ENHANCING MULTISPECTRAL IMAGERY OF ANCIENT DOCUMENTS

by

Trace A. Griffiths

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Dr. Todd K. Moon Major Professor	Dr. Jacob H. Gunther Committee Member
Dr. Gene A. Ware Committee Member	Dr. Byron R. Burnham Dean of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2011

Abstract

Enhancing Multispectral Imagery of Ancient Documents

by

Trace A. Griffiths, Master of Science

Utah State University, 2011

Major Professor: Dr. Todd K. Moon
Department: Electrical and Computer Engineering

Multispectral imaging (MSI) provides a wealth of imagery data that, together with modern signal processing techniques, facilitates the enhancement of document images. In this thesis, four topic areas are reviewed and applied to ancient documents. They are image fusion, matched filters, bleed-through removal, and shadow removal. These four areas of focus provide useful tools for papyrologists studying the digital imagery of documents. The results presented form a strong case for the utility of MSI data over the use of a single image captured at any given wavelength of light.

(103 pages)

Acknowledgments

I would especially like to thank Dr. Gene Ware for his constant support and advice. He has selflessly provided access to the vast amount of imagery he has taken of ancient artifacts over the past decade. His guidance and help in obtaining permissions for using the imagery was key to completing this thesis.

I would also like to thank Dr. Todd Moon and Dr. Jacob Gunther for instructing me in signal processing theory and their insights on how it could be applied to documents. Dr. Doran Baker has been very kind in his financial support of this thesis. Dr. Thomas Wayment has been extremely helpful and kind in sharing his knowledge of papyrology and interpreting the results of this thesis. I would also like to thank Dr. Dirk Obink at the University of Oxford for granting our petition to use the imagery.

Most importantly, I would like to thank my wife, Amber, for her wonderful encouragement of my endeavors. None of my accomplishments would have been possible without her.

Trace A. Griffiths

Contents

Page

Abstract . . . . . **iii**

Acknowledgments . . . . . **iv**

List of Tables . . . . . **vii**

List of Figures . . . . . **viii**

Acronyms . . . . . **xi**

1 Introduction **1**

2 Data Acquisition **5**
2.1 Multispectral Imaging Technology 5
2.2 Multispectral Image Data 7
2.3 Test Set Permissions 8
2.4 Pre-Processing 9

3 Image Fusion **12**
3.1 Pixel Level 13
3.1.1 Image Blending 13
3.1.2 Weighted Combinations 15
3.1.3 Principal Component Analysis 15
3.1.4 Wavelet Transform 16
3.1.5 Contrast Pyramid 16
3.2 Feature Level 18
3.3 Image Level 18
3.3.1 Color Composites 18
3.3.2 Checkerboard 19
3.4 Examples 20
3.5 Conclusion 20

4 Matched Filter Detection **24**
4.1 Models 24
4.2 Derivations 26
4.2.1 Simple Matched Filter (SMF) 26
4.2.2 Clutter Matched Filter (CMF) 26
4.2.3 Matched Subspace Filter (MSF) 29
4.2.4 Linearly Constrained Minimum Variance (LCMV) 30
4.2.5 Constrained Energy Minimization (CEM) 31
4.2.6 F-statistic 31

4.3 Observations . . . 31
4.3.1 Mean Removal . . . 32
4.3.2 Covariance Matrix . . . 32
4.3.3 Additional Observations . . . 33
4.4 Best Filter Methods . . . 35
4.5 Conclusion . . . 39

5 Bleed-Through Removal . . . 42
5.1 Introduction . . . 42
5.2 Methods . . . 43
5.3 Test Results . . . 47
5.3.1 Ancient Scroll . . . 47
5.3.2 19^{th} Century Document . . . 48
5.4 Conclusion . . . 49

6 Shadow Removal . . . 55

7 Summary and Conclusion . . . 61
7.1 Results . . . 61
7.2 Discussion . . . 63

References . . . 66

Appendices . . . 71
Appendix A MSI Enhancement Functions for MATLAB . . . 72
A.1 Creating an Image Cube . . . 72
A.2 Image Fusion Functions . . . 73
A.3 Matched Filtering . . . 76
A.4 Bleed-Through Removal . . . 80
A.5 Shadow Removal Functions . . . 82
Appendix B IEEE Asilomar Paper . . . 87

List of Tables

Table Page

4.1 Description of the matched filter acronyms. 27

4.2 Detection equations divided into four sections from Foy [46], Manolakis [47], Chang [48], and Hastie *et al.* [49], respectively, where $\mathbf{s}$ is the target spectrum, $\mathbf{x}$ is the observed data, $\mathbf{R}^{-1}$ and $\mathbf{\Gamma}^{-1}$ are inverted covariance and correlation matrices of the data with subscripts x, b, and s indicating the observed, background, and target dataset wherever specified, and Λ_x and V_x are the eigenvector and eigenvalue matrices of $\mathbf{R}$. 28

4.3 Results of whether or not to remove means and whether to use the covariance or correlation matrix. 38

4.4 Condensed list of useful matched filters, where the hat ($\hat{}$) indicates that the cube mean has been removed from the vector. 40

7.1 Listing of useful enhancement methods. 62

7.2 Most effective matched filters for different texts, listed in order of quality. . 63

List of Figures

Figure Page

1.1 Image cube representation of multispectral data [1]. 1

1.2 Oxyrhynchus papyrus fragment 1822 original color photograph [6] and 950 nm MSI image [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 2

2.1 MSI filter wheel. 6

2.2 MSI system computer setup. 7

2.3 MSI equipment setup at the Sackler Library, 2006. 8

2.4 Vignetting calibration frame. 10

2.5 Original and vignetting corrected images. 11

3.1 PCA component images to be fused [1]. 20

3.2 A comparison of image blending techniques [1]. 21

3.3 Pyramid and wavelet image fusion at decomposition level 7 using an absolute maximum fusion rule for the details and selecting PC 1 as the low-pass fusion rule [1]. These results were created using the Rockinger's image fusion toolbox [37]. 22

3.4 Color composite examples using the Derveni fragment C4 [1]: an RGB color composite from PC2,1,1 (a), a CMYK composite from PC1,2,3,1 (b), a composite formed by transforming PC1,2,1 to the IHS color space (c), and a composite formed by transforming PC1,2,3 from the IHS space to RGB (d). 23

4.1 Comparison of matched filters with and without a covariance matrix using MSI images of Oxyrhynchus fragment 1822 [7]. SMF and SAM do not use the covariance while CMF and ACE do. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 34

4.2 Comparison of matched filters using a covariance or correlation matrix. CMF uses a covariance matrix while CEM uses a correlation matrix. Results were obtained using MSI images of Oxyrhynchus fragment 1822 [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 35

4.3 Matched filtering results for the scratched out text found in Oxyrhynchus fragment 3340 [7]. The SAM matched filter was used with three different mean subtraction methods: no mean removed (b), background mean removed from the target (c), and cube mean removed from the target (d). The SAM results have been contrast enhanced. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 36

4.4 Comparison of matched filters using MSI images of Oxyrhynchus fragment 1822 [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 41

5.1 A cropped section of the uncorrected recto (a) and horizontally flipped verso (b) scroll fragment A [7] at 700 nm. The dark characters are the recto/verso text with hazy, gray characters as bleed-through. Each cropped section corresponds to the same registered area in both (a) and (b). Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 48

5.2 Scroll results for PCA, ICA, and symmetric orthogonalization using a stacked image cube of fragment A [7]: Principal components 1 and 2, independent components 1 and 12, and symmetric orthogonalization components 8 and 15. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 51

5.3 Original fragment A recto [7] with bleed-through removed using direct PC data vector subtraction, least squares method with a=1.0831, and the solver method with a=.4513. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 52

5.4 Original fragment A verso [7] with bleed-through removed using direct PC data vector addition, least squares method with a=-1.0831, and the solver method with a=-.4513. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 52

5.5 Bleed-through removed from fragment A [7] using the negated subtraction scaling and the recalculated solver scaling a=.3844. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 53

5.6 19^{th} century uncorrected document recto at 650 nm. The word *Book* is the overtext and the hazy letters filling the rest of the frame are bleed-through. 53

5.7 Bleed-through removed from the stacked image cube of the 19^{th} century document by adding the PC data vectors, the least squares method, and the solver method. 53

5.8 19^{th} century document results for PCA, ICA, and symmetric orthogonalization using the recto text image cube: Principal components 1 and 2, independent component 6, and symmetric orthogonalization component 12. . . 54

