed especially for degraded or historical documents. It combines:

- CNN-based classification of image patches
- Global thresholding (Otsu's method)
- Local thresholding (Niblack's method)

The main idea is to adaptively choose whether a region should be binarized globally or locally, based on a Convolutional Neural Network (CNN) that classifies each patch. The CNN uses a pre-trained ImageNet model as a feature extractor (e.g., VGG or ResNet backbone). The pipeline steps are:

Preprocessing: The input document image is resized and split into patches. Each patch is normalized and processed for feature extraction.

CNN Patch Classification: The patches are fed through a CNN classifier. The CNN predicts if the patch: Should be binarized globally (simple background, uniform lighting) or Needs local thresholding (complex background, shadows, back-to-front interference).

Binarization: Otsu's global thresholding is applied to patches predicted as "simple". Niblack's local thresholding is applied to patches predicted as "complex". The binarized patches are reassembled into the final output image. Using a CNN pre-trained on ImageNet provides strong feature extraction even with limited labeled data for binarization tasks. The model learns visual cues like shadows, ink bleed, or non-uniform illumination that indicate where

local thresholding is more effective. This algorithm was re-trained with photographed document images available at the IAPR-DIB website (https://dib.cin.ufpe.br/#!/)

(E) Sliedes, Switzerland *(Sami Liedes)*: This is a binarization algorithm that uses neural networks to handle challenging documents, such as those with shadows, stains, or uneven lighting. It was trained to focus on what really matters: the textual content. By using the pre-trained model available in the repository, there is no need for tuning or additional training, simply run the script and get clean, effective binarizations. It claims to be a practical and modern solution for extracting text from documents with quality. The algorithm is available at https://github.com/sliedes/binarize (was last updated in 2020).

The 98 other binarization algorithms used in this assessment are: ACM [35], AugraphyD [38], AugraphyF [38], Akbari_1 [1], Akbari_2 [1], Akbari_3 [1], Bataineh [3], Bernsen [6], Bhowmik [7], Bradley [11], Calvo-Zaragoza [12], dSLR [63], DeepOtsu (SL) [18], DE-GAN [66], DiegoPavan (DP) [65], DilatedUNet [41], DocBin1 [38], DocBin2 [38], DocBin3 [38], DocDLinkNet [76], DocUNet (WX) [39], DocEnTR [38], ElisaTV [2], Ergina-Global [24], Ergina-Local [25], FourBi [38], Gattal [15], Gosh [4], HandwritteNet [39], Howe [19], Huang (AH_1) [20], Huali [39] iNICK [58], Intermodes [53], ISauvola [17], IsoData [71], Jia-Shi [21], Johannsen-Bille [22], Kapur-SW [23], Li-Tam [28], LS-HDIB [38], Lu-Su [43], Mean [16], Mello-Lins [46], Michalak [48], MO1 (Michalak-Okarma) [47] MO2 (Michalak-Okarma) [48] MO3 (Michalak-Okarma) [49] MinError [27], Minimum [53], Moments [70], Niblack [51], Nick [26], Otsu [52], Percentile [13], Pun [54], Robin [2], Sauvola [61], Shanbhag [62], Singh [64], Su-Lu [10], Texture Based [5], T&M [69], Triangle [75], Vahid (RNB) [33], WAN [50], Wolf [72], Wu-Lu [44]. Yasin [39] Yen [74], YinYang [41], YinYang21 (JB) [41]. YinYang22 [32], YinYang23a [8], YinYang23b [8], YinYang23c [8], Yuleny [39] and ZigZag [9] (which has 20 different variations).

3 THE TEST SETS

The test set encompasses nine documents obtained from six different models of portable cell-phones widely used today, whose specifications are presented in Table 1[3]. The documents in this set were clustered according to having the in-built strobe-flash set as *on*, *off* or *auto* modes.

The physical documents used here are the same used in the 2023 competition [37], samples of which are presented in Figures 1 and 2, formed by laser and deskjet printed documents. Two test sets were used here. using the same set of documents throughout. Several digitization noises [29] may appear whenever the document photo was taken under unfavorable conditions or by an incautious user. The document images were cropped leaving a "region of interest" that encompasses little more than the page of the original physical document.

A ground truth binary image was generated by performing a bitwise AND operation between the binarizations produced by the two best-performing algorithms for each image, taking into account the ranking from [36]. The result of such operation, applied to the 1,000 images in the dataset, had to undergo visual inspection to ensure it met high image-quality expectations. This process allowed for the correction of artifacts present in one of the binarizations, ensuring a more reliable ground truth.

Table 1: Summary of device camera specifications

	Samsung S23 Ultra	**Samsung S24 Ultra**
Megapixels	20Mp+10Mp+ 12Mp+10Mp	200Mp+50Mp+ 12Mp+10Mp
Aperture	F1.7+ F4.9+F2.2+F2.4	F1.7+F3.4+F2.2+F2.4
Resolution	12000x9000 pixel	16330x12247 pixel
Sensor size	1/1.3"+1/3.52" +1/3.52"+1/2.55"	1/1.3"+1/2.55"" +1/3.52"
Release year	2023	2024
	iPhone 14	**iPhone 15 Pro Max**
Megapixels	12 Mp+12 Mp	48 Mp+12 Mp+12 Mp
Aperture	F 1.5+F 2.4	F 1.78+F 2.8+F 2.2
Resolution	4000x3000 pixel	8000x6000 pixel
Sensor size	-	1/1.28"+1/2.55"
Release year	2022	2023
	iPhone 16 Pro	**Motorola Edge 50 Pro**
Megapixels	48Mp+12Mp+48Mp	50Mp+13Mp+10Mp
Aperture	F1.78+F2.8+F2.2	F1.47+F2.2+F2
Resolution	8000x6000 pixel	8165x6124 pixel
Sensor size	1/1.28"+1/3.06"	1/1.3"
Release year	2024	2024

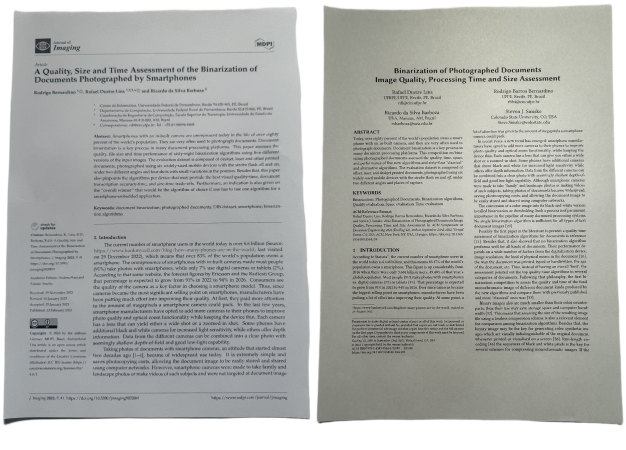

Figure 1: Samples of photographed images of inkjet printed documents taken with an iPhone 16 under artificial lighting, with flash auto(left) and off(right).

