

A binarization method for logos

Zhentaο Fan

Guangdong University of Technology Guangzhou, Guangdong, 510006, P.R. China

loadingovo@gmail.com

Abstract. In the field of digital image processing, the binarization of logo images presents a significant challenge due to the diversity in their color distribution and graphical structures. Logos, as distinctive visual symbols, are integral to brand recognition and commercial advertising. Traditional binarization methods often struggle with this diversity, leading to difficulties in preserving the original color and details of the images. To address this issue, this paper introduces a novel binarization approach specifically designed for logo images. The method comprises three stages: preprocessing, binarization, and noise removal. In the binarization stage, the Otsu method is adapted from a global to a local thresholding technique, combining both global and local thresholds for each pixel. The efficacy of this method is demonstrated through comparative experiments with logos of varying colors and structures, showing superior performance in maintaining image color and detail restoration over traditional methods. Additionally, the preprocessing and noise removal steps further enhance the method's effectiveness in handling logos with rich background colors and complex details. Overall, this approach proves to be highly adaptable across a wide range of logo types.

Keywords: Binarization, Local Threshold, Adaptive, Global Threshold, Optimize

1. Introduction

In today's digital age, image processing technology plays an increasingly significant role in various fields. Among these, binarization technology, as one of the fundamentals of image processing [1], finds wide applications such as character recognition [2-5], edge detection [6], and image segmentation [2]. Binarization of simple images, like brand logos, is also a part of this, with binarized logos being used for tasks such as image recognition, printing, and faxing. When selecting a binarization method [7], factors like similarity to the original image, recognition accuracy, and aesthetic appeal need to be considered which making the choice of a binarization method a crucial decision.

Binarization is a method that transforming pixel values into black and white in an image based on the grayscale characteristics of the target objects and the background, aiming to delineate the objects from the background. There are several binarization methods are widely used, and they can be broadly categorized into two types: global thresholding method and local thresholding method. Common global thresholding methods include the Mean Gray Thresholding and Otsu's methods. Local thresholding methods include the Sauvola method and the Niblack method. The key distinction between these two approaches lies in whether a single global threshold is applied to the entire image or if the image is divided into multiple local windows, each with its independent threshold. Global thresholding methods are straightforward and can intuitively process images [8]. They can yield excellent results for images with simple structures and colors, resulting in strong image consistency. Local thresholding methods are

better suited for complex scenes, such as those with non-uniform lighting and intricate color patterns, as they consider the variability in local pixels.

When dealing with small-scale images like logos, global thresholding methods have an advantage when the images have simple colors and structures, while local thresholding methods are more effective in the opposite scenarios. Therefore, traditional methods have limitations and cannot be widely applied to any image. To solve this problem, this article proposes a new method for small images such as logos. This method combines the advantages of global thresholding and local thresholding methods and is suitable for various logo images. The binarization effect is not affected by the color and effect of the image.

The paper will begin with an introduction to common binarization techniques, such as the Mean Gray Thresholding method, Otsu's method, Sauvola's method, and others. It will then conduct a comparative analysis of the strengths and weaknesses of current binarization techniques when applied to logos. Subsequently, it will propose an improved local thresholding method based on Otsu's method, compare the advantages and disadvantages of different methods, and present a comprehensive solution suitable for binarizing logos.

2. Common Binarization Method Overview

2.1. Otsu

Otsu's method, also referred to as the Maximum Interclass Variance method [3], is named for its objective of determining a threshold for image binarization that maximizes the interclass variance between the segmented foreground and background. Widely acknowledged as the optimal approach for threshold selection in image segmentation, this method boasts simplicity, resilience to variations in image brightness and contrast, and versatility in digital image processing. Its fundamental principle involves categorizing an image into background and foreground components based on grayscale characteristics.

The interclass variance serves as a metric for the uniformity of grayscale distribution, where a higher interclass variance signifies greater dissimilarity between the image's background and foreground. Misclassifying part of the foreground as background or vice versa diminishes the interclass difference. Therefore, maximizing interclass variance during segmentation translates to minimizing the probability of misclassification.

For an $M \times N$ -sized image denoted as $I(x, y)$, the threshold for foreground-background segmentation is represented as T . The proportion of pixels constituting the foreground in the entire image is ω_0 , characterized by an average gray level of μ_0 . Simultaneously, the proportion of pixels constituting the background is ω_1 , with an average gray level of μ_1 . The overall average gray level of the image is μ , and the interclass variance is denoted as g .