5.9 Bleed-through removed from recto image cube of the 19^{th} century document using PC data vector addition, the least squares method with a=2.3835, and the solver method with a=.3069. 54

6.1 Oxyrhynchus fragment 2383 at 800 nm as an example of shadows in MSI imagery. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 55

6.2 An example from Oxyrhynchus fragment 2838 [7] demonstrating how a light shadow changes at different wavelengths. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 56

6.3 An example from Oxyrhynchus fragment 2383 [7] demonstrating how a dark shadow changes at different wavelengths. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 57

6.4 Thresholded QDA result for light shadow detection (a), which is used to mask off the light shadows in Oxyrhynchus fragment 2383 (b) [7]. Thresholded QDA result for dark shadow detection (c), which is used to mask off the dark shadows (d). Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 58

6.5 Thresholded QDA result for light and dark shadow detection combined (a), which is used to mask off both light and dark shadows in Oxyrhynchus fragment 2383 (b) [7]. This is compared with the original color image (c) [6]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. 59

Acronyms

ACE	Adaptive Coherence Estimator
BSS	Blind Source Separation
CEM	Constrained Energy Minimization
CMF	Clutter Matched Filter
DMF	Distance Matched Filter
CMYK	Cyan Magenta Yellow Key (Black)
DOLP	Difference of Low-Pass
HMM	Hidden Markov Model
ICA	Independent Component Analysis
IHS	Intensity Hue Saturation
LCMV	Linearly Constrained Minimum Variance
LDA	Linear Discriminant Analysis
LED	Light-Emitting Diode
LRT	Likelihood Ratio Test
MGA	Multiscale Geometric Analysis
MSF	Matched Subspace Filter
MSI	Multispectral Imaging
MVDR	Minimum Variance Distortionless Response
PC	Principal Component
PCA	Principal Component Analysis
QDA	Quadratic Discriminant Analysis
RGB	Red, Green, Blue
ROLP	Ratio of Low-Pass
SAM	Spectral Angle Map
SMF	Simple Matched Filter

Chapter 1

Introduction

Multispectral imaging (MSI) has a decades-long history in the remote sensing field to observe the earth from space. MSI is now being applied to the study of ancient artifacts and documents. MSI captures views of an object at multiple narrow-band wavelengths which result in large image data "cubes." A typical MSI image cube is shown in Fig. 1.1. These image cubes may be processed using statistical signal and/or image processing techniques to provide a variety of image enhancement outcomes. This thesis explores the utility of selected signal processing algorithms applicable to MSI document images in four areas: image fusion, matched filter detection, bleed-through removal, and shadow removal.

Typically, previous MSI work has reviewed the multispectral images and selected one monospectral image for analysis. This method has given promising results on the Dead Sea scrolls. It was found that imaging documents in the infrared enabled the text to be viewed

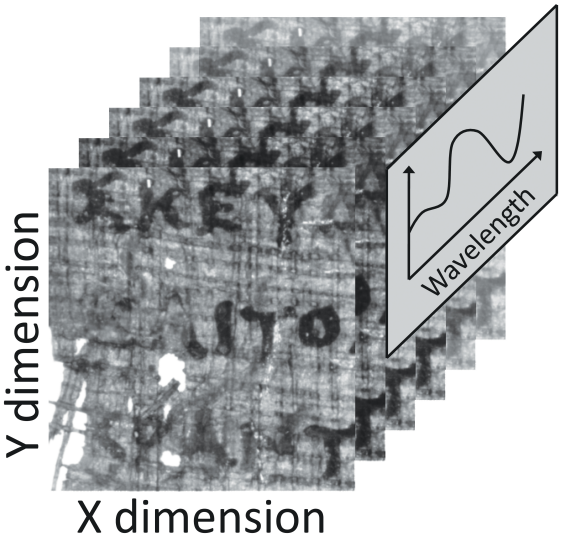
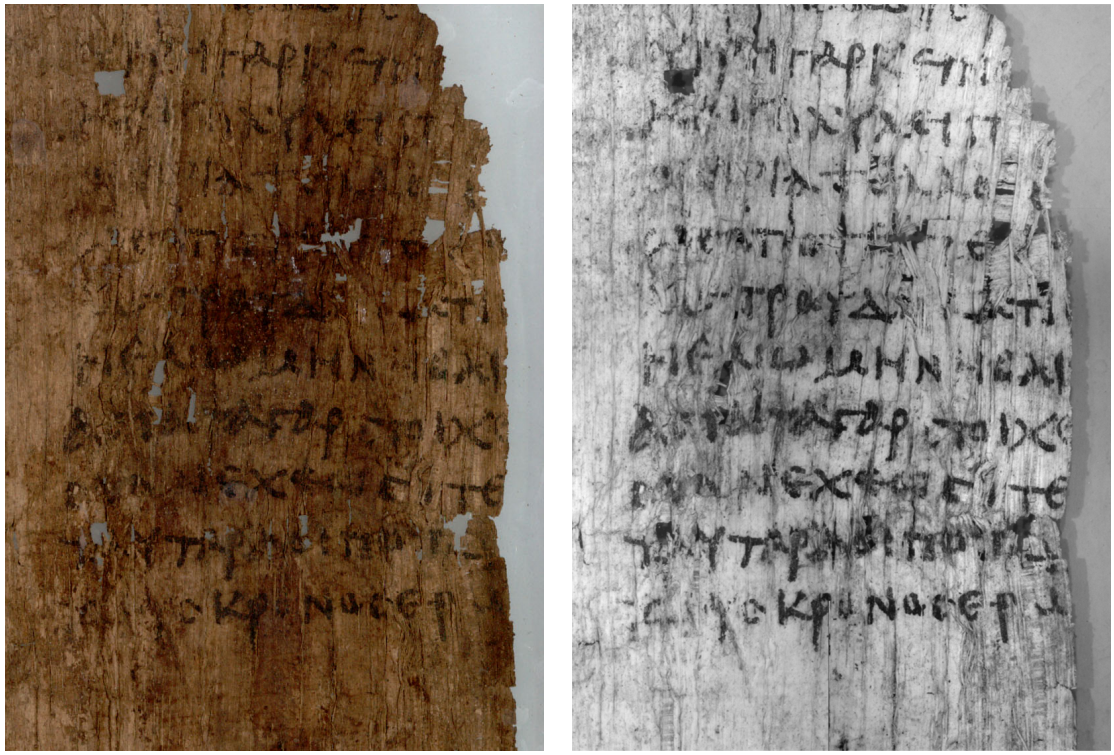


Fig. 1.1: Image cube representation of multispectral data [1].

with greater contrast between ink and parchment [2]. Infrared wavelengths have also been shown to enhance ancient murals, ceramics, and papyri by Ware *et al.* [3,4]. More recently an effort has been made to utilize the full MSI data cube to read the erased Archimedes Palimpsest [5]. Figure 1.2 illustrates the improvement infrared monospectral images have on the readability of ancient text. Digital signal processing techniques generally have not, however, been applied to enhance the usability of the information provided by MSI applied to documents.

MSI has allowed previously indistinguishable ink, and other details, to be viewed and read. Each image provides the observer with additional information which can be used to decipher the text. Ideally, the photographer would capture images at as many wavelengths as possible, but as the number of images increases, it becomes more difficult to store and utilize all the information. Those studying the documents typically desire all the information



(a) Original Color (b) 950 nm

Fig. 1.2: Oxyrhynchus papyrus fragment 1822 original color photograph [6] and 950 nm MSI image [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

available, but would much rather have a few images to work with than many. Each of the four areas covered in this thesis help to consolidate the MSI data into a single image that accurately displays the information found in all of the MSI images.

Overview

Each of the topics discussed in this thesis will be presented in its own chapter. The individual chapters will also contain a literature review of the respective topics in place of a comprehensive review here. MATLAB [8] code for each of the described methods is provided in Appendix A.

Chapter 2 describes the data set that is used for this thesis, including the equipment used to acquire the multispectral imagery and the format of the data. Registration and vignetting correction pre-processing steps are also described.

Chapter 3 discusses image fusion methods that can be used to combine information from multiple images into a single image. Fusion is performed by taking the values at each pixel location within the input images and combining them into a single pixel value. The fusion process can also be performed at the pixel, feature, and image levels. Techniques used for fusion at each of these levels are addressed, including image blending techniques, principal component analysis (PCA), wavelet transforms, contrast pyramid, and color composites. Test results for each method are provided and compared.

