EFFECTIVE AND EFFICIENT BINARIZATION OF DEGRADED DOCUMENT IMAGES

A Dissertation
submitted to the Faculty of the
Graduate School of Arts and Sciences
of Georgetown University
in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy
in Computer Science

By

Jon Ivan Parker, M.S.

Washington, DC
February 1, 2016
EFFECTIVE AND EFFICIENT BINARIZATION OF DEGRADED DOCUMENT IMAGES

Jon Ivan Parker, M.S.

Thesis Advisor: Ophir Frieder, Ph.D.

ABSTRACT

Extracting information from images of documents is easier when the image is crisp, clear and devoid of noise. Consequently, an algorithm that reliably removes noise from imperfect document images and generates better images could clean input to other image processing algorithms thereby improving their outputs and/or enabling simpler techniques. The importance of this task is evident given the rate at which scanners, copier, and smart phones are producing document images.

This dissertation makes three contributions to this problem area. The first contribution is an unsupervised method for converting a document image to a strictly white and black image (i.e., cleaning a document image). This initial contribution is the result of examining the hypothesis that acceptable binarization parameters can be found with an automatic parameter search and was patented in US Patent #8,995,782 "System and Method for Enhancing the Legibility of Degraded Image". The second contribution is an improvement on the prior method that eliminates the need for a computationally expensive parameter search. A patent for this contribution has been allowed, but not yet issued, under patent application number 13/949,799 "System and Method for Enhancing the Legibility of Images". The last contribution of this dissertation is a method that manipulates multiple images of the same document that were each captured with a different mono-chromatic frequency of light to improve image binarization.
Dedicated to my wife and daughter

I love you both,
Jon I. Parker
# Table of Contents

Chapter 1 : Introduction........................................................................................................1

Chapter 2 : Background Information and Related Works ..................................................5

2.1 Early Methods ...............................................................................................................5

2.2 Image Operations .........................................................................................................8

2.3 Methods Based on Foreground-Background Separation and Background Estimation . . .13

2.4 Binarization Methods Based on Machine Learning ......................................................16

2.5 Binarization Methods Based on Function Minimization .............................................18

2.6 Unique Binarization Methods .....................................................................................20

2.7 The Document Image Binarization Contest (DIBCO) Series .....................................21

2.8 The MultiSpectral Text Extraction Contest (MS-TEx 2015).........................................23

2.9 Judging the Quality of a Binary Image .........................................................................23

Chapter 3 : Automatic Enhancement and Binarization of Degraded Document Images ......26

3.1 Methodology ...............................................................................................................26

3.2 Creating a Greyscale Image .........................................................................................28

3.3 Process 1: Isolating Locally Dark Pixels ......................................................................28

3.4 Process 2: Isolating Pixels Near an Edge ....................................................................29

3.5 Combining Results from Processes 1 and 2 ................................................................30
3.6 Parameter Selection........................................................................................................... 31
3.7 Experimental Results and Discussion ................................................................................. 33
3.8 The DIBCO 2011 Test Set ................................................................................................... 35
3.9 Conclusion............................................................................................................................ 37

Chapter 4 : Robust Binarization of Degraded Document Images Using Heuristics ............ 38

4.1 Methodology ....................................................................................................................... 39
4.2 Creating a Greyscale Image ............................................................................................... 40
4.3 Identifying Locally Dark Pixels ......................................................................................... 40
4.4 Identifying Pixels that are Near an Edge ......................................................................... 43
4.5 The Intersection and Snippet Sizes ................................................................................. 44
4.6 Cleaning the Image ............................................................................................................ 46
4.7 Method Summary .............................................................................................................. 49
4.8 Experimental Results ...................................................................................................... 49
4.9 DIBCO Results ................................................................................................................ 50
4.10 Examples from the Frieder Diaries ................................................................................. 54
4.11 Conclusion ........................................................................................................................ 60

Chapter 5 : Robust Multispectral Text Extraction on Historic Documents ....................... 61

5.1 Methodology ................................................................................................................... 62
5.2 Evaluation .......................................................................................................................... 66
LIST OF FIGURES

Figure 2.1: Applying Otsu's method to an image. .......................................................... 6
Figure 2.2: Applying Niblack's method to an image. ......................................................... 7
Figure 2.3: Gaussian blur. ................................................................................................. 9
Figure 2.4: Sharpening an image with an unsharp mask (USM). ........................................ 10
Figure 2.5: The effect of the erosion and dilation operations. ........................................... 13
Figure 2.6: Two images with uneven lighting. ................................................................. 15
Figure 2.7: A synthetic test image with two correct binarizations. .................................. 22
Figure 3.1: Pseudo code for the image enhancement method. ......................................... 28
Figure 3.2: Process used to isolate pixels that are "near an edge". .................................... 29
Figure 3.3: Intermediate results when isolating pixels that are near an edge. ................. 30
Figure 3.4: A binarization with two different winSize values. ............................................ 32
Figure 3.5: Sample result: Document M.5_193_67. .......................................................... 33
Figure 3.6: Sample results: Document M.5_193_25. ......................................................... 34
Figure 3.7: DIBCO 2011 HW3. .......................................................................................... 36
Figure 3.8: DIBCO 2011 HW2. .......................................................................................... 36
Figure 4.1: A high level view of the image enhancement method. ..................................... 40
Figure 4.2: Process that identifies locally dark pixels. ...................................................... 41
Figure 4.3: Typical input and output when identifying locally dark pixels. ...................... 42
Figure 4.4: Process that identifies pixels that are near an edge. ....................................... 43
Figure 4.5: An example of identifying pixels that are near an edge. .................................. 44
Figure 4.6: The input and important outputs of the proposed method. ......................... 45
Figure 4.7: The visual artifact caused by using a snippet size that is too small when computing
locally dark pixels. .................................................................................................................. 45

Figure 4.8: Example pixels arrangements that are changed (columns 1 and 2) and not changed
(column 3). ...................................................................................................................................... 47

Figure 4.9: This letter exhibits two white islands because the pixels at the center of the letter are
not identified as pixels that are "near an edge". .............................................................................. 48

Figure 4.10: The proposed methods outperforms its predecessor and never fails catastrophically
on any image from the 2011 DIBCO dataset. .................................................................................. 51

Figure 4.11: DIBCO image HW4 and its corresponding black and white image. ......................... 53

Figure 4.12: DIBCO image HW7 and its corresponding enhanced image. ................................. 53

Figure 4.13: DIBCO image PR6 and its corresponding enhanced image. ................................. 54

Figure 4.14: DIBCO image PR7 and its corresponding enhanced image. ................................. 54

Figure 4.15: Excerpt from the input and output of diary image M.5_192_61. ........................... 56

Figure 4.16: Excerpt from the input and output of diary image M.5_192_92. ........................... 57

Figure 4.17: Excerpt from the input and output of diary image M.5_193_95. ........................... 58

Figure 4.18: Excerpt from the input and output of diary image M.5_193_207. ........................... 59

Figure 5.1: Three of the eight images from the MS-TEx z76 image set. ................................. 64
LIST OF TABLES

Table 3-I: Select results from DIBCO 2011 ............................................................. 37
Table 4-I: Top DIBCO 2011 results and the proposed method ..................................... 51
Table 4-II: Top DIBCO 2013 results and the proposed method ..................................... 51
Table 5-I: Description of the eight images within an “image set” from the MS-TEx 2015
Competition............................................................................................................... 63
Table 5-II: Top MS-TEx 2015 results and the proposed method ..................................... 67
Chapter 1: Introduction

Extracting information from images of documents is easier when the image is crisp, clear and devoid of noise. Consequently, an algorithm that reliably removes noise from imperfect document images and generates better images could clean input to other image processing algorithms thereby improving their outputs and/or enabling simpler techniques. The importance of this task is evident given the rate at which scanners, copier, and smartphones are producing document images.

This dissertation begins by introducing background information and relevant prior work necessary for a complete discussion of document image binarization – i.e., generating crisper clearer images. The literature review begins by introducing some of the earliest binarization routines. Then, a host of general purpose image processing methods are introduced. Many of these general purpose image processing techniques are used within other image binarization methods. Therefore, these techniques are explained here for completeness. Next, the literature review enumerates many newer, more effective image binarization methods. The enumeration covers these methods in batches of related methods including a collection of methods that rely on foreground-background separation and background estimation, a collection of methods that use machine learning, and a collection of methods that define a cost (or likelihood) function and then minimizes (or minimizes) this function. The literature review concludes with a discussion of metrics used to judge the quality of a binarized image and some image collections compiled to enable algorithm evaluation.
After the background information and prior work is introduced, a novel image binarization algorithm is proposed in Chapter 3. This image binarization algorithm embodies two guiding principles. The first is that "writing" should be darker than "non-writing", and the second is that "writing" should generate a detectable edge. These principles are acted upon in completely separate branches of the algorithm that are governed by independent parameters. The lynchpin of this algorithm is a process that estimates good parameters for the two independent branches. This parameter setting process is a direct result of examining the hypothesis that acceptable binarization parameters can be found with an automatic parameter search. The resulting algorithm is different from other binarization techniques because it requires no human interaction to determine parameters nor does it use training data to set the parameters. The parameters are set automatically. Consequently, this approach is embarrassingly parallel because each input image is processed independently of all other input images. This contribution was patented in US Patent "System and Method for Enhancing the Legibility of Degraded Image" and published at the 12th International Conference on Document Analysis and Recognition (ICDAR) in early 2013 (Parker 2013).

