

# **Binarization of Handwritten Minutes of the Early Synod Meetings of Reformed Churches in South-Africa Literature Study**

ITRI 671

Zander Labuschagne 23585137

B.Sc.(Hons) Computer Science & Information Systems 2017

Dissertation submitted in partial fulfillment of the requirements  
for the degree B.Sc.(Hons) in Computer Science and Information  
Systems at the Potchefstroom Campus of the North-West University

Supervisors: Prof. G.R. Drevin & Mev. S. Campher

May 2017

# Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>                      | <b>1</b>  |
| <b>2</b> | <b>Pre-processing Techniques</b>         | <b>2</b>  |
| <b>3</b> | <b>Binarization Algorithms</b>           | <b>5</b>  |
| <b>4</b> | <b>Combining Binarization Algorithms</b> | <b>12</b> |
| <b>5</b> | <b>Post-processing Techniques</b>        | <b>14</b> |
| <b>6</b> | <b>Methods of Evaluation</b>             | <b>15</b> |
| <b>7</b> | <b>Conclusion</b>                        | <b>17</b> |
|          | <b>References</b>                        | <b>21</b> |

## List of Figures

|   |  |   |
|---|--|---|
| 1 | De-noising example as explained in [4] by Chen & Wang. (a) Original document image. (b) de-noised image . . . . .                                  | 3 |
| 2 | Binarization result by using Rosenfeld's method for calculating a threshold value [4]. (a) De-noised document image. (b) binarized image . . . . . | 6 |
| 3 | Underlining fountain pen stroke disconnected due to fast movement when writing. .  | 7 |
| 4 | Letters with broken ink strokes. . . . .   | 7 |

## 1 Introduction

This literature study forms part of the B.Sc. Honours project titled as binarization of handwritten minutes of the early synod meetings of Reformed Churches in South-Africa. All literature researched and reviewed will be discussed in this document. The aim of this study is to propose a binarization algorithm that bests differentiate the foreground (text, tables and figures) from the background (paper and stains) in historical handwritten documentation, specifically on the dataset which is the minutes of the early synod meetings of the Reformed Churches in South-Africa.

A considerable number of binarization algorithms exist and they all have their advantages and disadvantages; some fit better for certain circumstances or for a certain type of image. There exist no binarization algorithm yet that suits all types of images [11], but one can improve the result by combining algorithms [16], by adding pre-processing and post-processing techniques to the binarization algorithm such as noise removal [1], [11], or by careful experimentation and analysis on a very specific dataset in order to train the algorithm accordingly. One of the key objectives in binarization is thresholding, which is the calculation algorithm of the middle or “split” value used to discriminate the foreground from the background in the image. The algorithm would mark the pixels in a binary fashion and divide them into two categories, ON denoting the set of pixels representing the foreground or OFF denoting the set of pixels representing the background [17].

This document starts with a literature study on pre-processing techniques used to prepare the document for optimal binarization. Pre-processing techniques is then followed by binarization algorithms and the combination thereof which is followed by post-processing techniques used to enhance and refine the result. Finally the literature study will conclude with evaluation techniques used to determine the successfulness of this project.

## **2 Pre-processing Techniques**

Some pre-processing techniques such as cropping, deskewing and grey scale transformations which are used to prepare the image for binarization in order to improve binarization efficiency will be discussed in this section. Other pre-processing techniques includes low pass filters which are good for noise removal, high pass filters which are good for contrast enhancement in order to better differentiate between foreground pixels and background pixels and gamma correction algorithms which are good for unevenly distributed intensity values [20].

### **2.1 Geometric Corrections**

The geometric corrections that'll be necessary to prepare the document image for effective binarization will mostly consist of cropping and a de-skewing process. Cropping will be necessary to exclude some unnecessary parts of the image which can be found surrounding the information relevant to the document image. This will result in increased binarization performance since the document image would be smaller, this will also result in a lower probability of error or inaccurate binarization since there would be less defective area for the binarization algorithm to take in account. The de-skewing process aims to correct the shape of the document image to a more rectangular shape, it would be perfect if the horizontal margins and the vertical margins are perpendicular.