Chapter 4 describes how matched filters can be used to identify specific characteristics within the data and create a single image for analysis. Matched filtering utilizes the spectral information within the MSI image cube by viewing each pixel in the wavelength direction. The output of the filter has one value for each pixel location, thus providing a means of reducing the data to a single image. The matched filter can also be used to enhance the information by selecting matched filters with the desired characteristics. Several different methods for implementing a matched filter are derived. A discussion is given on matched filter optimization as applied to document analysis and the resulting filter equivalence.

Chapter 5 describes a new PCA-based method to remove bleed-through text when MSI data is available from both sides of the document. This method is compared to independent

component analysis (ICA) and symmetric orthogonalization, two similar methods recently proposed in the literature. This new method involves performing PCA on a stacked image cube of registered recto and verso images followed by a subtraction of the first two principal components. The method is tested on two documents, one ancient and one modern, with good results.

Chapter 6 addresses the issue of removing shadows present in the MSI images. Several of the documents used in this thesis were encased in glass during the imaging process and have substantial shadows present. A method utilizing quadratic discriminant analysis (QDA) is presented to remove the shadows.

Chapter 7 is a summary of the four topic areas. A discussion is given on the relative merit of these methods when applied to various papyrus types.

Chapter 2

Data Acquisition

This thesis research studied multiple fragments of text found on papyri and parchment. Each of the fragments are housed in the Sackler Library at the University of Oxford. All imaging was done by a team from Brigham Young University in October, 2006.[1]

2.1 Multispectral Imaging Technology

Multispectral images may be obtained by capturing the light reflected from a subject within a narrow bandwidth of light. There are currently two main ways to do this: (1) light the subject with white light encompassing the entire light spectrum of interest with a filter in front of the lens to block all but a specific narrow bandwidth, or (2) illuminate the subject with LED lights that only emit a specific narrow bandwidth of light. In this thesis, all of the images were captured using the first strategy.

The equipment used to capture the images included a digital camera, filter wheel, lens, *xy*-positioning table, lights, and a control computer. The specific details of each of these components are discussed below as reported by Ware [9]. Components of the MSI system are shown in Figs. 2.1, 2.2, and 2.3.

Camera: A Kodak 4.2i Megaplus scientific-grade digital camera, with a class-1 array of 2033×2044 active pixels, captured the images. It is operated in a 10-bit mode resulting in images with 1024 grayscale levels. All functions of the camera were controlled by the system's computer.

Filter Wheel: A filter wheel capable of holding 14 two-inch filters was mounted on the front of the Kodak Megaplus camera. Ten 40 nm interference (Schott) filters ranging from 450 nm to 1000 nm were used.

[1]The BYU team included Stephen M. Bay, Roger T. Macfarlane, Gene A. Ware, and Thomas A. Wayment.

Fig. 2.1: MSI filter wheel.

Lens: A standard Tamron 180 mm macro lens with an aperture nominally set at f3.5 was used.

XY-Positioning Table: When documents are too large for a single image at the desired resolution, it is necessary to image the document in sections. Rather than move the camera, an xy-positioning table was used to move the documents under the camera. The table is controlled by the system's computer, which accurately moves the table to the desired location. A maximum movement of 40 cm in the x-direction and 60 cm in the y-direction is possible.

System Computer: The system's computer is an IBM Thinkpad A30 mounted on a docking station. The docking station contains a PCI bus with an EDT camera interface card connected to the Kodak Megaplus camera. MSI-Acquire, a custom software application, is used to control data acquisition, display, storage, and filter-wheel position.

Lights: Lighting was provided by four PAG PowerArc lights manufactured by PAG Limited of London. The lights produce a color temperature of 5500 K and consume about 32 W

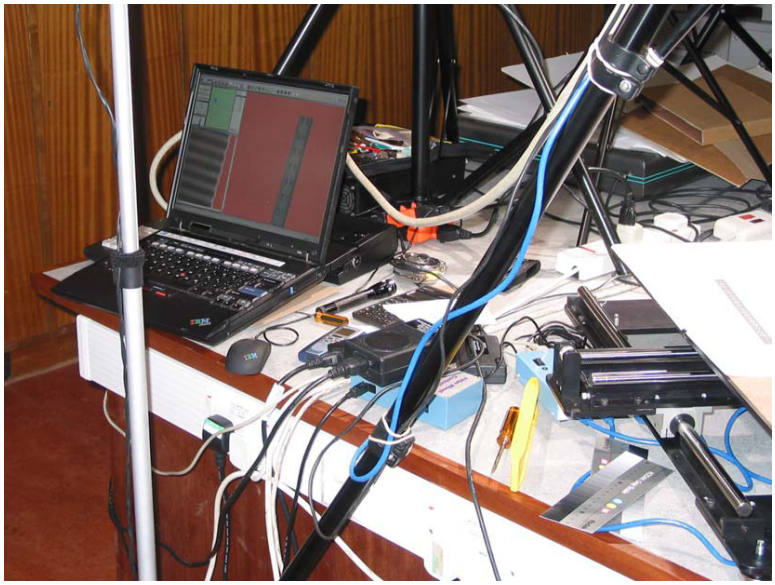


Fig. 2.2: MSI system computer setup.

of power each. The equivalent light brightness of each PAG light at visible wavelengths is approximately the same as an unfocused 300 W quartz-halogen lamp.

2.2 Multispectral Image Data

Each image captured through the 40 nm filters records the grayscale reflectivity at the wavelength selected by the filter. The saturation levels of the images were closely controlled to avoid any saturation of pixels, either black or white, in significant areas of the image. Usually, saturation levels were set so that the darkest/lightest pixels were within approximately 10% of the minimum/maximum value.

Each image, recorded at 10-bits/pixel, is saved as a 16-bit TIFF file resulting in a file size of 8,129 kB per image. In the past, 16-bit images were not supported by many image processing software packages, but, with the advances in technology, 16-bit images are now commonplace.

The metadata information for each image is stored in the file name which is divided into six fields of information: project identification, item identification, xy-position, camera

Fig. 2.3: MSI equipment setup at the Sackler Library, 2006.

parameters, date and time stamp, and lens information. This provides all of the imaging metadata without the use of TIFF tags within the image file header. For example,

$$Sackler06+POxy1822-r+x00y01+750(40)e0045g00b-0200+d060619t191803+f35T180$$

is the file name for the recto side of Oxyrhynchus fragment 1822 imaged during the Sackler 2006 imaging project at X-position 00 and Y-position 01 with a 40 nm bandwidth filter centered at 750 nm, exposure of 45 ms, gain of 0 dB, black level offset of -200, and taken on June 19, 2006 at 7:18:03 P.M.

2.3 Test Set Permissions

Eleven different text fragments were used as a test set for the discussed algorithms in this thesis. Ten are part of the Oxyrhynchus Papyri collection [7] and one is part of the Derveni papyri collection [1].

The Oxyrhynchus collection is housed at the University of Oxford in the Sackler Library. This collection is owned by the Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society. These images are made available for research on enhancement

techniques and permission has been granted to show a select few of them in this thesis as examples of results.

The Derveni collection was imaged by Gene A. Ware in conjunction with the Ancient Textual Imaging Group at Brigham Young University. This collection was imaged at the Thessaloniki Archaeological Museum, Thessalonike, Greece, October 2-10, 2006.

All images used in this thesis are copyrighted and used with permission.

2.4 Pre-Processing

In order to analyze the spectral characteristics of MSI data, certain initial processing steps must be performed. The most important of these steps is image registration, which aligns the individual images so that each pixel location in the images corresponds to the exact same subject location. Image registration has been actively studied and presented in the literature, resulting in very accurate and complex algorithms utilizing supervised and unsupervised techniques. This thesis utilized a common approach that employs marked control points on identical features in each of the images. These control points are then used to define a transformation matrix which may be linear or nonlinear. The MATLAB [8] image processing toolbox functions designed to perform the operation were used to place the control points and perform the registration. With this thesis' high resolution images, the results were sufficiently accurate.

Optical vignetting is another issue that should be corrected with the MSI images. Vignetting is an edge darkening effect caused by optical lenses and other factors. As light enters the camera lens, more light hits the center of the opening than the edges, causing the edges of the image to become darker than the center. Vignetting correction (or flat-fielding) is particularly important when working with the MSI data used in this thesis. As the wavelengths get longer, the effect becomes stronger and more pronounced. The simplest method for removing the vignetting effect is to use a vignetting calibration frame. This is an image of a flat white background at each wavelength used for acquisition. Figure 2.4 is such a calibration frame taken at 650 nm. This calibration frame is a representation of the vignetting curve representing the amount of darkness the camera will add to the

actual value of the subject at each pixel location. Removing the vignetting is achieved by normalizing and inverting the calibration frame followed by multiplication times the data images. The steps for this are:

1. Divide by the maximum value of the calibration frame,
2. Divide one by the new calibration frame,
3. Multiply the calibration frame and image data pixel by pixel.