The count of pixels possessing grayscale values below threshold T is N_0 , while the count of pixels with grayscale values surpassing threshold T is N_1 . Consequently, the following relationships: [1,8,9]

$$\omega_0 = N_0 / M \times N \quad (1)$$

$$\omega_1 = N_1 / M \times N \quad (2)$$

$$N_0 + N_1 = M \times N \quad (3)$$

$$\omega_0 + \omega_1 = 1 \quad (4)$$

$$\mu = \omega_0 * \mu_0 + \omega_1 * \mu_1 \quad (5)$$

$$g = \omega_0(\mu_0 - \mu)^2 + \omega_1(\mu_1 - \mu)^2 \quad (6)$$

2.2. Sauvola

The Sauvola method [10] is a local adaptive thresholding technique designed for the conversion of grayscale images into binary images. Introduced by Sauvola and others in 1997, this method offers

effective binarization for images subjected to varying illumination conditions across different regions. Unlike global thresholding methods, the Sauvola method calculates the threshold for each pixel based on the local mean and standard deviation within its neighborhood.

The fundamental concept of the Sauvola method revolves around determining the threshold by considering the local characteristics of the image. For every pixel, the method computes both the mean and standard deviation within its surrounding window. The threshold is then determined using the following formula: [2,11,12]

$$T = m * (1 + k * (s/r - 1))$$

Where:

- T is the threshold for the pixel.
- m is the mean value of the pixels within the window.
- s is the standard deviation of the pixels within the window.
- k is a parameter of the method.
- r is a specified constant.

If the grayscale value of a pixel is greater than the threshold T, it is set to white (255); otherwise, it is set to black (0). This adaptive approach allows the Sauvola method to effectively binarize images with varying local lighting conditions.

2.3. Adaptive Threshold

The ADAPTIVE_THRESH_MEAN_C is an adaptive thresholding method that calculates the threshold for each pixel based on the local mean and a constant value. Specifically, for each pixel, the method first computes the average (mean) value "m" of the pixels within a local window and then subtracts a constant "C" to obtain the threshold "T." This can be expressed as follows:

$$T = \frac{1}{n} \sum_{i=1}^n p_i - C$$

$$T = m - C$$

Here:

- "T" is the threshold for the pixel.
- "m" is the mean value of the pixels within the local window.
- "C" is a parameter of the method, which can be either a positive or negative value.
- "n" represents the number of pixels within the local window.

If the grayscale value of a pixel is greater than the threshold "T," it is set to white (255); otherwise, it is set to black (0). This adaptive approach allows the ADAPTIVE_THRESH_MEAN_C method to calculate thresholds based on local pixel statistics, which can be beneficial for images with varying lighting conditions.

3. The Proposed Binarization Method for Logos

Addressing the issue of diverse colors and complex structures in logo images, global methods like Otsu tend to lose a significant amount of detail, while local methods like Sauvola require manual parameter tuning and may suffer from global limitations, resulting in potential noise. Therefore, a logo-specific binarization method is proposed that balances the advantages of both global and local approaches. This approach involves a three-step process: preprocessing binarization, binarization, and noise removal, which takes into account both global and local considerations.

3.1. Flowchart

As shown in Figure 1, the provided diagram is a succinct representation of the stages involved in the image processing workflow. Commencing with the 'original image,' the process advances to 'preprocessing,' which convert the image to grayscale [9] and perform some preliminary binarization

processing based on the histogram of the image for subsequent steps. The next phase, 'binarization,' involves converting the image into a binary form, reducing it to fundamental black and white pixels, which simplifies analysis and processing. Subsequently, 'noise removal' is applied to purge the image of extraneous artifacts that may confound the analysis. The culmination of this systematic procedure is the 'final image,' which, having been refined through these steps, is presumably optimized for further use or examination.