The self-guided algorithm discussed in Chapter 3 has room for improvement because the parameter setting step is computationally taxing. The substantial computational cost is due to the parameters setting step picking the (likely) best output image from a collection of candidate images that are each fully computed using a different parameter setting. The 4th chapter of this dissertation presents a substantial revision to the algorithm from Chapter 3 conceived when examining the hypothesis that acceptable binarization output does not require a parameter search. The revised algorithm eliminates the need for the computationally taxing parameter setting step.
by more effectively identifying pixels that are both locally dark and near an edge. The resulting algorithm is both significantly faster than its predecessor and among the best performing image binarization algorithms known. This improvement was patented in US Patent "System and Method for Enhancing the Legibility of Images" and published at Document Recognition and Retrieval XXI in late 2013 (Parker 2014).

The final chapter addresses a slightly different image binarization problem. The goal is the same, but the input to the problem is different. The last chapter covers the case in which eight images of the same document are available. However, each of these images was captured with a different mono-chromatic wavelength of light. For example, one input image was captured using light with a 340 nm wavelength (ultraviolet), and another input image was captured using light with a 900 nm wavelength (infrared). This chapter introduces a method that relies on the single-input image binarization technique from Chapter 4 to address the multispectral input problem. The approach given has a mean performance that is statically indistinguishable from other top multispectral image binarization methods, including the winner of the MS-TEx 2015 competition, yet its variance is lower than other top methods. Moreover, the multispectral approach retains all benefits of not needing training data or human interaction for parameter setting because it uses simple rules to adapt to the multispectral input case.

In summary, the novel contributions of this dissertation are:

- This dissertation introduces an image binarization technique that uses a heuristic to automatically find acceptable parameters. This heuristic requires no training data and
no human interaction. Therefore, this image binarization technique is well-suited to address large image corpuses because it is embarrassingly parallel.

- This dissertation introduces an improved version of previous image binarization technique that eliminates the need for a parameter search. The improved method has better average performance as well as lower variance than other known image binarization methods. Thus, it more consistently provides better results than other document image binarization methods. Additionally, the improved method still requires no training data, no human interaction, and can be applied in an embarrassingly parallel fashion to tackle large image corpuses.

- This dissertation introduces a method to address image binarization where multispectral input images are available. The multispectral approach has a mean performance that is statically indistinguishable from other top multispectral image binarization methods, including the winner of the MS-TEx 2015 competition, yet its variance is lower than other top methods. The multispectral approach also retains the benefits of not needed training data or human interaction for parameter setting because it uses simple rules to adapt to the multispectral input case.
2.1 Early Methods

One of the earliest and most enduring image binarization methods is Otsu's method (1975). First published in 1975, this method finds a threshold for which all darker pixels are set to black and all lighter pixels are set to white. The computed threshold maximizes the between-class variance (which is the same as minimizing the within-class variance) of the pixels classified as black and the pixels classified as white. This threshold is easy to find from the input image's histogram of greyscale values. The beauty, and vulnerability, of Otsu's method is that the threshold it identifies is used for the entire image. Using a single threshold means that some input images cannot be well binarized using Otsu's methods. Figure 2.1 shows a document image that has been binarized by Otsu's method. The document depicted on the left side of this figure has stains that are darker than the paper but not nearly as dark as the writing. The output of Otsu’s method, shown on the right, sets these stains to black. This document image was selected to illustrate a type of blemish Otsu's method does not handle well.
Several years later Niblack (1985) proposed a binarization method that computes a different threshold for each pixel of an input image. The threshold is set to $\mu + k\sigma$ where $\mu$ and $\sigma$ are the mean and standard deviation, respectively, of all greyscale values in a square region around the current pixel and $k$ is a constant (typically ranging from .2 to .5). This method gave rise to a class of "adaptive thresholding" algorithms that adapt the threshold applied to each pixel to suit that pixel's local region. The work presented in Chapters 3 and 4 are "adaptive thresholding" algorithms like Niblack's. The flexibility of Niblack's adaptive thresholding enables it to perform significantly better than Otsu's method on many images. However, Niblack's method produces an undesirable artifact when very few (or zero) pixels in the local region should be set to black in the output image. In this case, the standard deviation is small because most of the pixels in the local regions are nearly the same color. This small standard
deviation causes seemingly random pixels to be incorrectly classified as black pixels because Niblack's method responds to subtle noise in the lightly shaded region when there are no dark pixels to detect. Sauvola and Pietikainen (2000) proposed an update to Niblack's method that changes the formula used to compute the threshold depending on properties of the local region. This revised method improves performance when very few pixels in the local region should be black. Figure 2.2 shows a document image that has been binarized by Niblack's method (on the left) and the corresponding output (on the right). The right side of Figure 2.2 clearly shows the “spotting” artifact that Niblack's method commonly produces.

---

**Figure 2.2:** Applying Niblack’s method to an image. The original image is shown on the left and the output of applying Niblack’s method is shown on the right.
2.2 Image Operations

Many of the more recent image binarization techniques use one or more standard image processing operations. These standard operations are covered here to provide a better understanding of the more sophisticated image binarization techniques that will be introduced later.

One basic step many algorithms take is converting a color image to a greyscale image. This transformation is performed one of three ways. They are (1) averaging a pixel's Red, Green, and Blue components together to get its greyscale value, (2) reweighting a pixel's RGB components to mimic the number of rods and cones in the human eye, and (3) using Principal Component Analysis, a well-known matrix operation, to reduce an image's three dimensional RGB values down to a set of one dimensional greyscale values.

Another simple component operation is the Gaussian blur. This operation produces a blurred version of the original image. Each pixel in the blurred image is found by computing the weighted average of pixels from the original image. The weights used when computing this average are taken from a two dimensional Gaussian distribution. Gaussian blurs are often used to reduce the effect of noise in the input image. An example of how increasing the blurring radius of the blurring operation removes more and more detail from an image is shown in Figure 2.3.
Image sharpening begins, somewhat surprisingly, by applying a low radius Gaussian blur to the input image. The difference between the original image and the blurred copy is computed. A multiple of this (usually subtle) difference is then added to the original image to sharpen it (i.e. emphasize all edges by lightening the light pixels and darkening dark pixels). Consequently, the standard image sharpening technique is called unsharp masking. Figure 2.4 depicts sharpening an input image (top left panel) using a small multiple (top right panel), medium multiple (bottom left panel) and large multiple (bottom right panel). Notice, the ‘R’ is a deeper shade of black when more sharpening is used. However, the background of the image also becomes pixelated as more sharpening is used. It should also be noted that strong sharpening produces a “halo” effect around darker portions of the image. This “halo” can be observed can be seen, in the bottom right panel of Figure 2.4.
Figure 2.4: Sharpening an image with an unsharp mask (USM).

Edge detection is another extremely common, and useful, image processing technique. Almost all of the more modern image binarization algorithms introduced later in this chapter use the output of an edge detection algorithm at some point. Edge detection algorithms come in two
flavors: those that produce a numeric value for each pixel and those that produce a binary classification for each pixel. Sobel edge detection (Sobel and Feldman 1968) generates a numeric value for each pixel in the input image by computing the color gradient at that particular pixel. When the gradient is high the color is changing quickly, and the input image is likely to have an edge at that location. On the other hand, Canny edge detection (Canny 1986) classifies each pixel in an image as an edge or not an edge. Canny edge detection identifies "obvious" edges in an input image by finding pixels with large color gradients. These "obvious" edges are then extended by tracing each edge throughout the image using a hysteresis process that depends not only on the current gradient but also on the past gradient. Hysteresis allows Canny edge detection to identify pixels in faint edges that would not otherwise get classified as edge pixels (by tracing an edge from an obvious region into a non-obvious region). Using Canny edge detection introduces two parameters that are required to govern the sensitivity of the hysteresis process.

Applying a blur to an image before performing edge detection will either reduce the "edge strength" values associated with each pixel or reduce the number of pixels classified as edge pixels. This effect is sometimes desirable and sometimes undesirable. If losing edges is undesirable, a bilateral filter can be applied instead. A bilateral filter (Tomasi & Manduchi 1998), which is sometimes called an edge preserving blur, is similar to a Gaussian blur except the weights used to compute the filtered/blurred image reflect both the geometric distance between pixels (as a regular Gaussian blur does) and the difference in color. Thus, pixels that are close together but have very different colors are not blurred together. Bilateral filters are
commonly used to retouch photographs of people because they can "blur away" the appearance of skin blemishes while leaving the silhouette of a face in sharp contrast.