Figure 2.5 shows an image with strong vignetting present and the effect of vigetting correction using the calibration frame from Fig. 2.4.

Vignetting correction should be performed on the raw images before image registration transformations are performed. After transformations are performed, the calibration frame is no longer aligned to the image. Imperfections may also be present in the calibration frame and should be removed using interpolation to ensure that the vignetting curve is smooth. If the imperfections are not removed, they will be transferred to the corrected data as well. Vignetting corrected images should appear flat with the corners at the same brightness as the center.

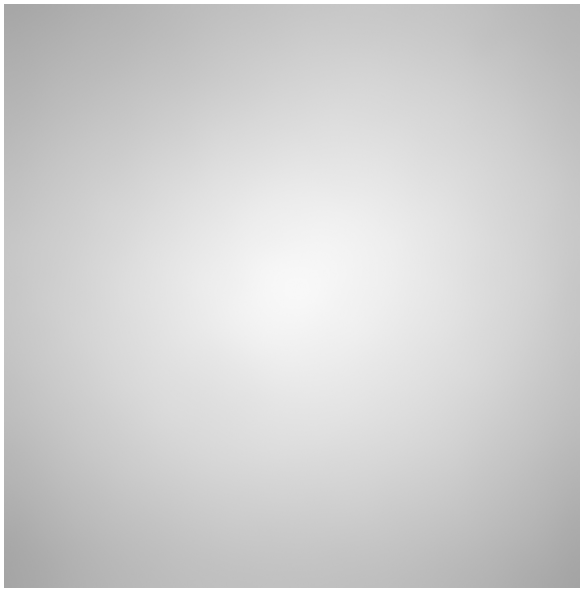
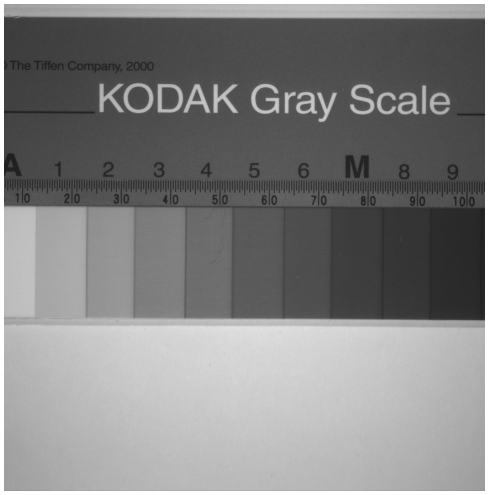
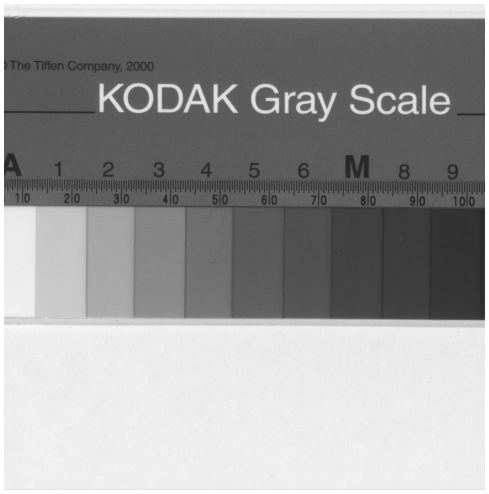


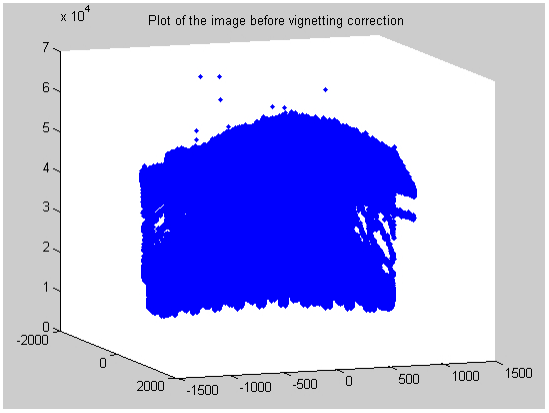
Fig. 2.4: Vignetting calibration frame.



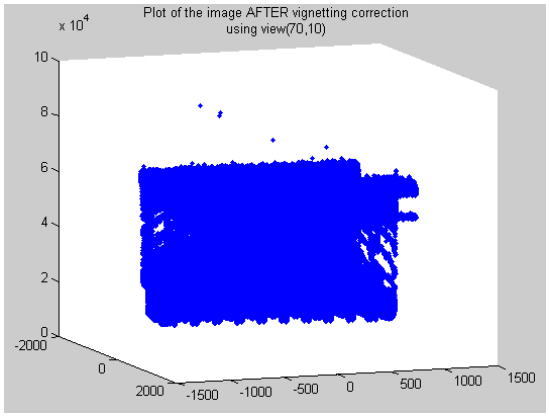
(a) Original image



(b) Corrected image



(c) Plot of original pixel values



(d) Plot of corrected pixel values

Fig. 2.5: Original and vignetting corrected images.

Chapter 3

Image Fusion

Image fusion is a broad term applied to algorithms which combine the information of two or more images into a single image by selecting the best information at the pixel level. Numerous reviews of image fusion have been written to categorize the breadth of methods used [10, 11]. Fusion is very useful in scientific fields such as remote sensing and medical imaging. Often the need for image fusion arises in document analysis using multispectral imaging (MSI) technology. Experts studying documents gain significant information from the multispectral views, but often desire a single image that contains all of the information they deem important. Image fusion may be used for this task.

Image and data fusion has been used in a number of other fields to enhance data for analysis. For example, it has been used to combine images from multiple sensors and resolutions as with remote sensing [10, 12–14] and medical applications [15–17]. It has also been used in the photography field to combine images with multiple depths-of-field and exposures [18, 19]. In surveillance and military operations, as well as remote sensing, fusion has been used to combine images from visual and thermal infrared sensors [20–22].

Image fusion can be performed at three processing levels:

1. Pixel,
2. Feature,
3. Image Level.

Image fusion at the pixel level is performed using the raw pixel data. Feature-level fusion first extracts the features of the input images and then performs the fusion. Fusion at the image level uniformly combines entire images. Each of these fusion levels are performed

using a fusion rule, which determines which input image has the best information for each given region of interest. The accuracy of the result at any fusion level is directly related to the fusion rule used.

3.1 Pixel Level

Pixel-level image fusion has been researched extensively over the past several decades and continues to be an area of interest. Techniques utilizing pixel-level fusion include arithmetic blending, principal component analysis (PCA), wavelets, filtering, hidden Markov models (HMM) [23], multiscale geometric analysis (MGA) [16], independent component analysis (ICA) [24], statistical methods [25], and many more. Often these techniques are used together and may even include fusion rules at the feature and image level as well.

3.1.1 Image Blending

The simplest pixel-level fusion technique is arithmetic blending. This method is not commonly found in the literature under image fusion techniques, but is a known method in image processing. Many have been implemented in popular software packages, such as Adobe Photoshop [26], to accomplish layer blending (a term synonymous with fusion). Image fusion may be performed by some common blending modes, taken from Gruschel [27] as follows:

Average: average of the values at each pixel location,

Difference: subtract one image from the other,

Multiply: multiply the values at each pixel,

Screen: invert and then multiply values at each pixel together,

Overlay: use the multiply method for dark pixels and the screen method for light pixels (light/dark pixels are more/less than half the highest possible value),

Hard Light: use the multiply method for dark pixels and invert the result of the screen method for light pixels,

Darken: darkest value at each pixel,

Lighten: lightest value at each pixel,

Color Dodge: divide the first image by the inverse of the second.

Many of these methods are performed using simple equations that can be applied to the entire image and might more correctly be categorized as fusion at the image level, but others operate at the pixel level. To maintain uniformity they are grouped together here.

The simplest of the blending methods is to average, add, subtract, or multiply the images together. Lighten and darken methods use logic to select the pixel that is lightest or darkest at each pixel location. Overlay, color dodge, hard light, and screen use slightly more complicated equations to combine the images [27].