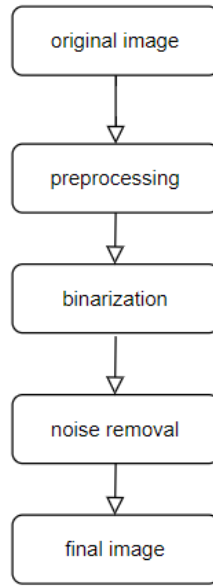


Figure 1. Flowchart

3.2. Preprocessing

By analyzing the grayscale histogram of an image, the first peak value (First_peak) and the last peak value (Last_peak) are obtained. Pixels with grayscale values less than or equal to First_peak are set to 0, and pixels with grayscale values greater than Last_peak are set to 255. For logo images where the background is typically white or black, this preprocessing step ensures that the original image, when converted to grayscale, preserves the black and white regions. Additionally, it enhances the accuracy of subsequent binarization operations.

$$Image[x, y] = \begin{cases} 255, & Image[x, y] > LastPeak \\ 0, & Image[x, y] < FirstPeak \\ Image[x, y], & others \end{cases}$$

3.3. Improved Local Thresholding Method Based on the Otsu Method

The improved local thresholding method is based on Otsu's global threshold and Otsu's local threshold. The final threshold is obtained by weighting these two thresholds. The specific calculation process is as follows:

1. Begin by calculating the Otsu global threshold for the global image.
2. Calculate the local threshold for a fixed region with the current pixel as the center.
3. Utilize weighting parameters to combine the global Otsu global threshold and the Otsu local threshold for each pixel.
4. Finally, obtain the threshold T for each pixel, considering both global and local characteristics.

$$T = (1 - k)globalOtsu + k * localOtsu$$

Pseudocode:

Algorithm 1: Local Otsu Thresholding

Data: Image *image*, radius *r*, global mean *global_mean*, weight *k*

Result: Binarized image *new_image*

```

1 for each row i in range(height) do
2   for each column j in range(width) do
3     roi ← extract_local_region(image, i, j, r, height, width);
4     mean ← calculate_otsu_threshold(roi);
5     threshold ← (1 − k) · global_mean + k · local_mean;
6     if threshold > image[i][j] then
7       | new_image[i][j] ← 0;
8     end
9     else
10      | new_image[i][j] ← 255;
11    end
12  end
13 end

```

As shown in Algorithm 1, the algorithm uses a fixed-size sliding window to traverse every pixel in the image. It calculates the threshold for each pixel by combining the global threshold and the window threshold calculation formula. The parameters ‘r’ and ‘k’ in this method need to be manually adjusted to achieve the best effect. This approach combines the advantages of global and local thresholding, resulting in an improved method for adaptive thresholding.

3.4. Noise Removal

To remove noise, a sliding window is used to traverse each pixel. When more than 60% of the pixels within the window have a grayscale value of 255, the center pixel of the window is also set to a grayscale value of 255. Conversely, when more than 60% of the pixels within the window have a grayscale value of 0, the center pixel of the window is also set to a grayscale value of 0. After experimenting with multiple samples, it was determined that a window size of 2x2 is the optimal choice.

$$p = \begin{cases} 255, \sum_{i=1}^n p_i > 0.6 * n^2 * 255 \\ 0, \sum_{i=1}^n p_i < 0.4 * n^2 * 255 \end{cases}$$

4. Experimental Results and Comparisons

4.1. Experimental Results

4.1.1. Preprocessing. Obtain the grayscale histogram of the image and examine the peaks in the histogram. Record the grayscale value of the first peak encountered as "FirstPeak" and the value of the last peak as "LastPeak".

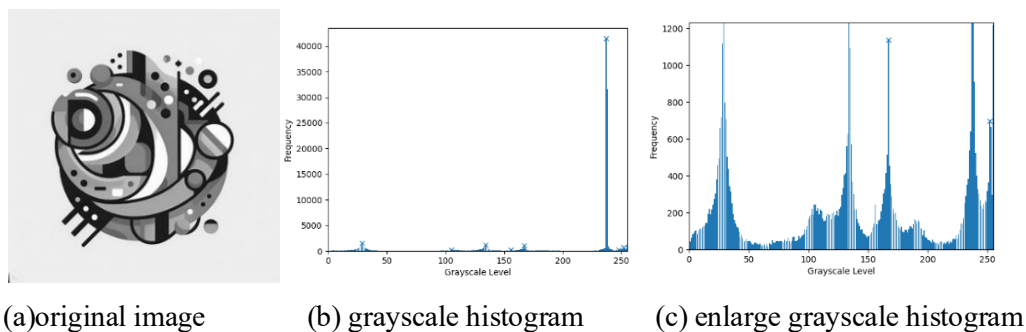


Figure 2. Preprocessing

Figure 2(a) is the input original image. The gray histogram obtained from Figure 1 is shown in Figure 2(b). In order to see the small peaks clearly, Figure 2(c) is obtained by controlling the maximum value of the ordinate, which is the enlarged gray histogram, which is convenient for subsequent operations.