The Wiener filter (Jain 1989) is another noise reduction technique used by some image binarization algorithms. This filtering method was originally conceived for signal processing and is built on the assumption that the incoming signal is actually a combination of the true signal and random noise. The Wiener filter cleans an image by making assumptions about the type of noise in an image, computing statistics that are not strongly affected by this type of noise, and using these statistics to locate and correct noisy pixels.

The last standard image processing operations commonly used in image binarization algorithms are the mathematical morphology operations of erosion, dilation, and closing (Dougherty 1992). These operations subtly manipulate binary images and are typically used to refine the raw output of a binarization algorithm. The erosion operation removes the outer pixel(s) of a shape making it slightly smaller. The dilation operation does the opposite and adds pixels to the outer boundary of a shape. The closing operation fills in concave corners of a shape. Repeatedly applying the closing operation will convert a concave shape to the convex hull of its vertices. The effect of the erosion and dilation operations is shown in Figure 2.5. In this figure the original shape, shown in blue, is eroded to become the yellow shape and dilated to become the green shape.
2.3 **Methods Based on Foreground-Background Separation and Background Estimation**

There is a wealth of research on foreground-background separation for images. This body of research is related to image binarization because it also seeks to isolate important parts of an image. Foreground-background separation techniques have been designed to facilitate optical character recognition by isolating text in color images (Garain et al. 2005, Garain et al. 2006), compress image files (Haffner et al. 1998, Haffner et al. 1999), and provide a crucial starting point for image enhancement methods (Agam et al. 2007). These works are not image binarization techniques, but they are quite similar because their core operation is to isolate pixels that are important.
Document image binarization methods in this vein rely on background estimation. This approach treats the foreground of the image (i.e., the text) as if it sits "on top of" the image's background. The primary assumption here is that the background varies smoothly. The goal is to approximate the smoothly varying background surface and use it to help identify the pixels that jut out above the surface. This approach is like detecting buildings on a hilly landscape using only altitude measurements. You begin by approximating the hilly landscape itself. Then, once the rolling hills are accounted for, you can more easily spot the buildings that have altered the altitude measurements.

Lu & Tan (2007) published a paper describing a technique that fits a single smooth polynomial surface to the pixels in a document image. The document image is then normalized by subtracting off the smooth fitted surface from the original image. The resulting image is then processed with a global thresholding algorithm. This method was designed for document images that are relatively unblemished except for some uneven lighting. For example, photocopies from a thick book often have a noticeably darker region around the spine of the book where the book's page did not rest directly on the scanning surface. This method performs well in this tightly defined use-case but is not well suited other document flaws. Figure 2.6 shows two examples of the uneven lighting problem Lu & Tan’s method was designed to correct. Notice, in both of these examples the unevenly lit image is otherwise clear and crisp.
Lu, Su & Tan (2010) improved the method from above by relaxing the assumptions being made about the polynomial surface. The revised method no longer assumes there was one single low-degree polynomial surface for the whole image. Instead, it computes an interpolating polynomial for each row and column of the image. This enables the fitted background surface to be more flexible and better approximate blemishes that were not well approximated by a single low degree polynomial surface.

Gatos et al. (2006) published a method that uses the binarization method of Sauvola and Pietikainen (2000) to guide its background estimation process. Their technique first uses Sauvola and Pietikainen's method to obtain an initial "rough" binary image. The background estimation process then uses the white or black status of each pixel (in the "rough" binary image) to determine which of two different equations should be used to estimate the background value. Once the background surface of an image is estimated a local thresholding step is performed.
The threshold is computed using the value of the background at a particular pixel, \( B(x,y) \), as well as the corresponding original pixel value, \( I(x,y) \). It should be noted that this method also has significant pre and post processing steps. The preprocessing steps include applying a Wiener filter (Jain 1989), and the post processing steps include applying some morphological operations that refines the binary image produced by the thresholding step.

### 2.4 Binarization Methods Based on Machine Learning

Machine learning techniques have been used to produce a handful of interesting image binarization techniques. These methods are quite different from one another but are presented together here because they use different methods from the same general field of knowledge – machine learning.

Su, Lu & Tan (2010) published “A Self-training Learning Document Binarization Framework”. This paper outlines a method that can be layered on top of existing end-to-end image binarization techniques. They show how their framework can be used to improve the output of Otsu's, Niblack's, and Sauvola and Pietikainen's methods. The framework first applies the “naïve” binarization method to a collection of test images with known ground truth binarizations. The purpose of this step is to compute statistics that are required to identify which pixels are likely to be classified incorrectly. These required statistics are computed by measuring the precision and recall of the “naïve” binarization. Once these statistics are in hand, the framework classifies all pixels in an image into one of three classes: foreground, background, and uncertain. Once the pixels are assigned to a class, the framework must decide how to
reassign the pixels that were labeled as "uncertain". This reassignment is done using k-means clustering that relies on Euclidean distance and other features computed from the input image.

A unique image binarization refinement was proposed by Obafemi-Ajayi et al. (2008, 2010). Their approach can be combined with any other image binarization techniques to improve their outputs if training data exists. The approach compares binarized document images with their corresponding ground truth images (which are often hand generated). Thousands of snippet pairs are taken from each binarized image and its corresponding ground truth image. These snippet pairs build up a look-up table that shows how often one pixel arrangement from a binarization output was corrected to a different pixel arrangement in a ground truth image. This look-up table based technique does not encounter memory problems because it relies on the sparsity of the pixel arrangements and approximate nearest neighbor search to find matches. The technique is effective at removing flaws found in images of documents produced with an old fashion typewriter. Such documents can become flawed when the physical key-press used to write a character is either too light or too strong (respectively producing a faint or smudged character). Their look-up table approach automatically finds other similar – and likely unflawed – characters that can be used to refine the output. This approach is effective at improving text, but it should not be applied to images with both text and images (at least not without text segmentation).
2.5  **Binarization Methods Based on Function Minimization**

A broad collection of image binarization algorithms are based on minimizing a function computed from the input image and the output binarized image. This collection of methods can be broken down into two related groups according to how they frame the problem. One group of methods views image binarization as an energy minimization problem. The second group of methods views binary images as a stochastic sample output from a Markov Random Field. Despite these different viewpoints the two groups of methods are remarkably similar because minimizing an energy function and computing a Maximum *a posteriori* probability configuration for a Markov Random Field require systems of equations that are remarkably similar.

Boykov & Kolmogorov (2004) published a review article covering the numerous ways a min-cut/max-flow algorithm can be used in computer vision. This review article did not focus specifically on image binarization but its lengthy enumeration of the possible uses for min-cut/max-flow make a convincing argument that this optimization technique could be applied to image binarization. Howe (2011) later proposed the first notable energy based image binarization method that employs min-cut. Howe's method assumes it costs energy to set an image pixel to white or black. The exact cost depends on the surroundings of the given pixel. The method begins by assuming it is cheap to set lightly colored pixels to white and darkly colored pixels to black. It also assumes it should cost "extra" to set a pixel to white when its neighbors are set to black and vice versa. Penalizing switches from white-to-black and black-to-white reduces image noise and generates smoother object outlines. Sensibly, this penalty is not applied if an edge is detected at that particular pixel boundary. Once all the cost coefficients are
assigned the final binarized image is obtained by solving a min-cut problem with linear programming. Sun et al. (2006) use a similar energy based method to perform foreground-background separation in real-time on video. They make multiple alterations to Howe's method to suit the new use-case. For example, they make changes to speed-up the computation and handle shadows (which is not a problem for document images).

The second group of binarization methods begins with Wolf & Doerman's (2002) method for binarizing low quality images of text. Their method assumes the output binary image is a stochastic sample from a Markov Random Field (MRF). The MRF's parameters are estimated using training data. This early MRF method does not use the min-cut algorithm to produce the final binary image. Instead, their approach uses simulated annealing to converge on an approximate global optimum. This optimization technique randomly flips pixels from white-to-black or from black-to-white. The flips are accepted or rejected depending on how large the change in the energy potential function is. Over time, the threshold for accepting/rejecting goes down, and the changes made are ever more subtle. A MRF based binarization method that is quite similar to Howe's method is published in Kuk, Cho & Lee (2008) and Kuk & Cho (2009). The primary difference between these methods is in how they compute the costs that enter into the final min-cut optimization. Kuk, Cho & Lee and Kuk & Cho use the mathematics of an MRF to set these costs while Howe uses the Laplacian of the image and other rules. Lelore & Bouchara (2009) proposed a subtly different MRF based image binarization technique. Their updated technique performs better because it: (1) uses training data to estimate the parameters of the underlying the MRF and (2) includes an extra term in the optimization formula to improve character connectivity.
The key criticism of all machine learning techniques is their dependence on training data. For some application areas, training data will readily available. However, when such data are unavailable these machine learning based methods will be rendered inapplicable.