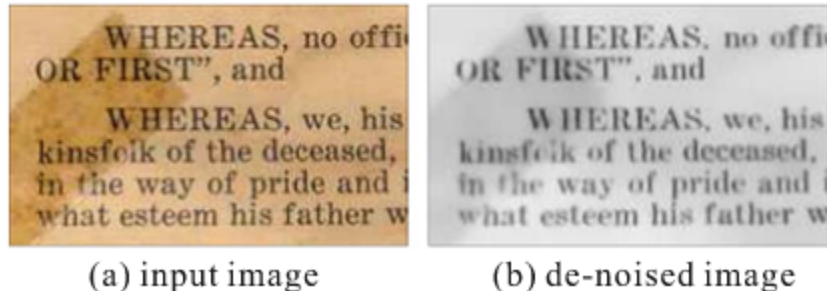
### **2.2 Grey-scale Transformation**

To simplify the binarization, the document images will undergo a grey-scale transformation. This transformation takes in a colour image and map the three red, green and blue colour scales to a single grey colour scale. It is uncertain whether the grey-scale transformation will be implemented before or after the geometric correction phase, but this will be experimented with to see whether the geometric corrections provide better results with red, green and blue colour scales or with grey colour scales.

## 2.3 Noise Removal

De-noising is the process of avergaing the pixel intensities so one can hide the outlier pixel intensities which is seen as noise, thus some detail in the image may be lost [3]. A non-local noise removal scheme that has proved itself can be implemented in the pre-processing phase of document binarization [4]. A noisy image can be defined as *equation 1* where  $v(i)$  is the observed image containing noise,  $u(i)$  the true image containing no noise and  $n(i)$  the noise component at pixel  $i$  [3]. *Figure 1* shows an example of what the result might look like if the de-noising objectives forms part of the pre-processing phase. One can see some loss in contrast in the de-noised image which could cause a decrease in accuracy during binarization but this can be dealt with since the detail is not completely lost.

$$v(i) = u(i) + n(i) \quad (1)$$



*Figure 1: De-noising example as explained in [4] by Chen & Wang. (a) Original document image. (b) de-noised image*

Low pass filters that are applied to the pre-processing phase proves to provide very good binarization results such as the Wiener, median, butterworth, ideal low pass, and Gaussian filters. These pre-processing filters can reduce effects such as non-uniform distributed intensities(shadows), wrinkles caused by humidity, poor contrast between foreground and background, broken characters in thin ink strokes, bleedthrough and dark dirt spots on the document itself [16]. Gatos's

binarization algorithm also incorporates noise removal into its pre-processing phase but will be discussed in more detail in *section 3.5*.

Another approach to noise removal is by using the Discrete Cosine Transform prior to binarization, followed by binarizing the transformed image rather than the image in original form, Arnia et al. [2] shows that this pre-processing techniques may be a good alternative to traditional noise removal algorithms. The Discrete Cosine Transform is defined in *equation 2*. Where  $f(x, y)$  denotes the original image, and  $F(\hat{x}, \hat{y})$  the Discrete Cosine Transform coefficient and  $C(\hat{x})$  the gain control which is defined in *equation 3* [2].

$$F_c(\hat{x}, \hat{y}) = \frac{2}{N} C(\hat{x}) C(\hat{y}) \sum_x \sum_y f(x, y) \cdot \cos\left(\frac{(2x+1)\hat{x}\pi}{2X}\right) \cdot \cos\left(\frac{(y+1)\hat{y}\pi}{2Y}\right) \quad (2)$$

$$C(\hat{x}) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } \hat{x} = 0 \\ 1, & \text{if } \hat{x} \neq 0 \end{cases} \quad (3)$$

The Inverse Discrete Cosine Transform is defined in *equation 4* which results in an image with a limited frequency band. Where  $\hat{f}_c(x, y)$  is the spatial image reproduced from the Discrete Cosine Transformation  $\hat{F}_c(\hat{x}, \hat{y})$ , in both the forward and backward transformations the  $X$  and  $Y$  denotes the width and height of the image respectively [2].

$$\hat{f}_c(x, y) = \frac{2}{N} \sum_{\hat{x}} \sum_{\hat{y}} C(\hat{x}) C(\hat{y}) \hat{F}(\hat{x}, \hat{y}) \cdot \cos\left(\frac{(2\hat{x}+1)x\pi}{2X}\right) \cdot \cos\left(\frac{(\hat{y}+1)y\pi}{2Y}\right) \quad (4)$$

As demonstrated by Ntogas and Ventzas [16] pre-processing does not improve the results for all binarization algorithms but provide some aid to some of the algorithms, which is why the

binarization algorithms will be chosen carefully and experimentation with combining them will also be investigated thoroughly while implementing pre-processing techniques.