Overlay, a combination of screen and multiply, is particularly useful. The overlay operation uses two equations that are selected by pixel values as

$$\begin{aligned} IF(A_{i,j} > 127.5): \;\; F_{i,j} &= 255 - \frac{(255 - A_{i,j}) \times (255 - B_{i,j})}{127.5}, \\ ELSE: \;\; F_{i,j} &= A_{i,j} \times B_{i,j}, \end{aligned} \tag{3.1}$$

where B is the base image, A is the image being blended into the base, $_{i,j}$ indicates the pixel coordinates, and 255 is the maximum possible pixel value (8 bits).

Another method used by Constable and Spencer [20] for medical imaging is to use the square root of the sum of the squares (rms), written

$$F_{i,j} = \sqrt{(\hat{A}_{i,j})^2 + (\hat{B}_{i,j})^2}, \tag{3.2}$$

where the hat ($\hat{}$) indicates that the mean has been removed from A and B. This tends to average the extreme lights and darks in an image without taking into account the average background values.

3.1.2 Weighted Combinations

The image blending operations described above are defined for only two input images. When it is necessary to fuse additional images, they can be combined in pairs or possibly in larger image groups. When linear combinations of multiple images are desired, a specific weight can be given to each of the images and then combined using

$$F = \sum_{i=1}^{N} w_i U_i, \tag{3.3}$$

where F is the fused image, i is the image index, N is the number of input images, U_i is the ith input image, and w_i is the weight of image U_i. This is a simple operation, but determining the optimal weights is difficult without prior knowledge. If all weights were equal to $1/N$ the result would be an average of the input images.

3.1.3 Principal Component Analysis

Principal Component Analysis (PCA) is a means of defining the weights for combining the input images using an eigenvalue decomposition. First, the input images are reshaped into vectors and placed as columns in the matrix X. Then the eigenvectors of $X^T X$ provide an orthogonal basis along the directions of maximum variance. Use of the eigenvector associated with the largest eigenvalue provides a weighted combination of the input images that contains the maximum amount of the variance in the combined image. This technique was used by Zheng *et al.* [28] to combine wavelet transform coefficients, but is also valid for combining entire images.

PCA is more commonly used as a transformation technique to reduce the dimensions of MSI data. In MSI document analysis, this serves as a method for fusing the spectral information from the data into fewer images. These images can then be further combined using the image-level fusion techniques [29]. Since the majority of the variance is usually contained within the first three principal component (PC) images, PCA color visualizations are particularly useful.

Substitution methods can be used with PCA by transforming the input images into the PCA domain, substituting an outside image for one of the PC images, and then performing the inverse transform. This is most commonly used for fusing multi-resolution images by substituting the high resolution image for the first principal component [30, 31].

3.1.4 Wavelet Transform

The wavelet transform has become the most popular transform within a family of tower-shaped multi-scale transforms which includes the pyramid decomposition. Wavelets, as used in fusion, decompose the input image cube into high and low frequency details. These components are then fused and transformed back into a single output image. Many excellent reviews of the two-dimensional wavelet transformation are available in the literature [19, 32–35] and will not be given here. Also, various wavelet families, or mother wavelets, are available. In the results section of this chapter, the Haar and DBSS wavelets are shown to yield similar results.

Along with selecting which wavelet family to use, a fusion rule must also be chosen. Multiple fusion rules have been used in the literature over the course of the past three decades and all seem to be data-specific. Common fusion rules are absolute minimum/maximum, average, addition, subtraction, and substitution. Different fusion rules can also be used for combining the high and low frequency details [17]. Area-based fusion rules are used to minimize the effect of noise on the fusion process [35]. An excellent review of fusion rules and concepts is given by Zhang and Blum [36].

An image fusion toolbox written by Rockinger [37] contains functions to perform wavelet fusion as well as the contrast and Laplacian pyramid methods. It allows for various fusion rules to be selected and which mother wavelet to use. This toolbox was used for the results shown at the end of this chapter.

3.1.5 Contrast Pyramid

Contrast and Laplacian pyramids are very similar and both have been used for image fusion. They are also referred to as the Difference of Low-Pass (DOLP) pyramid and

the Ratio of Low-Pass (ROLP) pyramid, respectively. These names help to clarify the distinction between the two. A description of the Laplacian pyramid will be presented first, followed by the contrast pyramid. The description of these pyramids comes primarily from Toet *et al.* [38, 39].

Each level in the Laplacian pyramid is obtained by downsampling in both spatial directions followed by low-pass filtering the image at the previous level using a Gaussian-like weighting function. This provides a binocular pyramid where images at each level are halved in size and reduced in detail. A set of bandpass filtered images are then obtained by the difference between the images at each level in the pyramid. Before this difference can be computed, an interpolation is performed to return the smaller image to the same scale as the larger image. The bandpass images of Laplacian pyramids formed from two separate images can then be combined using a fusion rule. After combining the pyramids, a fused image can be reconstructed by inverting the decomposition.

The contrast pyramid is constructed in a similar fashion, but computes a ratio of the images at each level in the pyramid in place of the difference. This ratio determines the contrast of each image based on a common contrast measurement defined as

$$\frac{A - B}{B}, \tag{3.4}$$

where A is the value at a certain pixel and B is the average value surrounding that pixel [38, 40–42]. This term may be written as

$$\frac{A}{B} - 1, \tag{3.5}$$

which gives greater meaning to the contrast pyramid. Contrast images are obtained at each level of the pyramid by computing the ratio of the low-pass filtered image to the original image. The contrast images are then combined and the fused image obtained by inverting the decomposition.

3.2 Feature Level

In theory, features from one image could be directly selected and injected into another image. This is difficult in practice due to the problem of accurately separating the features of interest from the background. Feature-level fusion in document analysis could be used to select characters from one image and place them into another, which is often the exact result desired in the fusion of document images. Since the identification of characters is typically the end goal in document analysis, if the complete characters can be identified by classification, then no further processing would be necessary. The difficulty of performing accurate feature level fusion on images has led to research being devoted mainly to the other two fusion levels.

3.3 Image Level

For MSI fusion, each image is generally the same size and can be combined into a single image using a global fusion rule that operates equally on the entire image. The simplest of these techniques is a color composite.

3.3.1 Color Composites

Three grayscale images may be viewed as the three color bands of an RGB image, which allows the human eye to determine which features are important. This can also be done with two grayscale images by using one of the images as two of the color bands or placing a blank image to fill the missing band. A similar color composite, using four grayscale images, is the CMYK composite which has an additional band to indicate black pixels.

Another color composite technique uses the intensity, hue, and saturation (IHS or HSI) transform. The three grayscale images from an RGB image are transformed into intensity, hue, and saturation images. There are a number of versions of the IHS transformation in the literature [10, 12–14, 31, 43–45] which differ mainly by coordinate systems and hue reference points. A popular version is the cylindrical coordinates method outlined by Jun and Baoming [14] as

$$
\begin{aligned}
I &= \frac{R+G+B}{3}, \\
S &= 1 - \frac{\min(R,G,B)}{I}, \\
H &= \arccos\left[\frac{R - G/2 - B/2}{\sqrt{(R-G)^2 + (R-B)(G-B)}}\right].
\end{aligned} \tag{3.6}
$$

A linear version of the IHS transform is described by Pohl and Van Genderen [10] as

$$
\begin{aligned}
\begin{bmatrix} I \\ v_1 \\ v_2 \end{bmatrix} &= \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \\
H &= \tan^{-1}\left(\frac{v_1}{v_2}\right), \\
S &= \sqrt{v_1^2 + v_2^2}.
\end{aligned} \tag{3.7}
$$

Further enhancement can be obtained by exchanging any one of the IHS images for another image. This is most commonly accomplished by exchanging the intensity image for one containing a better representation of the data, or from an entirely different data set. Inverting the transform back to the RGB space creates a fused image.

The IHS transform has also been useful in visualizing the first three PCA components. Intensity, hue, and saturation calculated by the IHS transform are orthogonal to each other, just as the PCA components. By interpreting the PC images as being the IHS images and performing an inverse transform, a set of fused images can be obtained and easily visualized [29].