### 2.6 Unique Binarization Methods

Three notable methods are conceptually very different from all other image binarization methods. Perhaps the most unique of these methods is Kim, Jung & Park's (2002) binarization method. Their technique treats the input image as a 3-D terrain map. Their method then mimics rain failing on the sloped terrain. As the method proceeds the simulated rain water pools in low lying terrain (i.e., dark regions of the input image). Black pixels in the output image correspond to portions of the terrain that had standing water while white pixels in the output image correspond to portions of the terrain that remained dry given the amount of simulated rainfall that was dropped of the terrain.

Moghaddam & Cheriet (2010) proposed an image binarization method that uses a multi-scale framework to generate the output image. Their multi-scale system combines the analysis performed on several different versions of the same image. Each version has $\frac{1}{2}$ the width and $\frac{1}{2}$ the height of the version that came before it. This iterative processing ensures that the effect of high frequency noise is limited and that the algorithm focuses on more macro-level objects.

Neves, Zanchettin & Mello (2013) published a binarization algorithm that focuses on identifying objects and binarizing each of them individually. Their method begins by running a standard edge detection routine. Detected edges are then expanded with the dilation operator.
This expansion ensures that when a bounding box is drawn around the detected edge it will fully contain an object of interest from the input image. Next, the image snippet within each bounding box is binarized as if it is a complete image unto itself. This complete binarization algorithm has good results and is conceptually appealing because it breaks the image binarization process down into more manageable steps. Moreover, this method can “plug-in” any other image binarization method when it processes each image snippet from a bounding box.

2.7 The Document Image Binarization Contest (DIBCO) Series

One reason image binarization is a complex topic is that a given input image may have multiple arguably correct binarizations. In fact, Figure 2.7 shows a synthetic test image from Sauvola and Pietikainen's (2000) frequently cited paper that has two correct binary ground truths. The left column of Figure 2.7 shows the test image, and the two possible correct binarizations are shown in the center and right columns. The existence of multiple correct binarization outputs complicates comparing different binarization algorithms.
In response to this problem, the International Conference on Document Analysis and Recognition (ICDAR) hosted a series of Document Image Binarization Contests (DIBCOs). Prior to each contest the organizers create a curated collection of test images as well as one handcrafted ground truth "ideal output" image for each test image. The algorithms submitted to the competition can then be evaluated based on how closely their outputs match the ideal outputs. DIBCOs were held in 2009 (Gatos et al. 2009), 2011 (Pratikakis, Gatos, & Ntirogiannis 2011e), and 2013 (Pratikakis, Gatos, & Ntirogiannis 2013). Each year's test collection consists of a few images of hand written and printed documents that are selected to ensure that each collection contains document images depicting multiple types of degradation like bleed through, blotching, and faint script. The average test collection contains 14 images, 7 of which are printed and 7 of which are handwritten. The 2013 and 2011 DIBCO datasets are available online at:

2.8 The MultiSpectral Text Extraction Contest (MS-TEx 2015)

Image binarization routines are usually provided with input images that were captured with standard visible light (i.e., not infrared, ultra-violet, or monochromatic light). But in 2015 the ICDAR conference held a MultiSpectral Text Extraction Contest (MS-TEx 2015). This competition is similar to the DIBCO competition mentioned earlier except this competition provides eight different images of each historic document in the training/test sets. Each of these eight images was captured using a different monochromatic frequency of light. The MS-TEx 2015 contest announcement (Hedjam 2015a) and contest results (Hedjam 2015b) thoroughly cover the sparse literature on multispectral image binarization methods as this is a new area of study. The MS-Tex 2015 competition description and dataset is available at: http://www.synchromedia.ca/competition/ICDAR/mstexicdar2015.html.

2.9 Judging the Quality of a Binary Image

As the mere existence of the DIBCO competition suggests, judging the quality of a binary image is not as straightforward as it could be. Some software programs like Adobe's Photoshop and the ENH system (Frieder, Lüdtke, & Miene 2007) enable human users to quickly adjust parameters of image manipulation routines on-the-fly to suit a user's personal preference. Relying on a human to optimally tune parameters is workable in some situations. However, it is usually preferable to mathematically, and automatically, arrive at optimal parameter values.
When a ground truth "ideal binarization" exists for a particular input image there are three different statistics that can measure how well an arbitrary binary image matches the ideal binarization. By far the most prominent of these statistics is $F_1$ score (also known as F-score or F-measure). The formula for $F_1$ score is:

$$F_1 \text{ score} = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision}$$

where $Recall = Fraction \ of \ "truly \ black" \ pixels \ detected$

and $Precision = Fraction \ of \ black \ pixels \ that \ were \ "truly \ black"

An important property of $F_1$ score is that this statistic cannot be gamed easily, if at all. It is possible to achieve perfect Recall by returning an entire black image. Likewise, it is possible to achieve a perfect Precision by identifying only the most easily classified black pixels and then setting the remainder of the image to white. The $F_1$ score convolves these two individually game-able statistics into one non-game-able statistic. The two other statistics are peak signal to noise ratio (PSNR) and distance reciprocal distortion metric (DRD) (Lu, Kot & Shi 2004). For the sake of completeness, it should be noted that some researchers view choosing a ground-truth image as a complex problem unto itself (Smith 2010, Smith & An 2012, Shaus et al. 2012). Never the less, having a carefully constructed ground-truth image is useful.

When ground truth images are not available it is possible to use the accuracy of an optical character recognition (OCR) software package like Tesseract (Smith 2007) as a proxy for the
quality of an image binarization algorithm. Many image binarization algorithms, especially prior
to the inception of DIBCO, use this measurement technique. Such papers generally start with an
image of a newspaper article that may or may not have lighting defects or noise. The
binarization algorithm is applied to the image of the newspaper article, and OCR is performed on
the processed output image. The accuracy of the OCR process is then given as a proxy for
quality of the binarization technique.
Chapter 3: Automatic Enhancement and Binarization of Degraded Document Images

This chapter examines the hypothesis that acceptable image binarization parameters can be found automatically. The hypothesis is supported by the presentation of a novel method to automatically enhance and binarize degraded historic images with neither training data nor human interaction. This method was patented in US Patent #8,995,782 “System and Method for Enhancing the Legibility of Degraded Image”. The patented image enhancement method introduced here is unique because it:

- Uses a heuristic to automatically find acceptable image binarization parameters.
- Requires no training data and no human interaction.
- Is well-suited to high throughput image processing and archival efforts because it is embarrassingly parallel due to its input page independence.

This method is illustrated by applying it to selected images from two different corpuses. The first corpus contains scans of historic documents that are currently stored at Yad Vashem Holocaust Memorial Museum in Israel. The second corpus contains test images from the 2011 Document Image Binarization Contest.

3.1 Methodology

The method described in this chapter converts a color or greyscale document image to a strictly black and white document image. The conversion technique is designed to simultaneously reduce the effect of document degradation and highlight the essence of the pre-
degraded document. The ultimate goal is to produce a single black and white image that makes the information in the original document as legible as possible.

We empirically evaluate our method by applying it to images from Yad Vashem's Frieder Diaries. These historic diaries, from the 1940s, survived adverse storage conditions during World War II and provide a wide variety of document image test cases. Diary pages contain typewritten text, handwritten script, pictures, multiple languages, and combinations of these. They also show differing amounts of degradation due to storage condition and paper type. This collection of images is available upon request. The diaries themselves are on permanent loan to Israel's Yad Vashem archives.

The image enhancement and binarization method presented here is based on the guiding principles that "writing" pixels should be darker than "non-writing" pixels nearby and "writing" should generate a detectable edge. Of course, these principles are not universally true; however, they are rarely violated in practice; at least as far as observed herein. Our method is outlined in Figure 3.1, with each of the 4 core steps: obtaining the greyscale image, isolating locally dark pixels, isolating pixels that are near an edge, and combining those groups are discussed in detail below.
3.2 Creating a Greyscale Image

The first step towards obtaining an enhanced black and white image is to create a greyscale version of the input image. We use principle component analysis (PCA) to reduce the 3-dimensional RGB (red, green, and blue) value for each pixel to a single greyscale value.

3.3 Process 1: Isolating Locally Dark Pixels

The second step determines which pixels in the greyscale image are "locally dark". We use a constant sized window of pixels from the greyscale image to analyze each pixel. The window is an \(n\) by \(n\) pixel region where \(n\) is always odd. As we slide this window across the greyscale image we make an "is locally dark" decision about the pixel at the center of this window. Each time the window is moved, we compute the Otsu threshold for the pixels within the window.
the window. If the center pixel is darker than the Otsu threshold we include that pixel in the set of "locally dark" pixels. For a pixel to be black in the final output image it must be flagged as "locally dark" in this step. This requirement is inspired by the general principle that "writing" pixels should be darker than the "non-writing" pixels nearby. The winSize parameter is set automatically using a method discussed in Section 3.6.