### **3 Binarization Algorithms**

After the document image has been pre-processed to be in a geometric shape as close to a perfect document as possible, it will be binarized to differentiate the foreground, meaning text, from the background, meaning the paper. The algorithms used to binarize an image will be inspected and discussed in this section.

Binarization in image processing is the process of discriminating the foreground from the background in an image by changing the foreground pixels or text to black and the background or paper pixels to white [16]. Binarization is also considered by many computer scientists as a critical step in the processing of images because it has the greatest impact on the quality grade of all the processing that follows such as document analysis [9], thus a good binarization result would greatly improve the results of page-segmentation, optical character recognition and any other subsequent processing [1], while a bad binarization result will cause any following processing techniques to yield inadequate results [8]. A good binarization algorithm is one that best preserves the foreground and omit as much background as possible.

#### **3.1 Local vs. Global Thresholding**

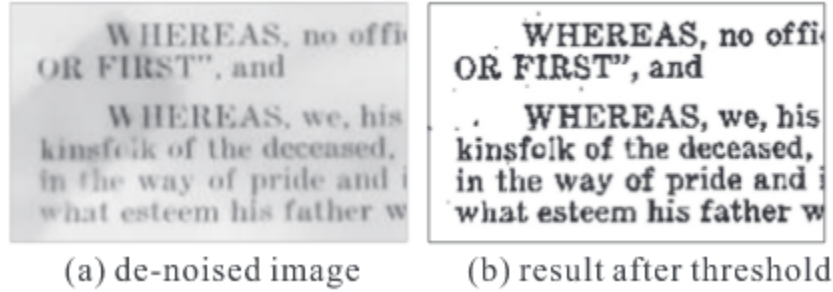
Thresholding is the process of determining a value that will divide the foreground and background, making this calculation a crucial objective to binarization. Global thresholding calculates a value by using the whole document image pixels, this method does not generate as much background noise but struggle to keep connectivity in thin stroke lines. Local adaptive binarization uses only a local window of surrounding pixels to calculate a new threshold value for each pixel which is adapted specifically for that pixel. this method is better for document images with non-uniform distributed intensity values and does not generate as much noise [4].



### 3.2 Rosenfeld's Threshold Calculation

Rosenfeld's method [4] for calculating the threshold value  $T$  analyzes the concavities of the histogram compared with its convex hull as seen in the log sigmoid *equation 5* [4] where  $I(x, y)$  denotes the original image,  $\delta$  the average difference in the white and black pixels,  $c_1$  and  $c_2$  the constant parameters with values  $c_1 = 0.5$  and  $c_2 = 0.8$  and  $w$  a weighting parameter with value  $w = 0.6$  as suggested by Chen & Wang [4]. *Figure 2* shows the result of the binarization by using Rosenfeld's method for calculating a threshold value. This method shows positive results for document images with lack of connectivity among characters as one would find in our document image dataset containing many areas where thin lines of ink strokes are disconnected due to the rapid movements of fountain pens as illustrated in *figures 3* and *4*.

$$T(I(x, y)) = w\delta\left(\frac{(1 - c_2)}{1 + \exp\left(\frac{-4I(x,y)}{b(1-c_1)} + \frac{2(1+c_1)}{(1-c_1)}\right)} + c_2\right) \quad (5)$$



*Figure 2: Binarization result by using Rosenfeld's method for calculating a threshold value [4]. (a) De-noised document image. (b) binarized image*

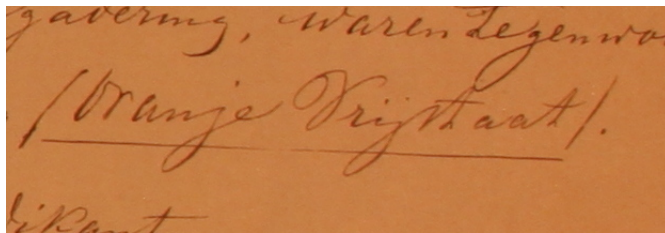


Figure 3: Underlining fountain pen stroke disconnected due to fast movement when writing.