3.3.2 Checkerboard

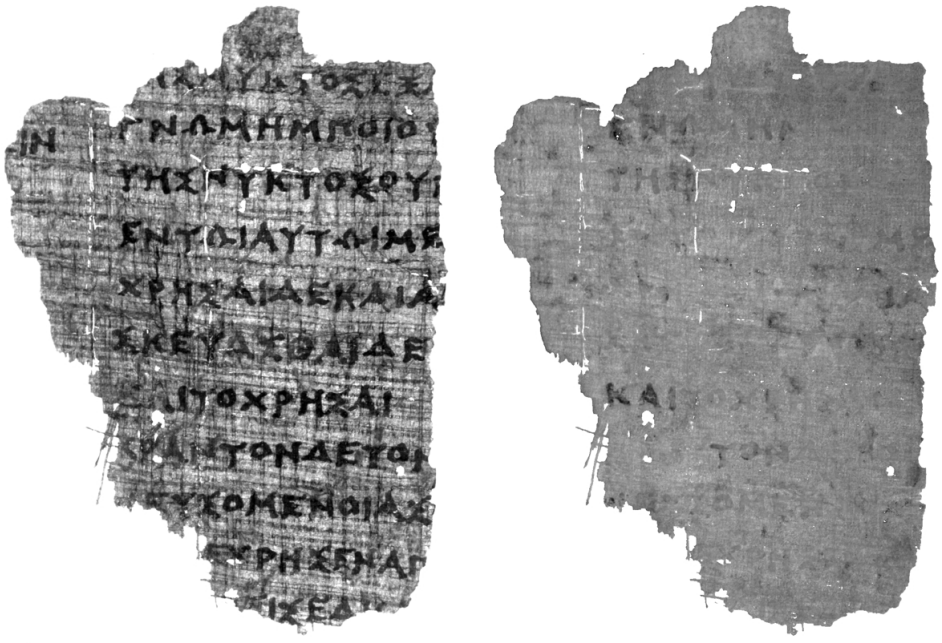
Another simple visualization technique is to create an image with alternate pixels coming from each of the two input images. This shows both images together and allows the eye to extract the important features contained in each image. This has negative effects on contrast as well as image resolution. For an overall view of an image, the results are generally good, but detailed analysis is impaired by the added visual noise.

3.4 Examples

A case study of image fusion techniques was performed on the Derveni fragment C4 [1]. This fragment has a theorized thin-film effect causing most ink characters to show up in the first PC image and the remainder in the second. Figure 3.1 displays the two PC images that are to be fused. Fusion results using image blending techniques are compared in Fig. 3.2. Wavelet and pyramid fusion examples are shown in Fig. 3.3. Color composite techniques are shown in Fig. 3.4. It appears that minimum, overlay, and root sum give the best results from the blending methods illustrated in Fig. 3.2. The contrast pyramid tends to be the best of the wavelet and pyramid fusion techniques and is similar to the blending techniques. Each color composite method is significantly different from the others. Deciding which method is best depends upon the application.

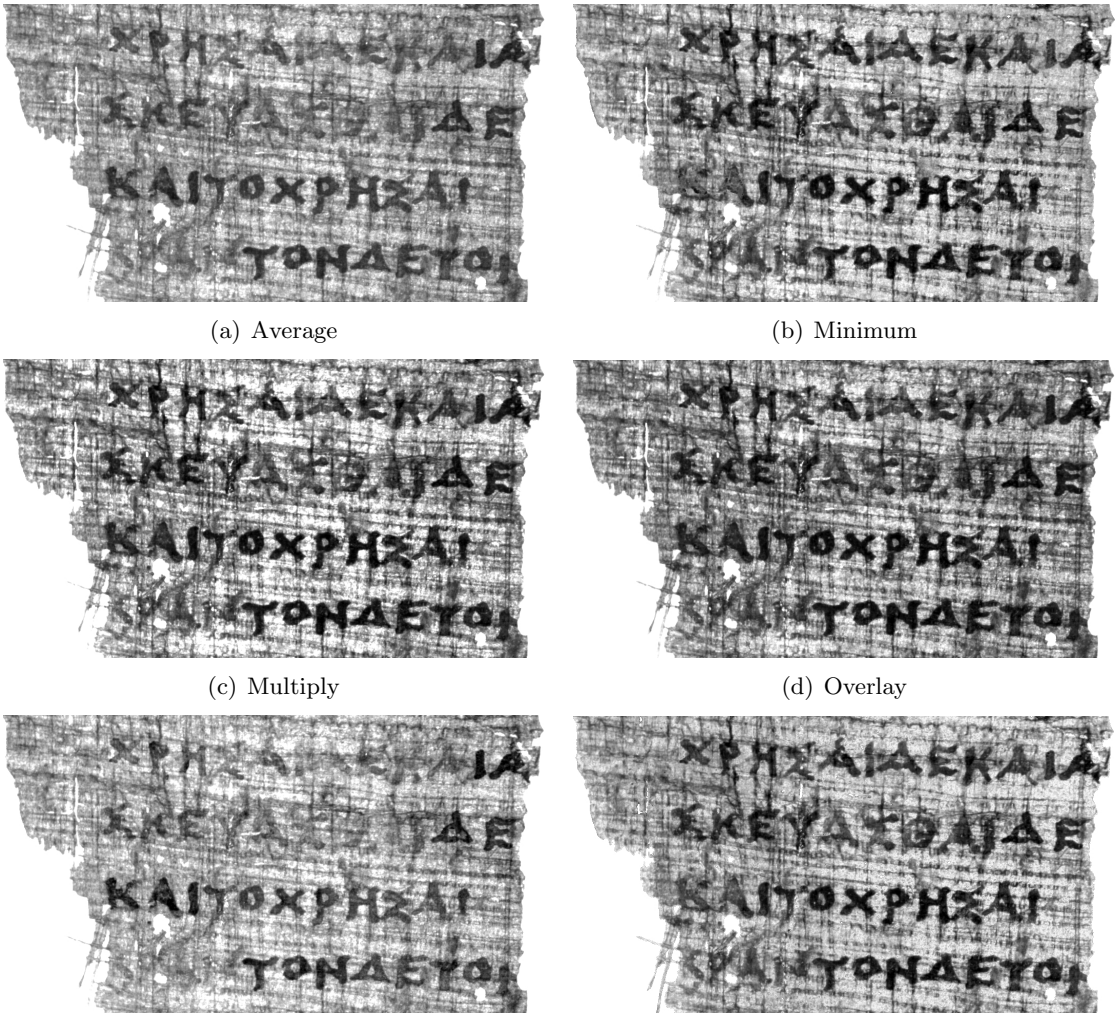
3.5 Conclusion

Image blending methods are able to combine the important features of the two input images without losing clarity or introducing noise. The overlay and root sum methods



(a) First (b) Second

Fig. 3.1: PCA component images to be fused [1].



(a) Average (b) Minimum

(c) Multiply (d) Overlay

(e) Hard Light (f) Root Sum

Fig. 3.2: A comparison of image blending techniques [1].

tend to outperform other methods. PCA is a very effective method for combining many images into three or less. Pyramid techniques, including wavelets, were able to fuse the two images. All of the tested pyramid techniques returned similar results, but the contrast pyramid tended to have better contrast and clarity in the output image. Color composites were shown to be very effective in visualizing the data and aided in the interpretation of PCA results. The color composites provide a method for viewing the information while still distinguishing the different components.

(a) DBSS Wavelet

(b) Haar Wavelet

(c) Contrast Pyramid

(d) Laplacian Pyramid

Fig. 3.3: Pyramid and wavelet image fusion at decomposition level 7 using an absolute maximum fusion rule for the details and selecting PC 1 as the low-pass fusion rule [1]. These results were created using the Rockinger's image fusion toolbox [37].

(a) RGB

(b) CMYK

(c) IHS

(d) IHS to RGB

Fig. 3.4: Color composite examples using the Derveni fragment C4 [1]: an RGB color composite from PC2,1,1 (a), a CMYK composite from PC1,2,3,1 (b), a composite formed by transforming PC1,2,1 to the IHS color space (c), and a composite formed by transforming PC1,2,3 from the IHS space to RGB (d).

Chapter 4

Matched Filter Detection

Matched filter detection has the same objective as image fusion, but employs a distinctly different method for combining the images. Rather than selecting the images having the finest detail, matched filter detection views the images simultaneously, using the spectral reflectance vector of each pixel. Filters are applied to identify the physical elements by their spectral signatures and then use that information in the creation of the output image.

Matched filters combine information across the spectral bands by maximizing the output value of spectral-vectors similar to a target, and minimizing those resembling background. This is accomplished by multiplying each pixel spectral-vector of the image cube by a vector $\mathbf{q}$, which concentrates the spectral band information into a single value. Matched filters differ in the method used to determine $\mathbf{q}$.