Mark the peaks in the histogram with 'X'. From the histogram where the Figure 2(b) and (c) show. We can find that the grayscale value of the first peak is 29, and the grayscale value of the last peak is 237. Using these two values, we perform an initial binarization of the histogram, resulting in the following effect:

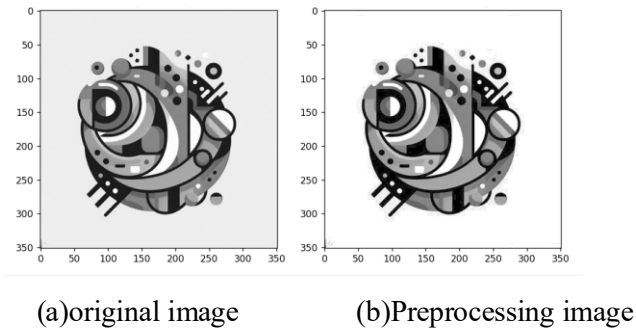


Figure 3. Comparison of original image and Preprocessing image

As shown in Figure 3, The background color is processed as white, and the dark lines of the image contour are also processed as black. This avoids the influence of the background color and light-colored lines on the next binarization operation. This step is beneficial for accurately restoring the edge contour of the image in subsequent steps.

4.1.2. Binarization Process. Binarization is performed using the previously mentioned improved local thresholding method based on the Otsu method. It is important to note that the weighting parameter "k" and the window size "r" in the method are manually set and require adjustment to achieve the targeted output image.

As Figure 4 shows, the image has been preliminarily binarized, and the effect is good, but there are some noise points.



Figure 4. Image after binarization

4.1.3. Noise Removal. Figure 5 shows that noise Removal makes the image cleaner and more details. Compared with Figure 4, it can be seen that the number of noise points is significantly reduced, the picture aesthetics is improved, and the edges and details have a better performance.



Figure 5. Image after noise removal

4.1.4. Comparison of Each Stage. As Figure 6(a) and (b) show, it can be clearly seen that the background color of the original image removed after the preprocessing step which reducing the influence of the background color on the next binarization operation. and the effect is very obvious in scenarios where the background color is large areas of white or black. After the binarization operation which Figure 6(c) shows, it can be seen that the structure of the pattern edge is complete, no details lost due to the binarization operation, and the gray level preserves the contrast of the color depth in the original image, and restores the original image details. In the last step of denoising, removing the noise from Figure 6(c). Figure 6(d) shows that the noise is obviously eliminated, the image is cleaner, and no details are lost due to the denoising operation.

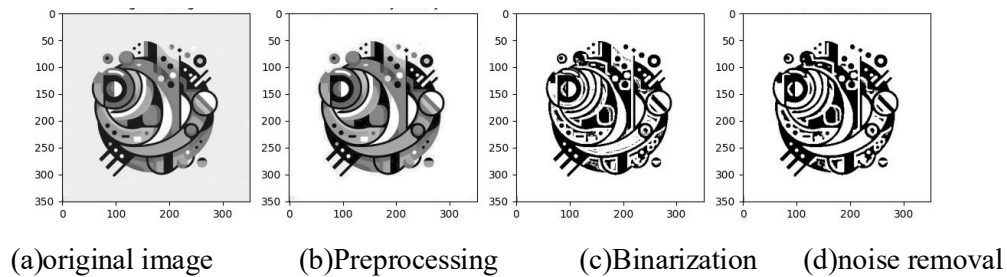


Figure 6. Comparison of Each Stage

4.2. Comparison with Traditional Methods

To demonstrate the effect of this method, we compare the Otsu method, the Sauvola method and the Adaptive method, where the Otsu method is a global threshold method and the other two are local threshold methods. From the comparison between Figure 7(a) and (b), we can find the disadvantage of the Otsu method as a global threshold method, which loses a lot of local details, but preserves the edges well. The Sauvola method, as a local threshold method, restores some local details, but also loses some edges which Figure 7(c) shows. The adaptive method considers the local mean, and restores the edge part well, but ignores the global image, resulting in only the edge and contour of the original image after binarization, without the dark areas but more noise. Figure 7(e) Compared with the previous traditional methods, it can be clearly seen that the method proposed in this paper restores the original image better in terms of edge and color, better retaining on shade of color, higher completion degree, and does not need secondary modification.