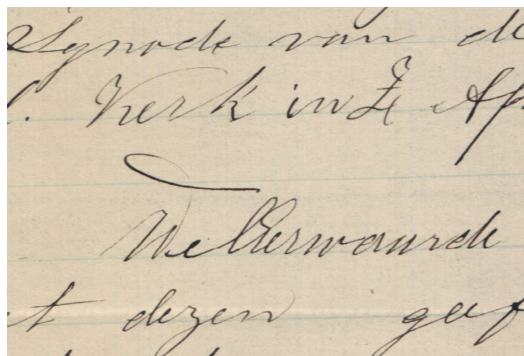


Figure 4: Letters with broken ink strokes.

### 3.3 Niblack Algorithm

The Niblack algorithm [13] calculates a local threshold by using the mean and the standard deviation of all the pixels in the local neighbourhood window. This method is good at classifying text as foreground but produces background noise. Equation 6 shows the calculation of the threshold value for the Niblack binarization algorithm where  $m(I(x, y))$  denotes the mean of the pixel intensity values of all the pixels inside of the local window which contains the pixel  $(x, y)$  in image  $I$ ,  $k$  a constant value of  $-0.2$  as suggested by Khurshid et al. [10] and  $S(I(x, y))$  the standard deviation of the pixel intensities of the complete image [10].

$$T(I(x, y)) = m(I(x, y)) + k \cdot S(I(x, y)) \quad (6)$$

The mean of the local window is calculated in *equation 7* where  $\text{floor}(x)$  rounds the value of  $x$  down to the closest integer less than  $x$ , and  $X$  denotes the local window size in the horizontal dimension and  $Y$  in the vertical dimension [10].

$$m(I(x, y)) = \sum_{i=\text{floor}(\frac{-Y}{2})}^{i<\text{floor}(\frac{Y}{2})} \sum_{j=\text{floor}(\frac{-X}{2})}^{j<\text{floor}(\frac{X}{2})} I(x + j, y + i) \quad (7)$$

The standard of the entire image is calculated in *equation 8* where  $N$  and  $M$  denotes the size of the entire image in the horizontal and vertical dimensions respectively and  $M(I(x, y))$  the mean of the pixel intensities of the entire image [10].

$$S(I(x, y)) = \sqrt{\frac{1}{NM} \sum_{i=1}^{i \leq N} \sum_{j=1}^{j \leq M} (I(i, j) - M(I(x, y)))^2} \quad (8)$$

The mean of the pixel intensities of the entire image is calculated in *equation 9* where  $X$  and  $Y$  denotes the size of the entire image in the horizontal and vertical dimensions respectively.

$$M(I(x, y)) = \sum_{i=1}^{i \leq X} \sum_{j=1}^{j \leq Y} I(i, y) \quad (9)$$

### 3.4 Sauvola's Algorithm

Sauvola's algorithm [19] calculates the threshold value by using the standard deviation of the image's dynamic range. This method is an improvement over the Niblack algorithm in high contrast images so it provides inadequate results when the contrast between the foreground and background decreases. *Equation 10* shows the calculation of the threshold value for Sauvola's binarization algorithm where  $k$  and  $R$  are constants set to 0.5 and 128, respectively, as suggested by Khurshid et al. [10],  $m(I(x, y))$  calculates the mean of the pixel intensities in the local window as seen in

equation 7, and  $S(I(x, y))$  the standard deviation of all the pixels in the entire image as seen in equation 8 [10].

$$T(I(x, y)) = m(I(x, y)) \cdot (1 - k \cdot (1 - \frac{S(I(x, y))}{R})) \quad (10)$$

### 3.5 Gatos' Algorithm

Gatos combined existing algorithms [7], his method uses a Wiener filter in a pre-processing phase to eliminate noise and to enhance the contrast of the image before binarization takes place. Equation 11 shows how the image is pre-processed.  $I_{pre}(x, y)$  denotes the image after pre-processing and  $I(x, y)$  the original gray scale image.  $\mu$  denotes the local mean,  $\sigma^2$  the local variance and  $v^2$  the average of all estimated variances of each pixel in the local window [18].

$$I_{pre}(x, y) = \mu + \frac{(\sigma^2 - v^2)(I(x, y) - \mu)}{\sigma^2} \quad (11)$$

Sauvola's algorithm is adopted once the pre-processing is done and can be seen in equation 10. Gatos also makes use of three post-processing techniques to further enhance the result such as the preservation of stroke connectivity to name one [18].