4 QUALITY, TIME & SIZE ASSESSMENT

The quality of the final monochromatic image is the most important assessment criterion. Once one has the top-quality images, one may

[2]https://github.com/masyagin1998/robin

[3]Some of the devices have much higher resolution than shown in this Table, which specifies the resolution used for document image acquisition in the used test sets.

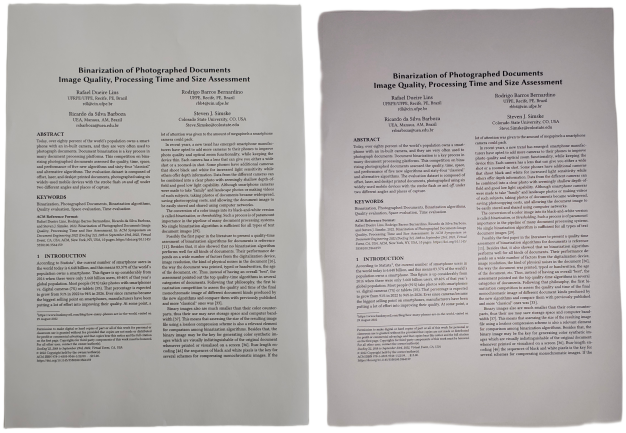

Figure 2: Samples of photographed images of laser printed documents taken with a Motorola Edge 50 under poor lighting, with flash off(left) and on(right).

consider the mean size of the monochromatic file and the mean time elapsed by each of the evaluated algorithms.

The quality measure used here to evaluate the performance of the binarization algorithms was the *PNSR (Peak-signal-to-noise ratio)* , a standard image quality measure.

The PSNR measures were ranked in the same way as in [39]. 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. For larger data sets, an excellent functional test would be document classification accuracy.

The mean size of the monochromatic image files in Tiff (G4) format, which is possibly the most efficient lossless compression scheme for binary images [40], is also assessed here.

The processing time evaluation provides the order of magnitude of the time elapsed for binarizing the whole datasets. The training-times for the AI-based algorithms were not computed. Four processing devices were used:

- **Device 1 (CPU)**: Intel Core i5-12500H CPU @ 2.50GHz, with 16GB RAM and GPU GeForce GTX 1650 4GB
- **Device 2 (CPU)**: Intel Core i9-9900K CPU @ 3.60GHz, with 64GB RAM and GPU NVIDIA GeForce RTX 2080 Ti 12GB
- **Device 3 (CPU)**: AMD Ryzen Threadripper PRO 5975WX 32-Cores, with 128 GB RAM and GPU NVIDIA GeForce RTX 3080 Ti 12 GB
- **Device 4(CPU)**: Intel(R) Xeon(R) CPU 4 vCPU cores @ 2.20GHz com 32GB de RAM, available at Kaggle platform

The algorithms were executed in the following platforms:

- **Device 1, Windows 11 (version 22H2):**
  - **Matlab:** Akbari_1 [1], Akbari_2 [1], [4].Akbari_3 [1], Bhowmik [7], ElisaTV [2], Ergina-Global [24], Ergina-Local [25], Gosh [4], Howe [19], Ghosh [4], Jia-Shi [21], Gattal [15], Lu-Su [43], Michalak [48], Yasin [39], iNICK [58], CNW [56], CLG [57], MO1 (Michalak-Okarma) [47], MO2 (Michalak-Okarma) [48], MO3 (Michalak-Okarma) [49].
- **Device 2, Linux Pop!_OS 22.04**
  - **Python Machine Learning with CPU:** DE-GAN [66], Calvo-Zaragoza [12], DiegoPavan [65], Vahid [33], DeepOtsu (SL) [18], Yuleny [39], Huali [39], DocDLinkNet [76], DocUNet (WX) [39], Degraded, Hybrid, ScanBot.
  - **Python with GPU:** DilatedUNet [41], DocEnTR.
- **Device 3, Linux Pop!_OS 22.04**
  - **Python Machine Learning with CPU:** HuangBCD [33], HuangUnet [33], FourBi, ACM, LS-HDIB, AugraphyD, AugraphyF, DocBin1, DocBin2, DocBin3.
  - **C++:** Bataineh [3], Bernsen [6], ISauvola [17], Niblack [51], Nick [26], Otsu [52], Sauvola [60], Singh [64], Su-Lu [67], WAN [50], Wolf [73].
  - **Java:** Bradley [11], dSLR [63], Huang [42], Intermodes [53], IsoData [71], Johannsen [22], KSW [23], Li-Tam [28], Mean [16], Mello-Lins [46], MinError [27], Minimum[53], Moments [70], Percentile [13], Pun [54], RenyEntropy [59], Shanbhag [62], Triangle [75], Sliedes, Wu-Lu [44], Yean-CC [74], YinYang [41], YinYang21[41], YinYang22[32], YinYang23a[8], YinYang23b[8], YinYang23c[8], ZigZag[9].
- **Device 4:**
  - **CNN Python Algorithm**: T&M [68], ThreeStage.

No significant time difference was noticed for those algorithms that could be executed on different OS. The mean processing time was used in the analysis. The primary purpose here is to provide the order of magnitude of the processing time elapsed.

Monochromatic images are much smaller than their color equivalent as only one bit needed to represent each image pixel. Run-length encoding [55] the sequences of black and white pixels of a binary image is part of several lossless compression schemes for monochromatic [45, 46]. If the binarization process leaves salt-end-pepper noise in the final image, sometimes imperceptible to the human eye, that noise will break the sequence of pixels, degrading the performance of the image compression scheme, providing an indirect measure of the quality of the monochromatic image.

5 RESULTS

This DocEng'2025 quality, time and size binarization assessment of photographed documents evaluated the performance of 103 algorithms. The evaluation was performed considering the quality of the produced images and then the processing time and the mean compression rate of the TIFF G4 file was analysed. Some unpredictable problems were faced:

- The algorithms Akbari_1 [1], Akbari_2 [1], and Bhowmik [7] implemented for Windows for some reason were unable to run in the more recent architectures and versions of the operating system used here. Thus, they had to be excluded from this assessment.
- DocDLinkNet [76], HandwritteNet [39], Huali [39] and Yuleny [39] were very slow and did not produce any output for some images, thus they were discarded from the global assessment.
- Gosh [4] has problems in the installation set-up and is unable to be called on a batch of images, having to be invoked to process each image one-by-one. That unstable "behavior" was also dependent on the smartphone device model, and has