3.4 Process 2: Isolating Pixels Near an Edge

The second guiding principle behind our method is that writing should generate a detectable edge. Process 2 isolates all pixels that are near detectable edges thus reflecting the second guiding principle. A summary of this pixel isolation process is depicted in Figure 3.2.

![Diagram of process steps]

**Figure 3.2:** Process used to isolate pixels that are "near an edge".

We begin this process by running Sobel edge detection. The Sobel operator approximates the gradient of the greyscale image at a particular pixel. When the gradient is large, a border between light pixels and dark pixels exists. An example of the output of the edge detection step can be seen in the panel A of Figure 3.3. Notice that the letters are outlined clearly in this example.
Once edge detection has been performed, we blur the resulting image one or more times. The blurring operation applies a Gaussian blur across a 5 by 5 window of pixels. The numBlurs parameter is set automatically using a method discussed in Section 3.6. Next, the pixels within the blurry edge detection image (shown in panel B of Figure 3.3) are clustered into 4 sets: dark, medium-dark, medium-light, and light pixels. An example of this clustering is shown in panel C of Figure 3.3. Pixels that are assigned to the dark, medium-dark, and medium-light clusters are considered "near an edge".

3.5 **Combining Results from Processes 1 and 2**

Processes 1 and 2 generate two sets of pixels: (1) pixels that are "locally dark" and (2) pixels that are "near an edge". The final step towards creating a black and white output image is to compute the intersection of these two sets. Every pixel that is both locally dark and near an
edge is set to black in the output image. If a pixel does not meet both of these criteria, it is set to white in the output image.

3.6 Parameter Selection

The processes discussed in Section 3.3 and Section 3.4 each requires one parameter: \textit{winSize} and \textit{numBlurs}, respectively. One of the more important aspects of this image enhancement and binarization method is that the only two parameters are determined automatically. Automatic parameter selection ensures that this method can be used with as little human interaction as possible.

The \textit{winSize} parameter is set so that the spotting shown in the middle panel of Figure 3.4 is significantly reduced. This spotting is generally caused by subtle noise in the background of an image that makes some background pixels slightly darker than their surrounding pixels. Increasing the \textit{winSize} parameter makes it more likely that a darker “writing” pixel is included in a window when the “is locally dark” decision is made. The inclusion of a darker “writing” pixel reduces the likelihood that false positives occur due to noise in the background of an image. The net result of increasing the \textit{winSize} parameter is that spotting is less prevalent in the output image. The left-hand panel of Figure 3.4 shows a snippet of an input image. The center panel shows the output when the \textit{winSize} parameter is too small and the right-hand panel shows binarization output with a larger \textit{winSize} parameter that reduced the spotting shown in the center panel.
The metric used to set the \textit{winSize} parameter is designed to be sensitive to the spotting we are attempting to minimize. We increase the \textit{winSize} parameter (from an initial value of 9) until our metric no longer changes appreciably. At this point we assume the level of spotting is also not changing appreciably.

The metric we use is the standard deviation of the standard deviations. To compute this metric, we randomly select many (on the order of 10,000) 25 by 25 windows from an output image. We count the number of black pixels in each random window. Next, that count is converted to a set of \( n \) 0's and \((25^2 - n)\) 1's where \( n \) is the number of black pixels in the corresponding window. We then compute the standard deviation of each set of 625 pixel color values. Next we compute the standard deviation of our sample of standard deviations. This metric is sensitive to spotting because the difference between a window composed of only white pixels versus a window composed of almost only white pixels is large.

The \textit{numBlurs} parameter is set second. This parameter is gradually increased until each successive image is nearly identical to the proceeding image. A pair of images is considered nearly identical if 99.5\% of their pixels match. The \textit{numBlurs} parameter is used mainly to enable our method to accommodate images of different resolutions.
3.7 Experimental Results and Discussion

Figure 3.5 and Figure 3.6 show typical results from when our image enhancement and binarization method is applied to images in our dataset. These images were selected to illustrate some of the variety within our dataset as well as how our algorithm responds to handwritten script, typewritten text, and photographs. Areas of interest in these results are circled in red.

Figure 3.5: Sample result: Document M.5_193_67. Areas of interest are circled in red. Automatically set parameters: \(\text{winSize} = 13, \text{numBlurs} = 4\).
The three areas circled in Figure 3.5 correspond to typewritten characters that are significantly lighter than their surrounding characters. Notice that these fainter characters are more legible in the enhanced document image than they are in the original document image (this is especially true when the images are viewed at their intended size). The phrase "hat mich darauf telefonish" is legible despite the image yellowing above "mich" and the boldly written "t" just prior to the faintly typed "uf" and almost undetectable "a".

Figure 3.6: Sample results: Document M.5_193_25. Areas of interest are circled in red. Automatically set parameters: $\text{winSize} = 17$, $\text{numBlurs} = 6$.  

34
The processed diary image on the right side of Figure 3.6 shows two minor defects. In the uppermost circle we see that only the bottom portion of the script loop is retained. A faint detectable edge is generated by the loop that is missing from the processed image. However, that detectable edge is "blurred away" when the 6 blurring operations are applied. The circle at the bottom-right highlights that the discoloration in that corner is not converted to a perfectly clean black and white image. The spotting discussed previously is visible in this corner of the processed image. Note, however, that spotting is not present in most of the right hand margin - the spotting is only prevalent in the corner. This is particularly relevant because the images from Figure 3.4 that introduce the spotting issue are excerpts from the original image shown in Figure 3.6.

The final observation to make about Figure 3.6 is that the four photographs in the original image are recognizable in the final black and white image. The presence of these photographs did not hinder the ability to enhance the faint script to the right of the photos.

### 3.8 The DIBCO 2011 Test Set

Figure 3.7 and Figure 3.8 show excerpts of original problem images HW3 and HW2 from DIBCO 2011 (upper panel), their corresponding ground truth images (center panel), and the result of applying the proposed method to those images (bottom panel). Table 3-I lists the recall, precision, and F₁ score of the proposed method on these images as well as the average F₁ score from the top three methods from the DIBCO 2011 competition. The primary reason the proposed
method’s recall lags behind its precision is that it produces lines that are about one or two pixels thinner than the ground truth images.

Figure 3.7: DIBCO 2011 HW3. Original image, ground truth, the output of the proposed method.

Figure 3.8: DIBCO 2011 HW2. Original image, ground truth, the output of the proposed method.
Table 3-I: Select results from DIBCO 2011

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F₁ Score</th>
<th>Avg F₁ of 1ˢᵗ, 2ⁿᵈ, 3ʳᵈ</th>
</tr>
</thead>
<tbody>
<tr>
<td>HW2</td>
<td>98.1</td>
<td>89.8</td>
<td>93.7</td>
<td>94.4</td>
</tr>
<tr>
<td>HW3</td>
<td>97.9</td>
<td>83.4</td>
<td>90.1</td>
<td>92.7</td>
</tr>
</tbody>
</table>

3.9 Conclusion

The image enhancement and binarization method presented here significantly improves the legibility of degraded historic images in our dataset. The main advantages of this algorithm is that it requires no human action to find parameters that yield good results nor is a training set of images needed. Avoiding the need for human interaction can significantly improve the throughput of image digitization and archival effects. Forgoing a training set enables the approach to be used on any collection. An ancillary benefit of this algorithm is that it is simple to implement and easy to understand.
Chapter 4: Robust Binarization of Degraded Document Images

Using Heuristics

This chapter examines the hypothesis that acceptable image binarization output does not require a parameter search. The hypothesis is supported by the presentation of a method to automatically enhance and binarize degraded historic images without a dedicated parameter search step. The image enhancement method introduced here is unique because it:

- Eliminates the need for a parameter search.
- Has better average performance than other known methods as well as lower variance than other known image binarization methods. Thus, it more consistently provides better results than other document image binarization methods.
- Requires no training data and is well-suited to high throughput image processing and archival efforts due to its input page independence and embarrassing parallelism.

We evaluated our method by applying it to two sets of images from the 2011 and 2013 Document Image Binarization Contests. We also demonstrated our method on a selection of images from a collection of historical document images.

The image enhancement and binarization method presented here is a descendant of, and significant improvement upon, the method introduced in Chapter 3 (Parker et al. 2013). Both methods are based on the guiding principles that: (1) writing should be darker than nearby non-writing and (2) writing should generate a detectable edge. The algorithm from Chapter 3 relies on parameters that are automatically set using heuristics. The parameter setting process has two
notable downsides. First, the process is time consuming because it relies on statistics that can only be computed when multiple fully rendered images are available for comparison. In other words, the parameter setting process can only crown an image as the (likely) best image from a collection of output images after the entire collection of potential output images has been produced. The second problem is that the heuristics guiding parameter selection have not been well studied.