### 3.6 Wolf's Algorithm

Wolf's algorithm [22] is an improvement over Sauvola's and Niblack's algorithm in most cases but performance decreases when there is non-uniform distributed intensity values in the image which is commonly found in document images sampled via a camera instead of a scanner. The drawbacks of this algorithm is caused by the inclusion of all the pixels in the image into the algorithm just as in a global thresholding method. The calculation for the threshold value for Wolf's algorithm can be seen in equation 12 where  $M$  denotes the minimum pixel intensity of the image,  $k$  a constant value

of 0.5 as suggested by Khurshid [10],  $R$  the maximum standard deviation calculated throughout the entire image (the standard deviation is calculated for each local window in Wolf's algorithm), and  $m(I(x, y))$  the mean of the pixel intensities in the local window as seen in *equation 7* [10].

$$T(I(x, y)) = (1 - k) \cdot m(I(x, y)) + k \cdot M + k \cdot \frac{S(I(x, y))}{R} (m(I(x, y)) - M) \quad (12)$$

### 3.7 Feng's Algorithm

Feng's algorithm [6] utilises two local sliding windows, the one is contained in the other. The local mean denoted by  $m(I(x, y))$ , standard deviation denoted by  $s(I(x, y))$  and the minimum grey level is calc denoted by  $M$  is calculated in the smaller windows while another dynamic range standard deviation denoted by  $R_s$  is calculated in the larger window. Feng's algorithm is an attempt to address the global drawback in Wolf's algorithm and the calculation for the threshold value can be seen in *equation 13* where  $\alpha_2 = k_1(s(I(x, y))/R_s)^\lambda$  and  $\alpha_3 = k_2(s(I(x, y))/R_s)^\lambda$ . Khurshid et al. recommends to set  $\lambda = 2$  based on their experience, while keeping  $\alpha_1$ ,  $k_1$  and  $k_2$  in the ranges of 0.1 - 0.2, 0.15 - 0.25 and 0.01 - 0.05 respectively. [10].

$$T(I(x, y)) = (1 - \alpha_1) \cdot m(I(x, y)) + \alpha_2 \cdot \frac{s(I(x, y))}{R_s} \cdot (m(I(x, y)) - M) + \alpha_3 \cdot M \quad (13)$$

### 3.8 Khurshid, Siddiqi, Faure and Vincent's Approach

This method focuses on ancient, degraded and noisy document images and is calculated by *equation 14* where  $k$  denotes the Niblack factor,  $m(I(x, y))$  the mean pixel intensity of the local window,  $I(i, j)$  the pixel intensity of pixel  $(i, j)$ ,  $N$  and  $P$  the size of the horizontal and vertical dimensions respectively.

$$T(I(x, y)) = m(I(x, y)) + k \sqrt{\frac{\sum_{i=1}^{i \leq P} \sum_{j=1}^{j \leq N} I(i, j)^2 - m(I(x, y))^2}{NP}} \quad (14)$$

This method was tested on the Bibliothèque Interuniversitaire de Médecine, Paris, and the Institute de recherche et d'histoire des textes and provided improved results in comparrison with Niblack, Sauvola's, Wolf's and Feng's algorithms when evaluated subjectively with a visual examination [10].

### 3.9 A Laplacian Energy for Document Binarization

This method of binarization uses three concepts to achieve a binzarized image result. It firstly utilizes the Laplacian of the image intensity to achieve illumination invariance. The Laplacian is very good at seperating the dark from the light because it assocaiates the peaks with negative values and the valleys with positive values, which can be used to find a good threshold value for the binarization. Secondly it finds an optimal solution to a global fitness function which keeps the result consistant. Third it utilizies Canny edge detection of which the results are used to coinsize the broken lines of text.[8].

The magnitude of the Laplacian operator increases in the areas of high frequencies and approach zero when the intensity values of the image are of low frequency, thus uniform areas such as the background approaches zero while areas with high rates of change such as text increases the magnitude of the Laplacian operator[8].

This method takes a greyscale image  $I$  with dimensions  $m \times n$ , thus  $I_{i,j} = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$  where 1 denotes white and 0 denotes black, and produces a binarized image  $B$  where  $B_{i,j} \in \{0, 1\}$ . This method is evaluated on four performance measures, F-measure, peak signal to noise ratio,

negative metric rate and misclassification metric penalty, the higher the F-measure and peak signal to noise ratio the better the result and the lower the negative metric value and misclassification metric penalty the weaker the result. This method was tested on the DIBCO-09 and H-DIBCO ground truth document image set and scored very high in comparison with the other methods[8].