Table 2: Final Results – Dataset of Inkjet Printed Photographed Documents

Inkjet - PSNR									
	OFF			ON			AUTO		
	Alg.	*PSNR*	T(s)	Alg.	*PSNR*	T(s)	Alg.	*PSNR*	T(s)
#	Apple IPhone 14								
1	Jia-shi	65.40	50.6	Jia-shi	66.70	50.6	ThreeStage	65.23	1.26
2	ZigZag0s30	65.06	0.32	ZigZag0s60	66.65	0.29	Yasin	65.05	1.61
3	ZigZag0s60	65.01	0.29	ZigZag0s30	66.58	0.32	ZigZag0s10	64.89	0.32
4	ZigZag0s100	64.96	0.29	ZigZag0s100	66.53	0.29	ZigZag0s30	64.74	0.32
5	ZigZag0s10	64.87	0.32	ZigZag0s10	65.54	0.32	Degra	64.71	0.60
6	Scanbot	64.87	0.26	DocEnTR	65.51	6.37	ZigZag0s60	64.62	0.29
7	Degra	64.63	0.60	Scanbot	65.28	0.26	ZigZag0s100	64.57	0.29
#	Apple IPhone 15								
1	ThreeStage	65.52	1.27	Scanbot	66.00	0.26	ThreeStage	65.57	1.27
2	Yasin	65.29	1.61	Jia-shi	64.65	50.6	Yasin	65.10	1.61
3	ZigZag0s10	64.92	0.32	Degra	64.53	0.60	ZigZag0s10	64.66	0.32
4	Degra	64.66	0.60	ThreeStage	64.21	1.27	Degra	64.52	0.60
5	ZigZag0s30	64.62	0.32	ZigZag0s60	63.65	0.29	ZigZag0s30	64.25	0.32
6	ZigZag0s60	64.36	0.29	ZigZag0s100	63.61	0.29	ZigZag0s60	64.00	0.29
7	ZigZag0s100	64.29	0.29	ZigZag0s30	63.49	0.32	ZigZag0s100	63.98	0.29
#	Apple IPhone 16								
1	ThreeStage	66.27	1.27	ThreeStage	65.98	1.27	ThreeStage	66.31	1.27
2	Nick	66.12	0.21	Singh	65.76	0.15	Nick	66.15	0.21
3	Singh	66.10	0.15	Su-Lu	65.73	2.18	Singh	66.14	0.15
4	Su-Lu	65.93	2.18	Nick	65.72	0.21	Su-Lu	66.00	2.18
5	Wolf	65.80	0.29	Wolf	65.58	0.29	Wolf	65.94	0.29
6	Sauvola	65.22	0.21	Wu-Lu	65.14	0.03	Sauvola	65.26	0.21
7	Wu-Lu	64.31	0.03	YinYang21	64.51	1.61	Wu-Lu	64.48	0.03
#	Motorola Edge 50 pro								
1	Sauvola	65.46	0.02	Sauvola	65.74	0.02	Sauvola	65.88	0.02
2	Wolf	65.03	0.03	Wu-Lu	64.88	0.00	Wolf	65.14	0.03
3	Nick	64.89	0.02	ThreeStage	64.32	1.27	Nick	64.97	0.02
4	Singh	64.80	0.01	Nick	64.30	0.02	Singh	64.88	0.01
5	ThreeStage	64.79	1.27	Wolf	64.25	0.03	ThreeStage	64.78	1.27
6	Wu-Lu	64.42	0.00	Singh	64.20	0.01	Wu-Lu	64.62	0.00
7	Su-Lu	64.40	0.26	Su-Lu	63.63	0.26	Su-Lu	64.43	0.26
#	Samsung S23								
1	ThreeStage	65.27	1.27	ThreeStage	65.15	1.27	ThreeStage	65.19	1.27
2	Degra	65.08	0.60	Degra	64.66	0.60	Degra	65.07	0.60
3	Yasin	64.56	1.53	Yasin	63.66	1.53	Yasin	63.62	1.53
4	ZigZag0s10	64.11	0.32	ZigZag0s10	63.05	0.32	ZigZag0s10	63.18	0.32
5	ZigZag0s30	64.07	0.29	DeepOtsu	62.93	0.60	ZigZag0s30	63.12	0.29
6	ZigZag0s60	64.04	0.28	ZigZag0s30	62.93	0.29	Scanbot	63.10	0.26
7	ZigZag0s100	64.04	0.29	ZigZag0s60	62.84	0.28	ZigZag0s60	63.09	0.28
#	Samsung S24								
1	Scanbot	65.34	0.26	Scanbot	65.99	0.26	ThreeStage	65.11	1.27
2	ZigZag0s10	65.01	0.32	ThreeStage	64.73	1.27	Yasin	64.63	1.53
3	ZigZag0s30	64.93	0.29	Jia-shi	64.24	50.7	Degra	64.48	0.60
4	ZigZag0s100	64.85	0.29	ZigZag0s10	63.99	0.32	ZigZag0s60	64.29	0.28
5	ZigZag0s60	64.83	0.28	ZigZag0s30	63.93	0.29	ZigZag0s100	64.29	0.29
6	ThreeStage	64.81	1.27	Degra	63.88	0.60	ZigZag0s30	64.26	0.29
7	Yasin	64.74	1.53	Yasin	63.84	1.53	ZigZag0s10	64.17	0.32

not been observed whenever applied to scanned document images.

- The binarizations produced by the DocBin family of algorithms exhibited polarity inversion (black background and white text), requiring conversion to the expected format (white background and black text) prior to analysis. The time involved in such an image conversion was not computed here.
- The outputs generated by the ZigZag0 and ZigZag1 algorithms, including their kernel size variations, exhibited a resolution that differed from the images in the data set. Thus, to ensure consistency in the analysis, a resizing procedure