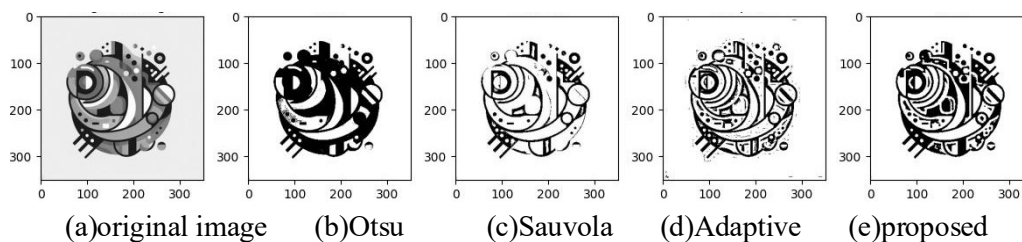


Figure 7. Comparison between different methods

4.3. Comparison with Other Samples

In order to demonstrate the differences between various methods when dealing with different types of images, the selected sample images considered the diversity of color and structure. The examples include images with rich colors and complex structures, as well as minimalist-style logos. This approach allows us to highlight the effectiveness of the binarization method proposed in this paper from multiple perspectives.

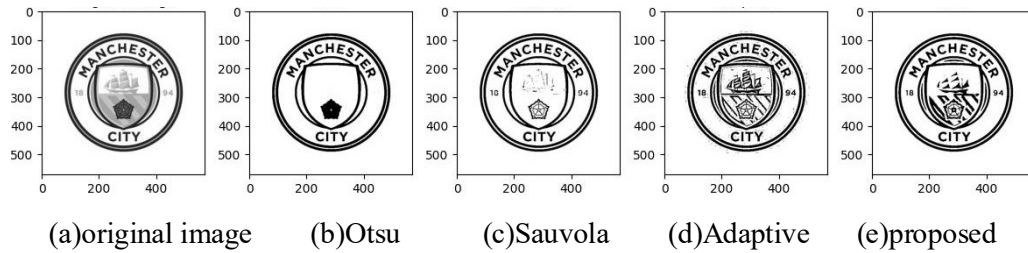


Figure 8. Sample 1

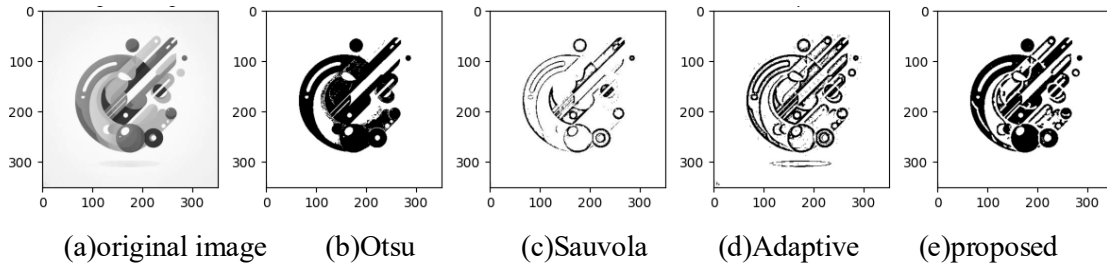


Figure 9. Sample 2

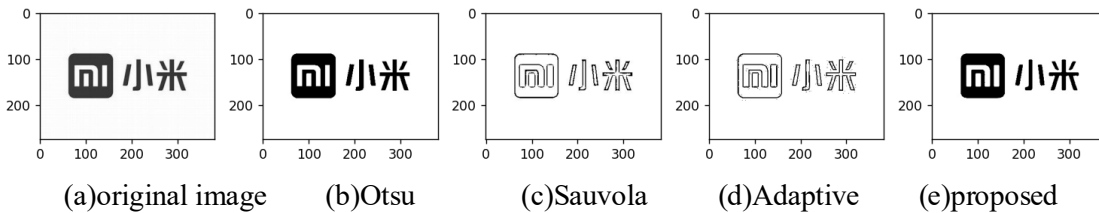


Figure 10. Sample 3

In the above comparison, we can analyze according to the type of image. In the images with complex structure and color, such as Figure 8 and Figure 9, the local threshold method has a better effect, and restores more details than the global threshold method. In Figure 9, it can also be seen that the method proposed in this paper has a better binarization effect when facing images with background color. In Figure 10, which is a simple image, the global threshold method has excellent binarization effect, while the local threshold method tends to depict the contour without color. From the three groups of comparisons, it can be seen that the method proposed in this paper combines the advantages of the local threshold method and the global threshold method in terms of color restoration and contour completeness, and has a wide range of applications.