This method of binarization should provide good results but only in document images that does not contain pictures, diagrams or any other large areas which does not form part of the background, it was also noticed that it does not perform as well on handwritten documents as it does on printed documents but still significant enough.

## 4 Combining Binarization Algorithms

In some cases one binarization algorithm would not suffice for the whole set of a specific type of documentation because outliers are to be expected. One can improve the likelihood of a successful binarization result by using multiple binarization algorithms for a more accurate classification of foreground and background. The combination of a set of binarization algorithms and their increase or decrease in performance will be discussed in this section.

In the proposed framework of Su, Lu and Tan [21] a feature extraction is implemented before the combined binarization step. This feature extraction consists of the contrast and the intensity values of each pixel as shown in *equation 15* [21].

$$Con(x, y) = \frac{f_{max}(x, y) - I(x, y)}{f_{max}(x, y) + \epsilon} \quad (15)$$

Where  $f_{max}(x, y)$  denotes the maximum pixel intensities within the local neighborhood,  $I(x, y)$  the intensity of the pixel  $(x, y)$ ,  $Con(x, y)$  the contrast value of the pixel  $(x, y)$  and  $\epsilon$  any positive real value to prevent division by zero. Thus the image is projected from the two-dimensional image

space  $f(x, y)$  into the two-dimensional feature space  $F(Con, I)$  where  $Con$  represents the contrast of the pixel  $(x, y)$  and  $I$  the intensity of the pixel  $(x, y)$  [21].

The image is then binarized with two existing binarization algorithms. If both algorithms classify the pixel as foreground the pixel is then classified as part of the foreground and same goes for the background classification as seen in *equation 16*. Where  $P(x)$  denotes the uncertain pixel to be classified,  $Con(x)$  the contrast value of pixel  $x$ ,  $I(x)$  the intensity value of pixel  $x$ ,  $Con_F$  the mean contrast of the foreground pixels,  $Con_B$  the mean contrast of the background pixels,  $I_F$  the mean intensity of the foreground pixels and  $I_B$  the mean intensity of the background pixels [21].

$$P(x) = \begin{cases} foreground, & \frac{Con(x)}{Con_F} > \frac{Con_B}{Con(x)} \parallel \frac{I_F}{I(x)} > \frac{I(x)}{I_B} \\ background, & otherwise \end{cases} \quad (16)$$

But when the algorithms don't agree to the same classification, the pixel is then marked as uncertain. All uncertain pixels are then examined under a local neighbourhood window, the uncertain pixels are classified depending on their distance from the local background and foreground pixels inside the window. This process repeats until there are no more unclassified pixels. During the following iteration a new binarization algorithm is used in combination with the existing results, thus as the iterations increases the number of binarization algorithms used in combination increases [21].

This method was tested on the DIBCO-09 and H-DIBCO datasets and provided excellent results [21] on both printed and handwritten document images. This method focusses on degraded document images which is exactly the type of images this project focusses on.



## **5 Post-processing Techniques**

After the document image has been binarized, containing only black and white pixels, post-processing can be done in order to improve on the result where necessary. For document analysis to succeed a clean document image is required meaning no noise or speckles on background areas. Segmentation for example uses connected lines in order to segment, and so do optical character recognition with connected strokes in characters, if these lines and strokes are interrupted or discontinuous it might throw any following processing off course providing inaccurate results. All classes of document images have their types of noise found commonly in their class. These types of noises can be rule-lines, bleed through, stray marks, clutter or bipolar noise [1]. Various post-processing techniques used to improve the image quality of a binarized image will be discussed in this section.

### **5.1 Bipolar Noise Removal**

Bipolar noise is very common in document images and is reduced by using either a median filter or morphological operator with a  $3 \times 3$  or smaller window size. Other methods do exist but these are more common. Bipolar noise does not depend on any of the properties of the image such as size or content which is probably why it is so common in various types of images, not just in document images [1].

### **5.2 Clutter Noise Removal**

Clutter noise may appear during the scanning process because of the scanning equipment's generic thresholding. This may be avoided by using some kind of image acquisition equipment that does not utilize thresholding, such as a camera, so one can do the thresholding more accurately in the binarization algorithm. However a camera introduces a new set of problems for the pre-processing for binarization [1] such as shadows or variable illumination on the document image.