Table 3: Final Results – Dataset of Laser Printed Photographed Documents

Laser - PSNR									
	OFF			ON			AUTO		
	Alg.	*PSNR*	T(s)	Alg.	*PSNR*	T(s)	Alg.	*PSNR*	T(s)
#	Apple IPhone 14								
1	Scanbot	65.21	0.26	ThreeStage	64.45	1.26	jia-shi	65.98	50.9
2	jia-shi	64.70	50.9	yasin	64.30	1.62	Scanbot	65.69	0.26
3	ZigZag0s30	64.26	0.32	ZigZag0s10	64.00	0.32	DocEnTR	64.70	6.33
4	ZigZag0s60	64.22	0.29	Degra	63.82	0.60	ZigZag0s30	64.17	0.32
5	DocEnTR	64.19	6.33	ZigZag0s30	63.70	0.32	ZigZag0s60	64.07	0.29
6	ZigZag0s100	64.19	0.29	ZigZag0s60	63.56	0.29	ZigZag0s10	64.02	0.32
7	ZigZag0s10	64.16	0.32	ZigZag0s100	63.50	0.29	ZigZag0s100	63.96	0.29
#	Apple IPhone 15								
1	ThreeStage	65.11	1.26	ThreeStage	65.48	1.26	ThreeStage	65.12	1.26
2	yasin	64.46	1.58	yasin	64.69	1.58	yasin	64.76	1.58
3	ZigZag0s10	64.38	0.32	ZigZag0s10	64.63	0.32	ZigZag0s10	64.60	0.32
4	ZigZag0s30	64.27	0.30	Degra	64.53	0.60	ZigZag0s30	64.31	0.30
5	Scanbot	64.24	0.26	ZigZag0s30	64.10	0.30	Degra	64.26	0.60
6	Degra	64.15	0.60	ZigZag0s60	63.77	0.29	ZigZag0s60	64.12	0.29
7	ZigZag0s60	64.08	0.29	ZigZag0s100	63.69	0.29	ZigZag0s100	64.06	0.29
#	Apple IPhone 16								
1	Sauvola-cpp	66.44	0.21	ThreeStage	65.98	1.26	Sauvola-cpp	66.55	0.21
2	ThreeStage	66.27	1.26	Su-Lu	65.86	2.18	ThreeStage	66.23	1.26
3	Su-Lu	66.14	2.18	Nick-cpp	65.82	0.21	Su-Lu	66.11	2.18
4	Nick-cpp	65.85	0.21	Singh	65.80	0.15	Nick-cpp	65.85	0.21
5	Wolf-cpp	65.79	0.29	Sauvola-cpp	65.66	0.21	Wolf-cpp	65.79	0.29
6	Singh	65.74	0.15	Wolf-cpp	65.51	0.29	Singh	65.74	0.15
7	Wu-Lu	64.72	0.03	Wu-Lu	65.08	0.03	Wu-Lu	64.66	0.03
#	Motorola Edge 50 pro								
1	Wu-Lu	65.64	0.00	Wu-Lu	64.87	0.00	Wu-Lu	65.66	0.00
2	Shanbhag	64.99	0.00	Minimum	64.60	0.00	Su-Lu	64.80	0.26
3	Su-Lu	64.80	0.26	Sauvola-cpp	64.40	0.02	Sauvola-cpp	64.66	0.02
4	Sauvola-cpp	64.73	0.02	ThreeStage	64.23	1.26	ThreeStage	64.62	1.26
5	ThreeStage	64.59	1.26	Su-Lu	64.01	0.26	Li-Tam	64.57	0.00
6	Li-Tam	64.56	0.00	Degra	63.70	0.60	Shanbhag	64.51	0.00
7	Degra	63.99	0.60	Nick-cpp	63.63	0.02	Degra	64.00	0.60
#	Samsung S23								
1	Scanbot	65.63	0.26	ThreeStage	65.05	1.26	ThreeStage	65.23	1.26
2	ThreeStage	65.15	1.26	Scanbot	64.97	0.26	Degra	64.55	0.60
3	Degra	64.21	0.60	Degra	64.22	0.60	yasin	64.49	1.49
4	yasin	64.01	1.49	DeepOtsu	63.21	5.72	Scanbot	64.46	0.26
5	ZigZag0s30	63.94	0.28	yasin	63.20	1.49	ZigZag0s10	64.33	0.32
6	ZigZag0s60	63.93	0.29	ZigZag0s10	63.18	0.32	ZigZag0s30	64.33	0.28
7	ZigZag0s100	63.93	0.29	ZigZag0s30	63.14	0.28	ZigZag0s60	64.30	0.29
#	Samsung S24								
1	ThreeStage	64.95	1.26	Scanbot	67.92	0.26	ThreeStage	65.02	1.26
2	Scanbot	64.76	0.26	jia-shi	65.55	47.8	yasin	64.87	1.49
3	Degra	64.23	0.60	DocEnTR	65.18	5.88	ZigZag0s10	64.70	0.32
4	yasin	63.99	1.49	ZigZag0s10	64.68	0.32	ZigZag0s30	64.70	0.28
5	ZigZag0s30	63.82	0.28	ZigZag0s30	64.50	0.28	ZigZag0s60	64.69	0.29
6	ZigZag0s60	63.81	0.29	ZigZag0s60	64.27	0.29	ZigZag0s100	64.69	0.29
7	ZigZag0s10	63.81	0.32	ThreeStage	64.23	1.26	DeepOtsu	64.69	4.21

was applied using a filtering method. The resizing processing time needed was not computed here.

The PSNR measure between the binary images generated by the algorithms listed and their ground-truth equivalent was used to rank the algorithms. The mean processing time was taken to evaluate the order of magnitude of the time complexity of the algorithms. For all measures, the mean value is presented and the ranking is decided at the individual image level.

The standard deviation of the *PSNR*, for the laser dataset, is close to the mean error, while for Deskjet it is, for most cases, between 0.20 and 0.25, implying on no significant image-quality variations.

Table 4: TIFF Group 4 Compression Rates

#	Algorithm	%	#	Algorithm	%	#	Algorithm	%	#	Algorithm	%
1	Ergina Global	55.50	**19**	DocBin1	31.29	**37**	Wolf	28.69	**55**	Johannsen-Bille	20.82
2	Khairun2	54.92	**20**	Texture Based	30.66	**38**	ThreeStage	28.28	**56**	YinYang23a	19.69
3	Michalak21a	54.86	**21**	DaSilva-Lins-Rocha	30.22	**39**	Minimum	28.22	**57**	DE-GAN	19.44
4	ElisaTV	54.77	**22**	Bataineh	30.01	**40**	Scanbot	27.83	**58**	YinYang22	19.28
5	Michalak	54.72	**23**	Kapur-Sahoo-Wong	29.43	**41**	Vahid	27.78	**59**	AugraphyF	16.89
6	Percentile	53.48	**24**	Yean-Chang-Chang	29.36	**42**	ACM	27.49	**60**	DocEnTR	15.75
7	Niblack	52.73	**25**	Wu-Lu	29.27	**43**	Robin	24.57	**61**	AugraphyD	14.18
8	Calvo-Zaragoza	48.92	**26**	Otsu	29.24	**44**	T&M	24.51	**62**	LS-hdib	12.69
9	DeepOtsu	44.07	**27**	DocBin2	29.23	**45**	Degra	23.74	**63**	ZigZag3s100	9.26
10	Gattal	37.45	**28**	DocBin3	29.16	**46**	ZigZag1s10	22.83	**64**	ZigZag3s60	8.20
11	Akbari3	36.55	**29**	ISauvola	29.06	**47**	ZigZag1s100	22.74	**65**	ZigZag3s30	7.25
12	ZigZag0s10	32.37	**30**	Nick	29.02	**48**	ZigZag1s60	22.70	**66**	ZigZag3s10	6.50
13	ZigZag0s100	32.34	**31**	Singh	29.00	**49**	ZigZag1s30	22.69	**67**	Sliedes	2.92
14	ZigZag0s60	32.25	**32**	Moments	28.98	**50**	ZigZag2s100	22.64	**68**	ZigZag4s100	2.82
15	ZigZag0s30	32.22	**33**	Su-Lu	28.96	**51**	ZigZag2s60	22.64	**69**	ZigZag4s60	2.45
16	YinYang21	32.04	**34**	WAN	28.95	**52**	ZigZag2s10	22.62	**70**	ZigZag4s30	2.08
17	Triangle	31.63	**35**	IsoData-ORIG	28.76	**53**	ZigZag2s30	22.62	**71**	ZigZag4s10	1.86
18	Hybrid	31.36	**36**	Bradley	28.74	**54**	YinYang23c	22.28			

Tables 2 and 3 present the top-7 quality results for the algorithms tested, and the six devices assessed. The algorithms name printed in **blue** were the algorithms of this year competitors. By analyzing the results, several conclusions may be drawn:

1. The five new algorithms enrolled in this competition often appear in the top-7 best quality images, showing that the research in this area is yielding better quality algorithms.