5. Conclusion

This paper presents a binarization scheme optimized for small images, particularly logos, which offers several advantages including a cleaner image output, enhanced detail preservation, and improved visual effects. This scheme effectively amalgamates the benefits of both global and local threshold methods. It surpasses the global threshold approach in detail retention and outperforms the local threshold method

in noise reduction and image consistency. Its versatility allows for broad applicability without necessitating a choice between global or local methodologies.

To further refine this binarization scheme, a focus on evaluation metrics is recommended. Traditional measures like Hamming distance and image similarity are less effective in assessing the binarization quality of small images such as logos. A novel evaluation method could provide a more accurate reflection of the results. Additionally, adapting the parameters 'k' and 'r' within the binarization process based on these evaluations could enhance the scheme's intelligence and usability, making it more user-friendly and efficient in various applications.

References

- [1] K. Saddami, K. Munadi, S. Muchallil and F. Amia, "Improved Thresholding Method for Enhancing Jawi Binarization Performance," 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017, pp. 1108-1113, doi: 10.1109/ICDAR.2017.183.
- [2] Jihong Liu and Chengyuan Wang, "An algorithm for image binarization based on adaptive threshold," 2009 Chinese Control and Decision Conference, Guilin, 2009, pp. 3958-3962, doi: 10.1109/CCDC.2009.5191959.
- [3] D. Gaceb, F. Lebourgeois and J. Duong, "Adaptative Smart-Binarization Method: For Images of Business Documents," 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 2013, pp. 118-122, doi: 10.1109/ICDAR.2013.31.
- [4] B. Vishnudharan and K. Anusudha, "Reclamation of information from degraded and damaged document images by image binarization method," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2017, pp. 1-4, doi: 10.1109/ICIIECS.2017.8276100.
- [5] C. Wolf, J. . -M. Jolion and F. Chassaing, "Text localization, enhancement and binarization in multimedia documents," 2002 International Conference on Pattern Recognition, Quebec City, QC, Canada, 2002, pp. 1037-1040 vol.2, doi: 10.1109/ICPR.2002.1048482.
- [6] N. Ezaki, M. Bulacu and L. Schomaker, "Text detection from natural scene images: towards a system for visually impaired persons," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., Cambridge, UK, 2004, pp. 683-686 Vol.2, doi: 10.1109/ICPR.2004.1334351.
- [7] D. Asatryan, M. Haroutunian, G. Sazhumyan, A. Kupriyanov, R. Paringer and D. Kirsh, "Comparative Quality Analysis of Image Global Binarization Procedures," 2023 IX International Conference on Information Technology and Nanotechnology (ITNT), Samara, Russian Federation, 2023, pp. 1-5, doi: 10.1109/ITNT57377.2023.10138953.
- [8] Jun Li, Peng Xiao and Jian-Ping Li, "Improved infrared image binarization by using a combination of global method and local method," The 2010 International Conference on Apperceiving Computing and Intelligence Analysis Proceeding, Chengdu, 2010, pp. 290-293, doi: 10.1109/ICACIA.2010.5709903.
- [9] R. Hedjam, H. Z. Nafchi, M. Kalacska and M. Cheriet, "Influence of Color-to-Gray Conversion on the Performance of Document Image Binarization: Toward a Novel Optimization Problem," in IEEE Transactions on Image Processing, vol. 24, no. 11, pp. 3637-3651, Nov. 2015, doi: 10.1109/TIP.2015.2442923.
- [10] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern Recognition, Volume 33, Issue 2, 2000, Pages 225-236, ISSN 0031-3203, [https://doi.org/10.1016/S0031-3203\(99\)00055-2](https://doi.org/10.1016/S0031-3203(99)00055-2).
- [11] Bilal Bataneh, Siti Norul Huda Sheikh Abdullah, Khairuddin Omar, An adaptive local binarization method for document images based on a novel thresholding method and dynamic windows, Pattern Recognition Letters, Volume 32, Issue 14, 2011, Pages 1805-1813, ISSN 0167-8655, <https://doi.org/10.1016/j.patrec.2011.08.001>.

- [12] B. Gatos, I. Pratikakis, S.J. Perantonis, Adaptive degraded document image binarization, Pattern Recognition, Volume 39, Issue 3, 2006, Pages 317-327, ISSN 0031-3203, <https://doi.org/10.1016/j.patcog.2005.09.010>.