The improved method presented here in Chapter 4 renders the entire parameter setting process unnecessary because it more robustly identifies pixels that are "near" an edge. This revised method also adds two post processing image cleaning steps. The first cleaning removes stray pixels and the second cleaning removes "white islands" from output images. White islands are a specific type of undesirable artifact that can appear in images produced by the prior method. These islands are discussed in a dedicated section that also describes the process to remove them.

4.1 **Methodology**

The image enhancement method described herein converts a color document image to a strictly black and white document image. The method highlights prominent aspects of an image while reducing the effect of degradation found in the original color image. This method, again, is based on the same two guiding principles:

1. "Writing" pixels should be darker than nearby "non-writing" pixels.

2. "Writing" should generate a detectable edge.
This improved method is summarized in Figure 4.1; each of the intermediate steps is discussed in detail in the sections below.

![Figure 4.1: A high level view of the image enhancement method.](image)

### 4.2 Creating a Greyscale Image

The image enhancement process begins by creating a greyscale image from the input color image. We use principal component analysis (PCA) to convert the input color image (which can be viewed as collection of 3-dimensional RGB values) to a greyscale image (which can be viewed as collection of 1-dimensional greyscale values). This step can be skipped if the input image is already in greyscale. However, applying PCA to an image that is already in greyscale will not alter the greyscale image.

### 4.3 Identifying Locally Dark Pixels

Identifying locally dark pixels begins by applying a Gaussian blur to the greyscale version of the input image. The blur radius is set to just one pixel because the purpose of this
A similar blurring step is used in other related work (Canny 1986). Each pixel in the blurred image is analyzed separately by applying Otsu's method to a “snippet of local pixels” extracted from the slightly blurred image (similar to the method from (Moghaddam & Cheriet 2012)). A pixel is added to the set of locally dark pixels if it is set to black when Otsu's method is applied to its corresponding snippet of local pixels. If a pixel is to be black in the final output image it must be flagged as "locally dark" in this step (excluding two exceptions discussed in Section 4.6). This requirement means that the output of this step (Figure 4.3, right panel) can be viewed as a filter each pixel must pass to be included in the final output. This process is illustrated in Figure 4.2 and example images are shown in Figure 4.3.

**Input:** Greyscale document image

1. Apply Gaussian blur with a radius of 1 pixel
2. For each pixel: Apply Otsu’s method to greyscale values found in its snippet of local pixels

**Output:** Black and white image identifying locally dark pixels

*Figure 4.2: Process that identifies locally dark pixels.*
The right side of Figure 4.3 illustrates a typical output of this process. Notice that there is a sharp white outline around the writing. Unfortunately, this process is highly ineffective when applied to the background portion of the image. When this filter is applied to the background it merely highlights noise from the greyscale image (similar to Niblack’s method as depicted in Figure 2.2). The next section introduces a second filter that, when intersected with this filter, clarifies the background portion of the image.

The “snippets of local pixels” mentioned above are created by extracting an $n$ pixel by $n$ pixel region where the region is centered on the pixel being analyzed. We require $n$ to be odd so that there is always a single pixel at the exact center of the $n$ by $n$ region. The results illustrated above were obtained when $n$ was set to 21.
4.4 **Identifying Pixels that are Near an Edge**

Next, we identify pixels that are near an edge using the process shown below in Figure 4.4. Identifying pixels that are near an edge begins by sharpening the starting greyscale image with an unsharp mask (USM). Like the hysteresis process of Canny edge detection, applying the USM increases the likelihood of detecting faint edges. Next, Sobel edge detection is performed resulting in an image like that shown in the left panel of Figure 4.5. After edge detection, we compute the standard deviation of all the greyscale values found within a square snippet of local pixels. Each pixel in the output image (for this particular sub step) is normalized to a shade of grey ranging from 0 to 255 according to its standard deviation of "local greyscale values". The result is an image where dark shades of grey indicate an edge was detected locally and light shades of grey correspond to regions without sharp edges. Finally, applying Otsu's method to the image showing normalized standard deviation values, as shown in the center panel of Figure 4.5, produces a filter that isolates pixels that are near an edge. An example of the output of this whole process is shown in the right panel of Figure 4.5.

<table>
<thead>
<tr>
<th>Input: Greyscale document image</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Sharpen image</td>
</tr>
<tr>
<td>2. Apply Sobel edge detection</td>
</tr>
<tr>
<td>a. Normalize greyscale values</td>
</tr>
<tr>
<td>3. For each pixel: compute standard deviation of</td>
</tr>
<tr>
<td>greyscale values in snippet of local pixels</td>
</tr>
<tr>
<td>a. Normalize standard deviations</td>
</tr>
<tr>
<td>4. Apply Otsu’s method</td>
</tr>
</tbody>
</table>

Output: Black and white image identifying pixels that are near an edge

*Figure 4.4: Process that identifies pixels that are near an edge.*
4.5 The Intersection and Snippet Sizes

The results obtained when identifying locally dark pixels and pixels that are near an edge can be viewed as filters based on one of the guiding principles described at the beginning of the Methodology section. As shown in the last panel of Figure 4.6 the intersection of these filters generates a strikingly good black and white version of the input image. The quality of this intersection is high as long as the snippets used to identify locally dark pixels (as in the center-left panel of Figure 4.6) are larger than the snippets used to identify pixels that are near an edge (as in the center-right panel of Figure 4.6). If this condition is not met, the intersection of these filters will contain a noticeable "halo" artifact around all detected text. An example of this "halo" artifact is shown in Figure 4.7. The artifact is caused because the black regions produced when identifying pixels that are near an edge (center-right panel) are too big with respect to the
white outlines produced when identifying locally dark pixels (center-left panel). When this occurs a small portion of the noisy region from identifying locally dark pixels (center-left panel) is seen in the final intersection. To prevent this artifact the snippet size is set to 15 when identifying pixels that are near an edge and 21 when identifying locally dark pixels. Empirical results show that keeping snippet sizes in a ratio 3:4 produce good results.

Figure 4.6: The input and important outputs of the proposed method. The input image (far left) and the output intersection (far right) of the locally dark pixels identified in Section 4.3 (left-center) and the pixels that are near an edge identified in Section 4.4 (right-center).

Figure 4.7: The visual artifact caused by using a snippet size that is too small when computing locally dark pixels.
4.6 Cleaning the Image

The intersection from Section 4.5, generally speaking, is a good result. Yet, two image improving corrections can be made. The first correction fixes what are likely to be erroneously classified pixels (white or black pixels can be corrected in this step). The second correction fixes what are likely to be erroneously classified white regions.

4.6.1 Stray Pixel Correction

The first cleaning step looks for stray pixels in the intersection. The presumption underlying this step is that a black (or white) pixel does not typically appear by itself or nearly by itself. To remove stray pixels, we examine each pixel along with its eight neighbors. When the pixel at the center of a group of 9 pixels is outnumbered by pixels of the opposite color by 1-to-8 or 2-to-7, it is flipped to the locally dominant color. We do not flip pixels that are outnumbered by 3-to-6 because doing so would destroy a fine line of pixels (see the bottom right panel of Figure 4.8). Figure 4.8 illustrates 4 examples in which the center pixel would be corrected and 2 examples in which the center pixel would be left unchanged.
4.6.2 White Island Correction

The method as described so far would not correctly handle large regions that should be classified black in the output image. The problem occurs because the pixels in the center of a large black region may not be included in the set of pixels that are "near an edge". A document with unusually large font may exhibit this error as shown in Figure 4.9. We refer to these incorrectly classified regions as "white islands".
Correcting white islands begins by finding contiguous regions of white that are surrounded by a single contiguous region of black - i.e., a black boarder. When a (white island, black board) pair is identified in the intersection computed in Section 4.5 we must refer back to the greyscale image produced at the beginning to determine if a correction should be made. If a correction is indicated then all the pixels in the white island are set to black thus "plugging a hole" in the black boarder.

We assume a correctly classified white island contains pixels with a statistically different mean greyscale value than the pixels within the black boarder. Consequently, to determine if a white island was incorrectly classifieds we perform a two sample z-test to see if pixels corresponding to those in the white island (but selected from the greyscale image) are statistically different from pixels corresponding to those from the black boarder (but selected...
from the greyscale image). If the pixels found in these regions are not statistically different, we flip the color of all the pixels in the white island to be black.

4.7 **Method Summary**

There are four aspects of the method described above that are worth emphasizing explicitly. First and foremost, this method requires no training data; therefore, this method can be applied to any image dataset. Second, this method is trivial to parallelize because: every sub-step is itself parallelizable, and the method as a whole is input page independent. Third, this method is simple to implement. There are no expectation maximization calculations to perform nor are there any other complex operations. Finally, this method requires no human (or non-human) interaction to set parameters. Thus, this method can easily be incorporated into a high throughput image processing/preprocessing workflow.