2. The ranking order vary among the several configurations of flash and capture device, which reinforces the claim that no binarization algorithm is good for all document images.

3. Using the same device and capture angle, but turning the strobe flash *on* or *off* had no impact on the PSNR for the deskjet printed dataset, however if using the $[L_{dist}]$, for all cases, the strobe flash state strongly impacts the algorithms ranking.

4. In some cases, some old classical algorithms had the best performance. They require a much smaller processing time than the new algorithms, but generated equally good binary images.

5. The the ACM DocEng 2024 ZigZag algorithm [9] presented outstanding quality images with a very reasonable processing time, appearing in the top-7 for all devices, strobe-flash set ups, and document types. Besides that, it also provides good binary image compression capability.

6. It is amazing to find some of the classical algorithms such as IsoData, Moments, Otsu, etc. appearing in the top-7 tables presented. The fact that such algorithms are global makes surprising such a good performance. In general those algorithms present excellent time efficiency also.

7. The algorithm by Tensmeyer and Martinez [68] (T&C) was retrained using only 24 images of the DIB platform, none of which used in the test phase.

8. It is important to remark that the training time of ML algorithms was not considered here.

9. The slowest algorithms were **DilatedUNet**, **ACM**, **Vahid Akbari3**, **Howe**, **Gattal**, **DocUNet**, **Lu-Su** and **T&C** which also appear as the best or among the best in image quality for the different devices and setups.

10. Jia-Shi algorithm is an outstanding example on how important it is to assess binarization processing time, as although providing good-quality binary images, the elapsed time makes it not viable to most of applications.

Google Vision was also used to perform Optical Character Recognition (OCR) on the documents. The Levenshtein distance (L_{dist} - the number of characters inserted, deleted or exchanged between two strings) of the document transcription (#*char*) and the original text was calculated. The error rate is calculated as:

$$[L_{dist}] = (\#char - L_{dist})/\#char \quad (1)$$

The OCR transcription measure is a functional test that avoids any global biases or oversights that can occur with the PSNR measure. The transcription error rate for the PSNR-top 15 algorithms was very low (around 0.9%), although the algorithm rank is not the same as in the PSNR-rank. The results are uniform and the transcriptions are, for the PSNR-top 15 algorithms, very close the ground-truth text. Such results show that there was an evolution in the quality of the transcription of the images in comparison to the previous ACM DocEng Binarization Competitions. That source for such an increase in performance may be credited to the devices and/or Google Vision OCR.

Monochromatic images are much smaller than their color equivalent as only one bit needed to represent each image pixel. This means that assessing the size of the resulting image file using a lossless compression scheme is also a relevant element for comparison among binarization algorithms. Besides that, the binary image may be the key for generating color synthetic images which are visually indistinguishable of the original document whenever

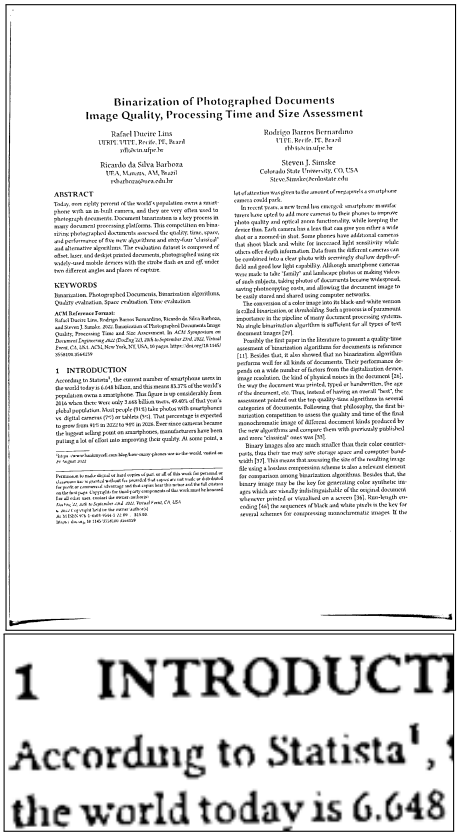

Figure 3: Top - Binarized image of a inkjet-printed document captured with a Motorola Edge 50 Pro in artificial lighting, produced by the ThreeStage algorithm. Bottom - Zoom into a part of the top image.

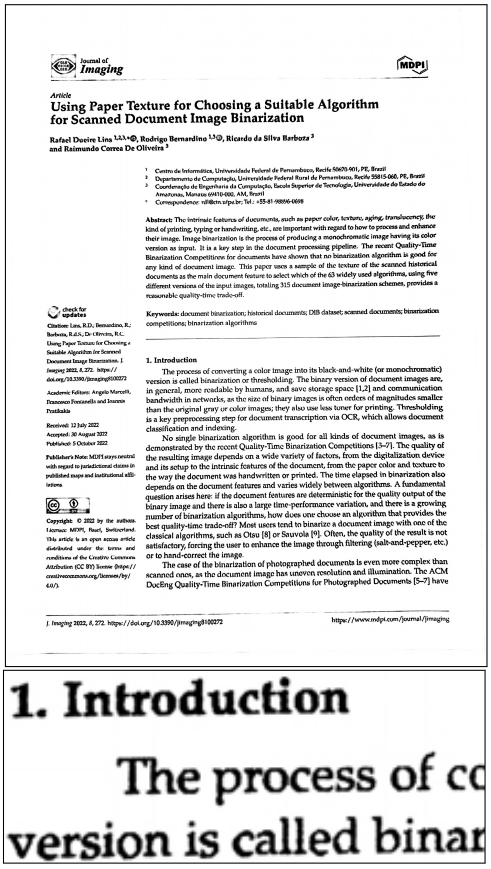

Figure 4: Top - Binarized image of inkjet-printed document captured under artificial lighting, generated by the Sliedes algorithm applied to a photo taken with an iPhone 16 Pro (flash on). Bottom - Zoom into a part of the top image.

printed or visualized on a screen [45]. Run-length encoding [55] the sequences of black and white pixels of a binary image is part of several lossless compression schemes for monochromatic [46] in its transmission through computer networks. If the binarization process leaves salt-end-pepper noise in the final image, sometimes imperceptible to the human eye, that noise will break the sequence of pixels, degrading the performance of the image compression scheme. Indirectly, that can be also seen as a measure of the quality of the monochromatic image, which was pioneered in this competition series in 2023 [37]. Table 4 presents the mean compression rate of the TIFF G4 file in relation to the size of the uncompressed PNG monochromatic image stored in gray scale format, for the top-7 quality binarization algorithms for the six devices, with the data sets and configurations tested. One may observe that some algorithms such as Ergina Global brings a good compression rate with an average size reduction of 55.50% of the size of the gray scale PNG file.