4.1 Models

Two models are frequently used to define the matched filter vector $\mathbf{q}$. The first is a linear additive model that has been used to detect weak gas plumes in hyperspectral data. It is defined as

$$\mathbf{x} = \epsilon\mathbf{s} + \mathbf{w}, \tag{4.1}$$

where $\mathbf{x}$ is the observed spectral vector of a given pixel, $\mathbf{s}$ is the target spectrum, $\mathbf{w}$ is the background spectrum, and ϵ is the strength of the target. Each pixel in this model is a measure of the ink spectrum added to the background. With ink, the ϵ weighting will either be very high or very low, indicating ink is present or absent, respectively. The additive model is better designed for occasions where the ink is semi-translucent and the spectrum of the background is truly adding to that of the ink in the observed pixel. It is an effective model that can be applied to multispectral document data.

The second model describes each pixel as a mixture of possible ink and background spectrums as follows:

$$\mathbf{X} = \mathbf{HA}, \tag{4.2}$$

where $\mathbf{H}$ is a matrix with columns representing known ink/background spectrums and $\mathbf{A}$ is referred to as an abundance matrix which defines the weights of each column in $\mathbf{H}$ for each pixel (column) in $\mathbf{X}$. This model has more theoretical appeal for document analysis than the additive model, since more than one type of ink or various levels of ink degradation are possible. It is different from the first model in that the focus is not to minimize/maximize the ink/background inputs, but rather, to estimate the percentage of each input spectrum present in each pixel. In document analysis, it may be more common to think of this problem as the determination of which input spectrum is most likely the one present, assuming there may be no mixing at all.

These two models depend on *a priori* knowledge of the target spectrum. The first uses a single constant target spectrum while the second uses multiple target spectra in a matrix. Each must be provided by the user or some other pre-processing step.

In matched filter detectors, it is desired to maximize $\epsilon\mathbf{s}$ while minimizing the influence of $\mathbf{w}$ by multiplying by some vector $\mathbf{q}$. This is represented by

$$\mathbf{q}^T\mathbf{x} = \epsilon\mathbf{q}^T\mathbf{s} + \mathbf{q}^T\mathbf{w} \tag{4.3}$$

in the case of (4.1). When the background is structured, it is often desirable to multiply by a matrix $\mathbf{Q}$ to incorporate the structural information, in which case we get

$$\mathbf{Qx} = \epsilon\mathbf{Qs} + \mathbf{Qw}. \tag{4.4}$$

The matched filter detector is then chosen so that $\left|\mathbf{q}^T\mathbf{w}\right|$ or $\|\mathbf{Qw}\|$ are as close to zero as possible. It is desired to find a vector $\mathbf{q}$ which will maximize the dot product with ink spectrums while simultaneously minimizing the dot product with the background spectrum.

This can be expressed as

$$\max_{\mathbf{q}} \left\|\mathbf{q}^T\mathbf{s}\right\|_2 \text{ subject to } \min_{\mathbf{q}} \|\mathbf{q}\mathbf{w}\|_2 . \tag{4.5}$$

4.2 Derivations

Matched filters are typically referred to by acronyms. Table 4.1 gives a description of the acronyms used in this chapter. A list of equations for thirteen matched filters, taken from Foy [46], Manolakis [47], Chang [48], and Hastie *et al.* [49], is provided in Table 4.2. These equations will be referenced throughout this section by name, acronym, and/or by filter number in Table 4.2. While Table 4.2 is sorted by where the matched filters are cited, this section orders the derivations of the matched filters by complexity.

4.2.1 Simple Matched Filter (SMF)

The simplest method for determining $\mathbf{q}$ in (4.3) is to use a known ink spectrum. Using the ink spectrum $\mathbf{s}$ as $\mathbf{q}$ results in the SMF detector (Table 4.2, filter 1). This tends to be greatly affected by the length or energy (defined as $\sqrt{\mathbf{x}^T\mathbf{x}}$) of the background due to the extreme difference in energy between the commonly dark ink and light background. One way to remove this effect is to divide the result by the energy of the background. Also, dividing by the energy of $\mathbf{s}$ will measure the multi-dimensional angle between $\mathbf{s}$ and $\mathbf{x}$, which is called the SAM detector (Table 4.2, filter 2).

4.2.2 Clutter Matched Filter (CMF)

Some detectors are derived using a Likelihood Ratio Test (LRT) of the probability densities for target-absent and target-present pixels. The LRT is expressed as

$$\frac{p_1(\mathbf{x})}{p_0(\mathbf{x})} > \gamma, \tag{4.6}$$

where $p_0(\mathbf{x})$ is the probability density function (pdf) of the background without target, $p_1(\mathbf{x})$ is the pdf of pixels with target, and γ is a threshold value. It is frequently assumed

Table 4.1: Description of the matched filter acronyms.

Acronym	Name
SMF	Simple Matched Filter
SAM	Spectral Angle Map
CMF	Clutter Matched Filter
ACE	Adaptive Coherence Estimator
MSF	Matched Subspace Filter
DMF	Distance Matched Filter
MVDR	Minimum Variance Distortionless Response
CEM	Constrained Energy Minimization
LCMV	Linearly Constrained Minimum Variance
LDA	Linear Discriminant Analysis
QDA	Quadratic Discriminant Analysis
PCA	Principal Component Analysis

Table 4.2: Detection equations divided into four sections from Foy [46], Manolakis [47], Chang [48], and Hastie *et al.* [49], respectively, where $\mathbf{s}$ is the target spectrum, $\mathbf{x}$ is the observed data, $\mathbf{R}^{-1}$ and $\mathbf{\Gamma}^{-1}$ are inverted covariance and correlation matrices of the data with subscripts x, b, and s indicating the observed, background, and target dataset wherever specified, and Λ_x and V_x are the eigenvector and eigenvalue matrices of $\mathbf{R}$.

Filter Number	Name	Equation
1	SMF	$\mathbf{s}^T\mathbf{x}$
2	SAM	$\frac{\mathbf{s}^T\mathbf{x}}{\sqrt{\mathbf{s}^T\mathbf{s}}\sqrt{\mathbf{x}^T\mathbf{x}}}$
3	CMF	$\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}$
4	ACE	$\frac{\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}}{\sqrt{\mathbf{s}^T\mathbf{R}^{-1}\mathbf{s}}\sqrt{\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}}}$
5	MSF	$\mathbf{x}^T\mathbf{H}\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{x}$
6	t-statistic	$\frac{\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}}{\sqrt{\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}-\rho^2\left(\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}\right)^2}}\rho\sqrt{d-1}$
7	F-statistic	$\frac{\left(\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}\right)^2}{\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}-\rho^2\left(\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}\right)^2}\rho^2(d-1)$
8	Quadratic detector	$\left(\mathbf{x}-\boldsymbol{\mu}_b\right)^T\mathbf{R}_b^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_b\right)-\left(\mathbf{x}-\boldsymbol{\mu}_s\right)^T\mathbf{R}_s^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_s\right)$
9	DMF	$\frac{\left(\boldsymbol{\mu}_s-\boldsymbol{\mu}_b\right)^T\mathbf{R}_b^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_b\right)}{\left(\boldsymbol{\mu}_s-\boldsymbol{\mu}_b\right)^T\mathbf{R}_b^{-1}\left(\boldsymbol{\mu}_s-\boldsymbol{\mu}_b\right)}$
10	RX Anomaly Detection	$\left(\mathbf{x}-\boldsymbol{\mu}_b\right)^T\mathbf{R}_b^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_b\right)$
11	MVDR	$\frac{\left(\mathbf{s}-\boldsymbol{\mu}_x\right)^T\mathbf{R}_x^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_x\right)}{\left(\mathbf{s}-\boldsymbol{\mu}_x\right)^T\mathbf{R}_x^{-1}\left(\mathbf{s}-\boldsymbol{\mu}_x\right)}$
12	CEM	$\frac{\mathbf{s}^T\mathbf{\Gamma}_x^{-1}\mathbf{x}}{\mathbf{s}^T\mathbf{\Gamma}_x^{-1}\mathbf{s}}$
13	LCMV	$\mathbf{x}^T\mathbf{\Gamma}^{-1}\mathbf{U}\left(\mathbf{U}^T\mathbf{\Gamma}^{-1}\mathbf{U}\right)^{-1}\mathbf{c}$
14	LDA	$\left(\mathbf{x}^T\mathbf{R}^{-1}\mathbf{s}-\frac{1}{2}\mathbf{b}^T\mathbf{R}^{-1}\mathbf{b}\right)-\left(\mathbf{x}^T\mathbf{R}^{-1}\mathbf{b}-\frac{1}{2}\mathbf{s}^T\mathbf{R}^{-1}\mathbf{s}\right)$
15	QDA	$-\frac{1}{2}\log\lvert\mathbf{R}_s\rvert-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_s\right)^T\mathbf{R}_s^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_s\right)$
16	PCA	$\Lambda_x^{-1/2}V_x^T\mathbf{x}$

that the background is normally distributed. Under this assumption $p_0(\mathbf{x})$ and $p_1(\mathbf{x})$ are distributed as

$$\begin{aligned} p_1(\mathbf{x}) &\sim \exp\left[-\frac{1}{2}\left(\mathbf{x}-\epsilon\mathbf{s}\right)^T\mathbf{R}^{-1}\left(\mathbf{x}-\epsilon\mathbf{s}\right)\right], \\ p_0(\mathbf{x}) &\sim \exp\left[-\frac{1}{2}\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}\right]. \end{aligned} \tag{4.7}$$