### **5.3 Dilation and Erosion**

Dilation is the process of enhancing the darker pixels, in this case the foreground or text, this results in a more clarified text in the document image. Erosion is the process of reducing the darker pixels, this would aid in the elimination of smear marks or other noise considered to be part of the foreground. [16].

## **6 Methods of Evaluation**

The different options available to evaluate the results of this study will be discussed in this section. Evaluation methods determines how successful the binarization method accomplishes its goal which is in this case to differentiate the background from the foreground. This result of this study prepares the handwritten document images for further processing to finally produce a digital counterpart. The end result would be a binarized document image with additional post-processing to prepare the document image as far as possible for optical character recognition.

### **6.1 Visual Inspection**

A visual inspectoin is a subjective evaluation where humans inspect the binarized document image using only their eyes and subjective judgement [15]. Usually this method of evaluation should be avoided because it is not as accurate as the other methods, but this method still has its place in some applications because after all it will be a human ending up appreciating the results. Though in this case it should be avoided since the resulting binarized image will be used for further processing such as segmentation, optical character recognition or word spotting which is done by a machine, not a human.

### **6.2 Optical Character Recognition Success Rate**

A very effective method to evaluate the results of this study is to feed the binarized document images to an optical charecter recognition system and test the success rate of this system and

compare it with other binarization algorithms's success rates [4]. This project specifically focusses on handwritten text which makes it very difficult to use optical character recognition systems as an evaluation method since optical character recognition systems mainly focus on printed text and not handwritten text. There are some optical character recognition systems that are developed specifically for handwritten text but are not as accurate [15].

### 6.3 Peak Signal to Noise Ratio

The *PSNR* is very commonly used in the assessment of image or signal quality but does not have adequate correlation with human vision for subjective assessment. The peak signal to noise ratio is calculated by using *equation 17* where  $MAX_i$  the maximum possible pixel intensity in the image,  $n$  is the size of the horizontal dimension of the image and  $m$  the size of the vertical dimension of the image [12].

$$PSNR = 20 \log_{10}(MAX_i) - 10 \log_{10}\left(\frac{1}{nm} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (f(i, j) - f'(i, j))^2\right) \quad (17)$$

### 6.4 Precision & Recall

Precision is a performance measure of how consistent the algorithm is with its classifications, whereas recall is a performance measure of how accurate the algorithm classifies the pixels. Precision is calculated by using *equation 18* and recall is calculated by using *equation 19*. *TP* denotes the *True Positive* pixels which are the pixels classified as foreground that is truly part of the foreground. *FP* denotes the *False Positive* pixels, these are the pixels that are classified as part of the foreground but are actually part of the background. *TN* denotes the *True Negative* pixels, it is the set of pixels that are classified as part of the background and which is truly part of the background. This leaves *FN* as the *False Negative* pixels which are the pixels classified as part of the background but is actually part of the foreground [5]. A disadvantage when using precision and recall to evaluate binarization is that groundtruth is required, because one needs a set of truly

correctly binarized document images to compare with the results obtained during the research but groundtruth aren't always available [15].

$$precision = \frac{TP}{TP + FN} \quad (18)$$

$$recall = \frac{TP}{TP + FN} \quad (19)$$

## 6.5 F-Measure

The F-Measure evaluation is a widely used method in binarization to determine the effectiveness of a binarization algorithm, it is calculated by using *equation 20*. Since it uses the recall and precision measures it also requires a set of groundtruth images to be compared with [14].

$$F_m = \frac{2 \cdot recall \cdot precision}{recall + precision} \quad (20)$$

## 7 Conclusion

We looked at pre-processing techniques that can be done to improve the efficiency of binarization algorithms, some of the pre-processing techniques will definitely be implemented such as the geometric corrections and most probably noise removal since it shows such an improvement in Gatos' binarization algorithm. Various binarization algorithms were looked at, some stood out performance-wise such as Gatos' algorithm, Khurshid's et al. approach, the Laplacian energy and the combination of different binarization algorithms, these will definitely be experimented with to find a good algorithm or combination thereof which produces the best results in the scope of this project. Other algorithms such as Feng's algorithm that does not look as promising as the

others will be of less priority and receive less attention but will not be overlooked. The binarization algorithms were followed by post-processing techniques that can be done on the binarized images in order to refine the results for better performance in procedures such as segmentation and optical character recognition. Finally methods for evaluation were discussed to choose appropriate evaluation method(s) to determine whether the results are successful or not. The most important evaluation methods that are considered are the precision and recall and optical character recognition success rate.