Due to space limitations the test set of offset printed documents generated for the devices tested here using the same set of algorithms had to be suppressed. Although, the performance of the algorithms vary from one test set or device to another, there was no anomalous behavior observed for the algorithms tested.

Feeding the binarization algorithms with the different RGB channels may yield a better quality two-tone image, besides saving processing time [31].

Some combination of multiple excellent algorithms presented here is likely to provide even better quality results with a time overhead, however.

ACKNOWLEDGEMENTS

The organisers of this competition are very grateful to all teams enrolled, and to all those who made the code for their algorithms available, for their cooperation and good spirit. The assessment of algorithms and comments on their ability to be fully assessed in no way reflect on the high quality of research and collaboration shown by all of the teams entering the contest.

Rafael Dueire Lins holds a Senior Researcher Grant from CNPq - Brazil.

REFERENCES

[1] Younes Akbari et al. 2019. Binarization of Degraded Document Images using Convolutional Neural Networks based on predicted Two-Channel Images. In *ICDAR*.

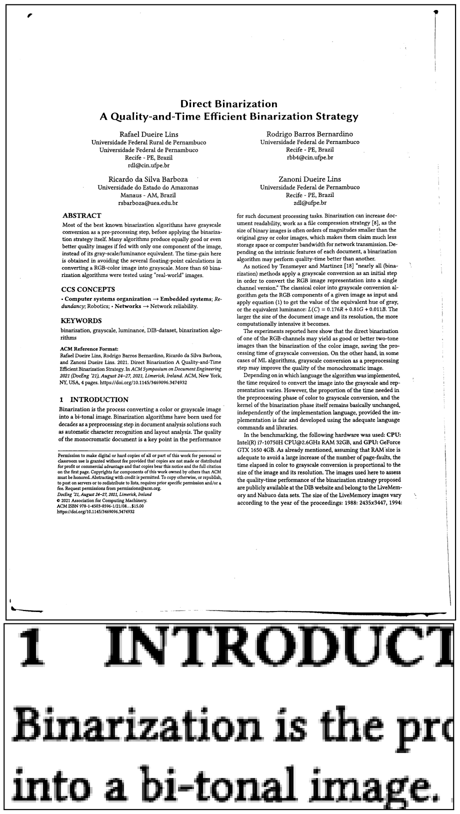

Figure 5: Top - Binarized image of a inkjet-printed document captured with a Samsung S24 in artificial-light conditions, produced by the Scanbot algorithm.Bottom - Zoom into a part of the top image.

Figure 6: Top - Binarized image of a laser-printed document captured with a Samsung S23 in window-light conditions produced by the Degra algorithm. Bottom - Zoom into a part of the top image.

[2] Elisa H. Barney Smith, Laurence Likforman-Sulem, and Jérôme Darbon. 2010. Effect of Pre-processing on Binarization. In *Document Recognition and Retrieval XVII*. 75340H.

[3] Bilal Bataineh et al. 2011. An adaptive local bin. method for doc. images based on a novel thresh. method and dynamic windows. *Ptrn. Recog. Letters* 32, 14 (2011).

[4] Suman Kumar Bera et al. 2021. A non-parametric binarization method based on ensemble of clustering algorithms. *Multimedia Tools and Applications* 80, 5 (2021), 7653–7673.

[5] Rodrigo Bernardino, Rafael Dueire Lins, and Ricardo Barboza. 2024. Texture-based Document Binarization. In *Proceedings of ACM Symposium on Document Engineering 2024 (DocEng '24)*. ACM, New York, NY, USA.

[6] J Bernsen. 1986. Dynamic thresholding of gray-level images. In *International Conference on Pattern Recognition*. 1251–1255.

[7] Showmik Bhowmik, Ram Sarkar, Bishwadeep Das, and David Doermann. 2018. GiB: a Game theory Inspired Binarization technique for degraded document images. *IEEE Transactions on Image Processing* 28, 3 (2018), 1443–1455.

[8] Jean-Luc Bloechle, Jean Hennebert, and Christophe Gisler. 2023. YinYang, a Fast and Robust Adaptive Document Image Binarization for Optical Character Recognition. In *Proceedings of the ACM Symposium on Document Engineering 2023* (Limerick, Ireland) *(DocEng '23)*. Association for Computing Machinery, Article 19, 4 pages. https://doi.org/10.1145/3573128.3609354

[9] Jean-Luc Bloechle, Jean Hennebert, and Christophe Gisler. 2024. ZigZag: A Robust Adaptive Approach to Non-Uniformly Illuminated Document Image Binarization. In *Proceedings of ACM Symposium on Document Engineering 2024 (DocEng '24)*. ACM, New York, NY, USA.

[10] Bolan Su, Shijian Lu, and Chew Lim Tan. 2013. Robust Document Image Binarization Technique for Degraded Document Images. *T. on I. Processing* 22, 4 (2013), 1408–1417.

[11] Derek Bradley and Gerhard Roth. 2007. Adaptive Thresholding using the Integral Image. *Journal of Graphics Tools* 12, 2 (2007), 13–21.

[12] Jorge Calvo-Zaragoza and Antonio-Javier Gallego. 2019. A selectional auto-encoder approach for document image binarization. *Pattern Recognition* 86 (2019), 37–47.

[13] W. Doyle. 1962. Operations Useful for Similarity-Invariant Pattern Recognition. *J. ACM* 9, 2 (1962), 259–267.

[14] Rafael Dueire Lins, Rodrigo Bernardino, and Darlisson Marinho Jesus. 2019. A Quality and Time Assessment of Binarization Algorithms. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*. 1444–1450. https://doi.org/10.1109/ICDAR.2019.00232

[15] Abdeljalil Gattal, Faycel Abbas, and Mohamed Ridda Laouar. 2018. Automatic Parameter Tuning of K-Means Algorithm for Document Binarization. In *7th ICSENT*. ACM Press, 1–4.

[16] C Glasbey. 1993. An Analysis of Histogram-Based Thresholding Algorithms. *Graphical Models and Image Processing* 55, 6 (1993), 532–537.