4.8 **Experimental Results**

Three document image datasets are used to evaluate the proposed method. The first two datasets come from the 2011 and 2013 Document Image Binarization Contests (DIBCO) that were held at the International Conference on Document Analysis and Recognition (ICDAR). The images within the DIBCO corpuses were selected because they reflect various types of document degradation including bleed through, blotching, and faint script. Each of these DIBCO corpuses contains 8 images of hand written script as well as 8 images of printed script. Importantly, these corpuses permit a methodical evaluation of an image binarization algorithm.
because they contain one hand-created strictly black and white "ground truth" image for each of the test images. The third dataset contains scans of historic documents that are currently stored at Yad Vashem Holocaust Memorial in Israel. The input-output results shown from this corpus are selected both to illustrate the versatility of the proposed method as well as to illustrate the variety of test cases available in this corpus. Pages from this corpus contain typewritten text, handwritten script, pictures, multiple languages, and combinations of these.

4.9 DIBCO Results

As discussed above, each DIBCO test image comes with a corresponding hand-created black and white ground truth image. These ground truth images permit each output image to be compared against the “ideal result”. One metric used to judge the DIBCO competition is F₁ score. Figure 4.10 shows how the proposed method compares to its predecessor (i.e., Chapter 3) as well as the 1st (Lelore & Bouchara 2009) 2nd (Lu, Su & Tan 2010) and 3rd place (Howe 2012) methods (out of 18) from the 2011 DIBCO competition.
Figure 4.10: The proposed methods outperforms its predecessor and never fails catastrophically on any image from the 2011 DIBCO dataset.

Table 4-I: Top DIBCO 2011 results and the proposed method

<table>
<thead>
<tr>
<th></th>
<th>1st Place</th>
<th>2nd Place</th>
<th>3rd Place</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean F₁ Score</td>
<td>80.9</td>
<td>85.2</td>
<td>88.7</td>
<td>88.9</td>
</tr>
<tr>
<td>Median F₁ Score</td>
<td>92.3</td>
<td>92.1</td>
<td>90.6</td>
<td>89.7</td>
</tr>
<tr>
<td>Var of F₁ Score</td>
<td>847.9</td>
<td>322.7</td>
<td>53.1</td>
<td>20.5</td>
</tr>
</tbody>
</table>

Table 4-II: Top DIBCO 2013 results and the proposed method

<table>
<thead>
<tr>
<th></th>
<th>1st Place</th>
<th>2nd Place</th>
<th>3rd Place</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean F₁ Score</td>
<td>91.3</td>
<td>91.3</td>
<td>90.6</td>
<td>91.1</td>
</tr>
<tr>
<td>Median F₁ Score</td>
<td>95.5</td>
<td>95.6</td>
<td>95.5</td>
<td>92.7</td>
</tr>
<tr>
<td>Var of F₁ Score</td>
<td>77.6</td>
<td>151.4</td>
<td>106.7</td>
<td>32.7</td>
</tr>
</tbody>
</table>
As shown in Table 4-I, the proposed method has a higher mean F\textsubscript{1} score than any method entered in the 2011 competition. The proposed method also has a significantly lower variance than any of the top 3 methods. The dramatic difference in variance is due to the 1\textsuperscript{st} place and 2\textsuperscript{nd} place algorithms failing catastrophically on image PR6 and/or PR7 of the DIBCO 2011 dataset. These two images, shown in Figure 4.13 and Figure 4.14, have textured backgrounds that complicate distinguishing the text from the background texture. Table 4-II compares the proposed method against the top entrants in the 2013 DIBCO competition. Once again, the proposed method has a high average F\textsubscript{1} score and a significantly lower variance than other top methods. The variances from DIBCO 2013 are not as high as the variances from the 2011 competition. It is not known if the reduction in variance from 2011 to 2013 is due to unpublished advances in the binarization algorithms submitted or if it is due to the absence of test images with textured backgrounds in the 2013 DIBCO test set. Figure 4.11 – Figure 4.14 show examples of DIBCO 2011 test images and their corresponding black and white output images.
Figure 4.11: DIBCO image HW4 and its corresponding black and white image.

Figure 4.12: DIBCO image HW7 and its corresponding enhanced image.
4.10 Examples from the Frieder Diaries

The Yad Vashem Armin Frieder Diaries image corpus contains approximately 900 high resolution scans of historically significant documents (Frieder 1938-1946). Most of the originals were authored from the late 1930s to the middle of the 1940s. Due to a variety of causes some of
the original documents were better persevered than others. Figure 4.15 through Figure 4.18 show how the proposed method performed when it was applied to 4 different images from this corpus. The pages in Figure 4.15 and Figure 4.16 show different types defects. The excerpt shown in Figure 4.15 has a two-tone effect while the excerpt shown in Figure 4.16 contains bleed-through (where the typewritten script from the reverse side is visible). The pages shown in Figure 4.17 and Figure 4.18 are both difficult to read with the naked eye. The blotching in Figure 4.17 and the faint text in Figure 4.18 leave the script difficult to understand. Figure 4.15 through Figure 4.18 contain excerpts from four different diary pages as well as their corresponding enhancements. The enhanced versions are noticeably clearer. It is worth noting that the enhanced version of diary page M.5_193_95 (Figure 4.17) has no blotching even though the dotted i’s, commas, and accent marks were retained.
Podpisaná miestna skupina cionistického svážu pre Slovensko v Novom Meste nad Váhom zastúpená Arminom Friederom hlavným rabinom ako predsedom a Ing. Deziderom Šoložvárom ako tajomníkom týmto účtive žiadame o povolenie založenia uvedeného spolku, poťažne povolenie učinkovania.

Túto našu žiadosť podporujeme nasledovnými deklami:

1. Dňa pod 1/ pripojených stanoch Ústredného svážu cionistického pre Slovensko, je tento sváž ako taký povolený a stanovery jeho sú schválené.
<table>
<thead>
<tr>
<th>Betreff</th>
<th>Ihre Nachricht</th>
<th>Unsere Zeichen</th>
<th>Tag</th>
</tr>
</thead>
</table>

Wir danken Ihnen sehr für Ihre ausführliche Informationen.
Wir möchten folgendes resümieren:
Wir nehmen gerne zur Kenntnis, dass Sie die innerliche und
Figure 4.17: Excerpt from the input and output of diary image M.5_193_95.
Figure 4.18: Excerpt from the input and output of diary image M.5_193_207.
4.11 Conclusion

The image enhancement and binarization method presented herein was evaluated by applying the method to the corpuses distributed as part of the 2011 and 2013 Document Image Binarization contests. The proposed method has a higher average performance than any entrant in the 2011 Document Image Binarization contest. Moreover, it has a significantly lower variance than all of the top entrants into the 2011 competition. The proposed method has equally good results in the 2013 DIBCO competition. Consequently, the proposed method returns high quality results more consistently than other image binarization methods.

The proposed method was also applied to select images of pages found in the Yad Vashem's Frieder Diaries - a real world corpus of historically significant documents with corresponding images. When the proposed method was applied to the diary images containing a variety of defects the results showed no sign of the defects that occluded the original document.
Chapter 5: Robust Multispectral Text Extraction on Historic Documents

This chapter addresses a slightly different image binarization problem than the one addressed in Chapters 3 and 4. The goal of producing a clean black and white document image is the same but the input to the problem is different. This chapter covers the case in which eight greyscale images of the same document are available as opposed to a single color or greyscale input. These eight images are each captured with a different mono-chromatic wavelength of light. For example, one input image was captured using light with a wavelength of 340 nm (ultraviolet) and another input image was captured using light with a wavelength of 900 nm (infrared).

This chapter examines the hypothesis that a composite image, built from a combination of the eight mono-chromatic input images, could be passed to the single-input image binarization technique from Chapter 4 to achieve results superior to any single input image. This hypothesis is supported by the proposed multispectral image binarization method that:

- Performs as well as the winner of the MS-TEx 2015 competition.

- Uses a simple rule to generate an improved input image for the single-input method. This rule performs best when applied to just the MS-TEx 2015 training set, just the MS-TEx 2015 test set, and the combination of both sets.

To support experimentation, evaluation, modifications, and improvements by others, the source code for this effort is available online at: https://github.com/jon1van/ImageBinarization.
5.1 Methodology

5.1.1 Binarization of a Single Input Image

The binarization method proposed in Chapter 4 is used to convert a greyscale image to a black and white image. The caveat here is that the input to this single input binarization technique is a composite image computed from some combination of the eight available input images. The next section discusses how the composite image is computed.