## References

- [1] Mudit Agrawal and David Doermann. Stroke-like pattern noise removal in binary document images. In *2011 International Conference on Document Analysis and Recognition (ICDAR)*, pages 17–21. IEEE, 2011.
- [2] Fitri Arnia, Khairul Munadi, Sayed Muchallil, et al. Improvement of binarization performance by applying DCT as pre-processing procedure. In *Communications, Control and Signal Processing (ISCCSP), 2014 6th International Symposium on*, pages 128–132. IEEE, 2014.
- [3] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 60–65. IEEE, 2005.
- [4] Yiping Chen and Liansheng Wang. Broken and degraded document images binarization. *Neurocomputing*, 2017.
- [5] Tom Fawcett. An introduction to ROC analysis. *Pattern recognition letters*, 27(8):861–874, 2006.
- [6] Meng-Ling Feng and Yap-Peng Tan. Contrast adaptive binarization of low quality document images. *IEICE Electronics Express*, 1(16):501–506, 2004.
- [7] Basilios Gatos, Ioannis Pratikakis, and Stavros J Perantonis. Adaptive degraded document image binarization. *Pattern recognition*, 39(3):317–327, 2006.
- [8] Nicholas R Howe. A Laplacian energy for document binarization. In *2011 International Conference on Document Analysis and Recognition (ICDAR)*, pages 6–10. IEEE, 2011.
- [9] Bency Jacob and SB Waykar. A survey on binarization of historical degraded documents. 2014.

- [10] Khurram Khurshid, Imran Siddiqi, Claudie Faure, and Nicole Vincent. Comparison of Niblack inspired binarization methods for ancient documents. In *IS&T/SPIE Electronic Imaging*, pages 72470U–72470U. International Society for Optics and Photonics, 2009.
- [11] Rafael Dueire Lins, P Gabriel de França, and Marcos Martins de Almeida. Binarizing complex scanned documents. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 56–60. IEEE, 2015.
- [12] Ning Liu and Guangtao Zhai. Free energy adjusted peak signal to noise ratio (FEA-PSNR) for image quality assessment. *Sensing and Imaging*, 18(1):11, 2017.
- [13] Wayne Niblack. *An introduction to digital image processing*. Strandberg Publishing Company, 1985.
- [14] Konstantinos Ntirogiannis, Basilios Gatos, and Ioannis Pratikakis. An objective evaluation methodology for document image binarization techniques. In *Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on*, pages 217–224. IEEE, 2008.
- [15] Konstantinos Ntirogiannis, Basilios Gatos, and Ioannis Pratikakis. Performance evaluation methodology for historical document image binarization. *IEEE Transactions on Image Processing*, 22(2):595–609, 2013.
- [16] Nikolaos Ntogas and Dimitrios Veintzas. A binarization algorithm for historical manuscripts. In *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering*, number 12. World Scientific and Engineering Academy and Society, 2008.
- [17] Lawrence O’Gorman and Rangachar Kasturi. *Document image analysis*, volume 39. IEEE Computer Society Press Los Alamitos, 1995.
- [18] Yu-Ting Pai, Yi-Fan Chang, and Shanq-Jang Ruan. Adaptive thresholding algorithm: Efficient computation technique based on intelligent block detection for degraded document images. *Pattern Recognition*, 43(9):3177–3187, 2010.

- [19] Jaakko Sauvola, Tapio Seppanen, Sami Haapakoski, and Matti Pietikainen. Adaptive document binarization. In *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*, volume 1, pages 147–152. IEEE, 1997.
- [20] Mahmoud Soua, Rostom Kachouri, and Mohamed Akil. Improved hybrid binarization based on Kmeans for heterogeneous document processing. In *Image and Signal Processing and Analysis (ISPA), 2015 9th International Symposium on*, pages 210–215. IEEE, 2015.
- [21] Bolan Su, Shijian Lu, and Chew Lim Tan. Combination of document image binarization techniques. In *2011 International Conference on Document Analysis and Recognition (ICDAR)*, pages 22–26. IEEE, 2011.
- [22] Christian Wolf and J-M Jolion. Extraction and recognition of artificial text in multimedia documents. *Pattern Analysis & Applications*, 6(4):309–326, 2004.