[17] Zineb Hadjadj, Abdelkrim Meziane, Yazid Cherfa, Mohamed Cheriet, and Insaf Setitra. [n.d.]. ISauvola: Improved Sauvola's Algorithm for Document Image Binarization. 737–745.

[18] Sheng He and Lambert Schomaker. 2019. DeepOtsu: Document Enhancement and Binarization using Iterative Deep Learning. *Pattern Recognition* 91 (jan 2019), 379–390.

[19] Nicholas R. Howe. 2013. Doc. binarization with automatic parameter tuning. *IJDAR* 16 (2013).

[20] Liang Kai Huang and Mao Jiun J. Wang. 1995. Image thresholding by minimizing the measures of fuzziness. *Pattern Recognition* 28, 1 (1995), 41–51.

[21] Fuxi Jia, Cunzhao Shi, Kun He, Chunheng Wang, and Baihua Xiao. 2018. Degraded document image binarization using structural symmetry of strokes. *Pattern Recognition* 74 (2018), 225–240.

[22] J Johannsen, G and Bille. 1982. A threshold selection method using information measures. In *Int'l Conf. Pattern Recognition*. 140–143.

[23] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong. [n.d.]. A new method for gray-level picture thresholding using the entropy of the histogram. *C. Vision, Graphics, I. Processing* 29, 1 ([n. d.]), 140.

[24] Ergina Kavallieratou. 2005. A binarization algorithm specialized on document images and photos. *ICDAR* 2005, 1 (2005), 463–467.

[25] Ergina Kavallieratou and Stamatatos Stathis. 2006. Adaptive binarization of historical document images. *Proceedings - International Conference on Pattern Recognition* 3 (2006), 742–745.

[26] Khurram Khurshid, Imran Siddiqi, Claudie Faure, and Nicole Vincent. 2009. Comparison of Niblack inspired binarization methods for ancient documents. In *SPIE*. 72470U.

[27] J. Kittler et al. 1986. Minimum error thresholding. *Pattrn. Recog.* 19, 1 (1986).

[28] C.H. Li and P.K.S. Tam. 1998. An iterative algorithm for minimum cross entropy thresholding. *Pattern Recognition Letters* 19, 8 (1998), 771–776.

[29] Rafael Dueire Lins. 2009. A Taxonomy for Noise in Images of Paper Documents - The Physical Noises. In *Lecture Notes in Computer Science*, Vol. 5627 LNCS. 844–854. https://doi.org/10.1007/978-3-642-02611-9_83

[30] Rafael Dueire Lins. 2011. Nabuco Two Decades of Document Processing in Latin America . *Journal of Universal Computer Science* 17, 1 (2011), 151–161.

[31] Rafael Dueire Lins, R. B. Bernardino, et al. 2021. DocEng'2021 Direct Binarization A Quality-and-Time Efficient Binarization Strategy. In *DocEng 2021*. ACM.

[32] Rafael Dueire Lins, Rodrigo Barros Bernardino, Ricardo Barboza, and Steven J. Simske. 2022. DocEng'2022 Quality, Space, and Time Competition on Binarizing Photographed Documents. In *DocEng'22*. ACM, 1–4.

[33] Rafael Dueire Lins, Rodrigo Barros Bernardino, Elisa Barney Smith, and Ergina Kavallieratou. [n.d.]. ICDAR 2021 Competition on Time-Quality Document Image Binarization. In *ICDAR 2021*. 1539–1546. https://doi.org/10.1007/978-3-030-86337-1_47

[34] Rafael Dueire Lins, R. B. Bernardino, and et.al. 2017. Binarizing Document Images Acquired with Portable Cameras. In *2017 14th ICDAR*. IEEE.

[35] Rafael Dueire Lins, Rodrigo Barros Bernardino, and Steven J. Simske. 2021. DocEng'2021 Time-Quality Competition on Binarizing Photographed Documents. In *DocEng'2021*. ACM, 1–4. https://doi.org/10.1145/3395027.3419578

[36] Rafael Dueire Lins, Gustavo P. Chaves, Gabriel de F. P e Silva, Thaylor Vieira, Ricardo da Silva Barboza, and Steven J. Simske. 2024. Competition on Binarizing Photographed Document Images 2024 Quality, Time and Space Report. In *Proceedings of the ACM Symposium on Document Engineering 2024* (San Jose, CA, USA) *(DocEng '24)*. Association for Computing Machinery, New York, NY, USA, Article 11, 12 pages. https://doi.org/10.1145/3685650.3686793

[37] Rafael Dueire Lins, Gabriel de F. Pe Silva, Gustavo P. Chaves, Ricardo da Silva Barboza, Rodrigo Barros Bernardino, and Steven J. Simske. 2023. Quality, Space and Time Competition on Binarizing Photographed Document Images. In *Proceedings of the ACM Symposium on Document Engineering 2023* (Limerick, Ireland) *(DocEng '23)*. Association for Computing Machinery, Article 5, 10 pages. https://doi.org/10.1145/3573128.3604903

[38] Rafael Dueire Lins, Gabriel de F. Pe Silva, Gustavo P. Chaves, Ricardo da Silva Barboza, Rodrigo Barros Bernardino, and Steven J. Simske. 2023. Quality, Space and Time Competition on Binarizing Photographed Document Images. In *Proceedings of the ACM Symposium on Document Engineering 2023* (Limerick, Ireland) *(DocEng '23)*. Association for Computing Machinery, Article 5, 10 pages. https://doi.org/10.1145/3573128.3604903

[39] Rafael Dueire Lins, Ergina Kavallieratou, Elisa Barney Smith, Rodrigo Barros Bernardino, and Darlisson Marinho de Jesus. [n.d.]. ICDAR 2019 Time-Quality Binarization Competition. In *ICDAR*. 1539–1546. https://doi.org/10.1109/ICDAR.2019.00248

[40] Rafael Dueire Lins and Domingos Machado. 2004. Comparative study of file formats for image storage and transmission. *Journal of Electronic Imaging* 13, 1 (2004), 175 – 181. https://doi.org/10.1117/1.1634591

[41] Rafael Dueire Lins, Steven J. Simske, and Rodrigo Barros Bernardino. 2020. DocEng'2020 Time-Quality Competition on Binarizing Photographed Documents. In *DocEng '20: ACM Symposium on Document Engineering 2020, Virtual Event, CA, USA, September 29 - October 1, 2020*. ACM. https://doi.org/10.1145/3395027.3419578

[42] Di Lu, Xin Huang, and LiXue Xue Sui. 2018. Binarization of degraded document images based on contrast enhancement. *IJDAR* 21, 1-2 (2018), 123–135.

[43] Shijian Lu, Bolan Su, and Chew Lim Tan. 2010. Document image binarization using background estimation and stroke edges. *IJDAR* 13, 4 (2010), 303–314.