5.1.2 Binarization of Multispectral Input

As described in Chapter 2, the multispectral historic document image dataset provided by the MS-TEx 2015 competition includes 8 different image files for each document in the collection as opposed to one image file for each document as was the case in the DIBCO competitions. Each of the 8 greyscale images in an "image set" depicts the same document as seen under a different spectrum of light. Table 5-I describes the 8 different image files within each image set. Figure 5.1 depicts three of the eight images in the z76 image set taken from the MS-TEx competition training set. F2.png is on top, F5.png is in the middle, and F8.png is on the bottom. Notice, the background/texture of the paper itself is clearer in images F5 and F8 but the writing is clearer in images F2 and F5. See (Hedjam 2015a) and (Hedjam 2015b) for more about this dataset.
Table 5-I: Description of the eight images within an “image set” from the MS-TEx 2015 Competition

<table>
<thead>
<tr>
<th>Image</th>
<th>Wavelength (nm)</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1.png</td>
<td>340</td>
<td>UV</td>
</tr>
<tr>
<td>F2.png</td>
<td>500</td>
<td>Visible 1</td>
</tr>
<tr>
<td>F3.png</td>
<td>600</td>
<td>Visible 2</td>
</tr>
<tr>
<td>F4.png</td>
<td>700</td>
<td>Visible 3</td>
</tr>
<tr>
<td>F5.png</td>
<td>800</td>
<td>IR 1</td>
</tr>
<tr>
<td>F6.png</td>
<td>900</td>
<td>IR 2</td>
</tr>
<tr>
<td>F7.png</td>
<td>1000</td>
<td>IR 3</td>
</tr>
<tr>
<td>F8.png</td>
<td>1100</td>
<td>IR 4</td>
</tr>
</tbody>
</table>
Figure 5.1: Three of the eight images from the MS-TEx z76 image set.
We hypothesized that a composite image produced using equation (1) or (2) to combine the mono-chromatic images described in Table 5-I will generate improved results from the single-input image binarization method from Chapter 4.

\[ F_i - F_j \]  \hspace{1cm} (1)

\[ F_i + (F_j - F_k) \]  \hspace{1cm} (2)

Where \( F_i, F_j, \) and \( F_k \) indicate one of the eight images listed in Table 5-I. Equation (1) represents the possibility that the difference between two well-chosen images could "subtract off" the background texture of an image leaving behind a cleaned image. Equation (2) represents that possibility that the difference between two images may directly approximate the background texture of an image thus producing a "correction term" that could be used to clean a 3rd image. Equations (1) and (2) rely on image addition and subtraction. These operations occur on a pixel by pixel basis but minor normalization steps are added to ensure all output pixels range from 0 to 255 in value.

Given this hypothesis we systematically binarized all possible \((F_i, F_j)\) couples and \((F_i, F_j, F_k)\) tuples. The best possible input of either of these forms is:

\[ F_2 + (F_5 - F_6) \]  \hspace{1cm} (3)

Where \( F_2, F_5, \) and \( F_6 \) are three of the monochromatic input channels described in Table 5-I. It is interesting to note that the multispectral approach outlined in equation (3) facilitates removing image texture/background that caused some top algorithms from DIBCO 2011 to fail catastrophically (see Figure 4.10 and Figure 4.14).
Document Neutral Training

The MS-TEx competition distributed a training set (with 21 image sets) and a test set (with 10 image sets). It is important to emphasize that the differences that are important in these data are not the differences and/or similarities between the documents at the heart of each image set but the differences and/or similarities regarding how the eight wavelengths of light interact with the paper each document is written on. For example, the last panel of Figure 5.1, produced using infrared light, clearly shows the wrinkles and folds on the piece of paper document z76 was written on. The fact that infrared light interacts with paper and permits the imaging of wrinkles and folds is not likely to change from document to document. Consequently, it is not surprising that the rule for composing an input image shown in equation (3) is the best possible rule (obeying either form (1) or (2)) whether you consider just the training set, just the test set, or the combination of both the training and test sets.

5.2 Evaluation

The multispectral image binarization method is evaluated against the 2015 MultiSpectral Text Extraction Contest (MS-TEx) results. The MS-TEx competition was held at the International Conference on Document Analysis and Recognition in 2015. The MS-TEx data contains 21 training image sets and 10 testing image sets. Table 5-I describes the 8 different image channels within each image set.
Images produced by algorithms entered into the MS-TEx competition are judged using several metrics including their $F_1$ score. Table 5-II gives the results of using the input computed as per equation (3) on the data from the MS-TEx competition. The second row of this table does not list variance values for the 1st, 2nd, and 3rd place entrants as they are unknown. The last column of Table 5-II gives the mean and variance of the $F_1$ score that was computed using only the 10 image sets from the test collection. The numbers in parenthesis in the last column of Table 5-II were computed using both the test and training collections. As shown in Table 5-II the proposed methods has a high mean $F_1$ score when compared to other top performing methods from the MS-TEx 2015 competition.

<table>
<thead>
<tr>
<th></th>
<th>1st Place</th>
<th>2nd Place</th>
<th>3rd Place</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean $F_1$ Score</td>
<td>83.3</td>
<td>81.9</td>
<td>79.1</td>
<td>83.3 (84.8)</td>
</tr>
<tr>
<td>Var of $F_1$ Score</td>
<td>Unknown</td>
<td>Unknown</td>
<td>Unknown</td>
<td>44.0 (35.5)</td>
</tr>
</tbody>
</table>

5.3 Conclusion

Image binarization is an important preliminary step in multiple image processing operations like optical character recognition and page segmentation. The single-input binarization method from Chapter 4 can be used to achieve superior results when multispectral inputs are available. In this case, a composite image computed from three images that were captured with different monochromatic light sources shows little background texture and permits
superior binarization results. The rule used to produce the composite image passed to the
binarization method is simple and reflects the physics of how infrared light can be used to correct
for wrinkles and folds on a piece of paper that might otherwise degrade document binarization.
Source code developed is available online at: https://github.com/jon1van/ImageBinarization.
Chapter 6: Conclusion

Extracting information from images of documents is easier when the image is crisp, clear and devoid of noise. This dissertation makes three contributions towards reliably removing noise from imperfect document images to generate better images for other image processing algorithms and improved human legibility.

The first contribution is a novel image binarization algorithm that embodies the guiding principles that "writing" should be darker than "non-writing" and that "writing" should generate a detectable edge. The cornerstone of this algorithm, found while examining the hypothesis that acceptable binarization parameters can be found with an automatic parameter search, is a process that automatically estimates good parameters for its two core sub-steps. The resulting algorithm is different from other binarization techniques because it requires no human interaction to determine parameters nor does it use training data to set the parameters. Consequently, this approach is embarrassingly parallel because each input image is processed independently of all other input images. This contribution was patented in US Patent #8,995,782 "System and Method for Enhancing the Legibility of Degraded Image" and published at the 12th International Conference on Document Analysis and Recognition (ICDAR) in middle of 2013 (Parker 2013).

The second contribution is a pair of significant improvements to the method that automatically sets its own parameters. The improved method is a result of studying the hypothesis that acceptable binarization output does not require a parameter search. The resulting method has higher quality output and is more computationally efficient because the costly parameter setting step was eliminated. Both of these improvements were enabled by more
effectively identifying pixels that are both locally dark and near an edge. A patent for this improved method has been allowed, but not yet issued, under patent application number 13/949,799 "System and Method for Enhancing the Legibility of Images". This improved method was also published at Document Recognition and Retrieval (DRR) XXI in early 2014 (Parker 2014).

The final contribution of this dissertation is a method to address multispectral input developed while examining the hypothesis that a composite image, built from a combination of the eight mono-chromatic input images, could be passed to the single-input image binarization technique to achieve results superior to any single input image. The approach given relies on the way different frequencies of light interact with the paper a document is written on. These differences permit the wrinkles, folds, and texture of a piece of paper to be captured separately from the ink. Surface defects are adjusted for and the resulting cleaned image can be processed directly. Importantly, this approach does not attempt to model how different inks reflect different frequencies of light; such an approach would produce a methodology that is sensitive to the document image data it was trained on. The multispectral approach described has a mean performance that is statically indistinguishable from other top multispectral image binarization methods, including the winner of the MS-TEx 2015 competition.

In summary, the novel contributions of this dissertation are:

- This dissertation introduces an image binarization technique that uses a heuristic to automatically find acceptable parameters. This heuristic requires no training data and
no human interaction. Therefore, this image binarization technique is well-suited to address large image corpuses because it is embarrassingly parallel.

- This dissertation introduces an improved version of previous image binarization technique that eliminates the need for a parameter search. The improved method has better average performance as well as lower variance than other known image binarization methods. Thus, it more consistently provides better results than other document image binarization methods. Additionally, the improved method still requires no training data, no human interaction, and can be applied in an embarrassingly parallel fashion to tackle large image corpuses.

- This dissertation introduces a method to address image binarization where multispectral input images are available. The multispectral approach has a mean performance that is statically indistinguishable from other top multispectral image binarization methods, including the winner of the MS-TEx 2015 competition. The multispectral approach corrects for the wrinkles and folds on a piece of paper but does not attempt to model how different inks on a particular document reflect different frequencies of light.
References