[44] Wu Lu, Ma Songde, and Hanqing Lu. 1998. An effective entropic thresholding for ultrasonic images. *14th ICPR* (1998), 1552–1554, vol. 2.

[45] Carlos A.B. Mello and Rafael Dueire Lins. 2002. Generation of images of historical documents by composition. In *DocEng '02: Proceedings of the 2002 ACM symposium on Document engineering*. 127–133. https://doi.org/10.1145/585058.585082

[46] Carlos A. B. Mello and Rafael Dueire Lins. 2000. Image segmentation of historical documents. *Visual 2000* (2000).

[47] Hubert Michalak and Krzysztof Okarma. 2019. Adaptive image binarization based on multi-layered stack of regions. In *International Conference on Computer Analysis of Images and Patterns*. Springer, 281–293.

[48] Hubert Michalak and Krzysztof Okarma. 2019. Fast Binarization of Unevenly Illuminated Document Images Based on Background Estimation for Optical Character Recognition Purposes. *J. Univers. Comput. Sci.* 25, 6 (2019), 627–646.

[49] Hubert Michalak and Krzysztof Okarma. 2019. Improvement of image binarization methods using image preprocessing with local entropy filtering for alphanumerical character recognition purposes. *entropy* 21, 6 (2019), 562.

[50] Wan Azani Mustafa and Mohamed Mydin M. Abdul Kader. 2018. Binarization of Document Image Using Optimum Threshold Modification. *J. Physics: C. Series* 1019, 1 (2018), 012022.

[51] Wayne Niblack. 1985. *An introduction to digital image processing*. Strandberg.

[52] Nobuyuki Otsu. 1979. A threshold selection method from gray-level histograms. *IEEE T. on Systems, Man, and Cybernetics* 9, 1 (1979), 62–66.

[53] Judith M. S. Prewitt and Mortimer L. Mendelsohn. 2006. The Analysis of Cell Images. *Annals of the New York Academy of Sciences* 128, 3 (2006), 1035–1053.

[54] T. Pun. 1981. Entropic thresholding, a new approach. *Computer Graphics and Image Processing* 16, 3 (1981), 210–239.

[55] A.H. Robinson and C. Cherry. 1967. Results of a prototype television bandwidth compression scheme. *Proc. IEEE* 55, 3 (1967), 356–364. https://doi.org/10.1109/PROC.1967.5493

[56] Khairun Saddami, Putri Afrah, Viska Mutiawani, and Fitri Arnia. 2018. A New Adaptive Thresholding Technique for Binarizing Ancient Document. In *INAPR*. IEEE, 57–61.

[57] Khairun Saddami, Khairul Munadi, Yuwaldi Away, and Fitri Arnia. 2019. Effective and fast binarization method for combined degradation on ancient documents. *Heliyon* (2019).

[58] Khairun Saddami, Khairul Munadi, Sayed Muchallil, and Fitri Arnia. 2017. Improved Thresholding Method for Enhancing Jawi Binarization Performance. In *ICDAR*, Vol. 1. IEEE.

[59] Prasanna Sahoo, Carrye Wilkins, and Jerry Yeager. 1997. Threshold selection using Renyi's entropy. *Pattern Recognition* 30, 1 (1997), 71–84.

[60] J. Sauvola, M. Pietikäinen, and M Pietikainem. 2000. Adaptive document image binarization. *Pattern Recognition* 33, 2 (2000), 225–236.

[61] Jaakko Sauvola, Tapio Seppanen, Sami Haapakoski, and Matti Pietikainen. 1997. Adaptive document binarization. In *ICDAR*, Vol. 1. IEEE Comput. Soc, 147–152.

[62] A.G. G Shanbhag. 1994. Utilization of Information Measure as a Means of Image Thresholding. *CVGIP: Graphical Models and Image Processing* 56, 5 (1994), 414–419.

[63] J M M Silva, Rafael D. Lins, and Valdemar C Rocha. 2006. Binarizing and Filtering Historical Documents with Back-to-Front Interference. In *ACM SAC 2006*. 853–858. https://doi.org/10.1145/1141277.1141471

[64] T. Romen Singh, Sudipta Roy, O. Imocha Singh, Tejmani Sinam, and Kh. Manglem Singh. 2011. A New Local Adaptive Thresholding Technique in Binarization. *IJCSI* 08, 6 (2011), 271–277.

[65] Mohamed Ali Souibgui and Yousri Kessentini. 2021. DE-GAN: A Conditional Generative Adversarial Network for Document Enhancement. *Ptrn. Analysis and Machine Intellig.* (2021).

[66] Mohamed Ali Souibgui and Yousri Kessentini. 2022. DE-GAN: A Conditional Generative Adversarial Network for Document Enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 3 (2022), 1180–1191. https://doi.org/10.1109/TPAMI.2020.3022406

[67] Bolan Su, Shijian Lu, and Chew Lim Tan. 2010. Binarization of historical document images using the local maximum and minimum. In *8th IAPR DAS*. ACM Press, 159–166.

[68] Chris Tensmeyer and Tony Martinez. 2017. Document Image Binarization with Fully Convolutional Neural Networks. 99–104. https://doi.org/10.1109/ICDAR.2017.25

[69] Chris Tensmeyer and Tony Martinez. 2020. Historical document image binarization: a review. *SN Computer Science* 1, 3 (2020), 1–26. https://doi.org/10.1007/s42979-020-00176-1

[70] Wen-Hsiang Tsai. 1985. Moment-preserving thresolding: A new approach. *Computer Vision, Graphics, and Image Processing* 29, 3 (1985), 377–393.

[71] Flavio R. Velasco. 1979. *Thresholding Using the Isodata Clustering Algorithm*. Technical Report. OSD or Non-Service DoD Agency. 14 pages.

[72] Christian Wolf and David Doermann. 2002. Binarization of low quality text using a Markov random field model. In *Object recognition supported by user interaction for service robots*, Vol. 3. IEEE Comput. Soc, 160–163.

[73] C. Wolf, J.-M. Jolion, and F. Chassaing. 2002. Text localization, enhancement and binarization in multimedia documents. In *2002 International Conference on Pattern Recognition*, Vol. 2. 1037–1040 vol.2. https://doi.org/10.1109/ICPR.2002.1048482

[74] F J.; Chang S Yen J. C.; Chang, Jui Cheng Yen, Fu Juay Chang, and Shyang Chang. 1995. A New Criterion for Automatic Multilevel Thresholding. *T. on Image Processing* 4, 3 (1995), 370–378.

[75] G W Zack, W E Rogers, and S A Latt. 1977. Automatic measurement of sister chromatid exchange frequency. *J. Histochemistry and Cytochemistry* 25, 7 (1977), 741–753.

[76] Lichen Zhou et al. 2018. D-linknet: Linknet with pretrained encoder and dilated convolution for satellite imagery road extraction. In *Comp. Vision and Ptrn. Recog.*