Absorbing the logarithm and common terms into the threshold γ reduces (4.6) to

$$\left(\mathbf{x}-\epsilon\mathbf{s}\right)^T\mathbf{R}^{-1}\left(\mathbf{x}-\epsilon\mathbf{s}\right) - \mathbf{x}^T\mathbf{R}^{-1}\mathbf{x} > \gamma, \tag{4.8}$$

which is the same as the Quadratic detector (Table 4.2, filter 8). Writing out the quadratic term and combining terms results on the left side of the inequality in (4.8), labeled V here, yields

$$V = \epsilon^2\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x} - 2\epsilon\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}. \tag{4.9}$$

Maximizing V with respect to ϵ finds the V which is closest to the threshold γ. This is found by setting the derivative of V with respect to ϵ equal to zero, resulting in

$$\epsilon = \frac{\mathbf{s}^T\mathbf{R}^{-1}\mathbf{x}}{\mathbf{s}^T\mathbf{R}^{-1}\mathbf{s}}, \tag{4.10}$$

which is the CMF detector (Table 4.2, filter 3).

4.2.3 Matched Subspace Filter (MSF)

When the k target spectrums $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k$ are known, the matrix $\mathbf{H} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k]$, formed by stacking them as columns, spans the target subspace. The MSF detector (Table 4.2, filter 5) is a measurement of the energy of $\mathbf{x}$ projected into the target space using $\mathbf{x}^T\mathbf{P}_\mathbf{H}\mathbf{x}$, where

$$\mathbf{P}_\mathbf{H} = \mathbf{H}\left(\mathbf{H}^T\mathbf{H}\right)\mathbf{H}^T \tag{4.11}$$

is the orthogonal projection matrix.

4.2.4 Linearly Constrained Minimum Variance (LCMV)

The LCMV detector (Table 4.2, filter 13) is similar to the MSF but utilizes the j known background spectra $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_j$ to suppress unwanted characteristics. Let

$$\mathbf{U} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_j] \tag{4.12}$$

be a matrix containing the known target, $\mathbf{t}_1$ to $\mathbf{t}_k$, and background, $\mathbf{b}_1$ to $\mathbf{b}_j$, spectral vectors. In this case the desire is to minimize the filter output energy while constraining $U^T\mathbf{x} = \mathbf{c}$, where $\mathbf{c}$ is a constant vector containing ones from subscripts 1 to k and zeros from subscripts $k+1$ to $k+j$. Let y_i be the output of the detector,

$$y_i = \sum_{j=1}^{L} \mathbf{q}_j \mathbf{x}_{ij} = \mathbf{q}^T \mathbf{x}_i = \mathbf{x}_i^T \mathbf{q}, \tag{4.13}$$

where L is the number of input images (spectral bands). The average energy of y_i is given by

$$\frac{1}{N} \sum_{i=1}^{N} y_i^2 = \frac{1}{N} \sum_{i=1}^{N} \left(\mathbf{x}_i^T \mathbf{q}\right)^T \left(\mathbf{x}_i^T \mathbf{q}\right) = \mathbf{q}^T \mathbf{\Gamma} \mathbf{q}, \tag{4.14}$$

with $\mathbf{\Gamma} = \frac{1}{N} \Sigma_{i=1}^{N} \mathbf{x}_i \mathbf{x}_i^T$ being the autocorrelation matrix. The LCMV matched filter $\mathbf{q}$ can then be determined by

$$\min_{\mathbf{q}} \left\{ \mathbf{q}^T \mathbf{\Gamma} \mathbf{q} \right\} \text{ subject to } \mathbf{U}^T \mathbf{q} = \mathbf{c}. \tag{4.15}$$

The optimal solution to (4.15), as given by Chang [48], is

$$\mathbf{q} = \mathbf{\Gamma}^{-1} \mathbf{U} \left(\mathbf{U}^T \mathbf{\Gamma}^{-1} \mathbf{U} \right)^{-1} \mathbf{c}. \tag{4.16}$$

Combining (4.13) and (4.16) gives the LCMV detector, written as

$$\mathbf{x}^T \mathbf{\Gamma}^{-1} \mathbf{U} \left(\mathbf{U}^T \mathbf{\Gamma}^{-1} \mathbf{U} \right)^{-1} \mathbf{c}. \tag{4.17}$$

4.2.5 Constrained Energy Minimization (CEM)

The CEM detector (Table 4.2, filter 12) derivation is similar to the LCMV, but with only one target spectrum vector, $\mathbf{s}$, in place of the matrix $\mathbf{U}$. In this case, the constraint $\mathbf{c}$ becomes a scalar 1 and the problem is written as

$$\min_{\mathbf{q}} \left\{ \mathbf{q}^T \mathbf{\Gamma} \mathbf{q} \right\} \text{ subject to } \mathbf{s}^T \mathbf{q} = 1, \tag{4.18}$$

with the optimum solution given by

$$\mathbf{q} = \frac{\mathbf{\Gamma}^{-1}\mathbf{s}}{\mathbf{s}^T \mathbf{\Gamma}^{-1} \mathbf{s}}. \tag{4.19}$$

Substituting back into (4.13), the CEM detector is written

$$\frac{\mathbf{s}^T \mathbf{\Gamma}^{-1} \mathbf{x}}{\mathbf{s}^T \mathbf{\Gamma}^{-1} \mathbf{s}}. \tag{4.20}$$

4.2.6 *F*-statistic

The F-statistic (Table 4.2, filter 7) is derived by Foy [46] with references to other methods for derivations as well. Basically, the F-statistic is the ratio of the CMF and the sum of squares due to error. The t-statistic (Table 4.2, filter 6) is then the square root of the F-statistic.

4.3 Observations

Different variations were used to determine the best method for using each of the equations in Table 4.2. A few issues needed to be resolved by trial and error. Each issue arose from observations during initial testing. The issues were:

- Should the mean be removed?
- Should the covariance matrix be determined with cube or background data?
- Should correlation be used instead of covariance?

Before any of these issues can be resolved, the desired characteristics of the best readable image must be determined. As a detection problem, the goal is to find a method that can accurately identify the ink and produce a binary image indicating the ink's location. This, however, is commonly not the expert's desired output. Since much useful information is available in the background, it is desired to have the background available for the reader to infer where ink might have been or to better understand why it is missing. This understanding of the desired output significantly influences the answers to the above questions.

4.3.1 Mean Removal

Almost all of the matched filters require that the data contain a zero mean. This removes any effect that a data offset might have on the matched filter dot product. There are three data means possible: the entire cube, the selected target, and the selected background. Which mean should be removed is a significant question.

In theory, the mean of the background should be removed from all the data to center it, but there is also reason to remove the target mean. Testing showed that in all but four matched filters, the mean of the cube should be removed from both the target spectrum, $\mathbf{s}$, and the cube used to provide $\mathbf{x}$. The exceptions were the SAM and MSF detectors, which do not use a covariance matrix, and the F- and t-statistics. The DMF and MVDR differ only in which mean is removed. Comparing the results of these two filters show that they are the same except in shadow areas, which are not of interest here.

Essentially, these two methods are equivalent, implying that the means of the background and cube are very similar. This indicates that the ink is sparse and has little effect on the data cube's statistics. When mean removal is desired, the cube mean can be used in place of the target mean.

4.3.2 Covariance Matrix

The covariance matrix is used to whiten the data and, therefore, suppress the influence of the undesired background features. This matrix should be calculated using only the background data. Target pixels will "contaminate" the data and should not be included.

Similar to the findings of mean removal, researchers typically use the assumption that the target is sparse in comparison to the background and, therefore, has little influence in the covariance of the data. This assumption results in the covariance commonly calculated using the entire cube. Although this assumption might not apply to documents with large characters, it is generally applicable.

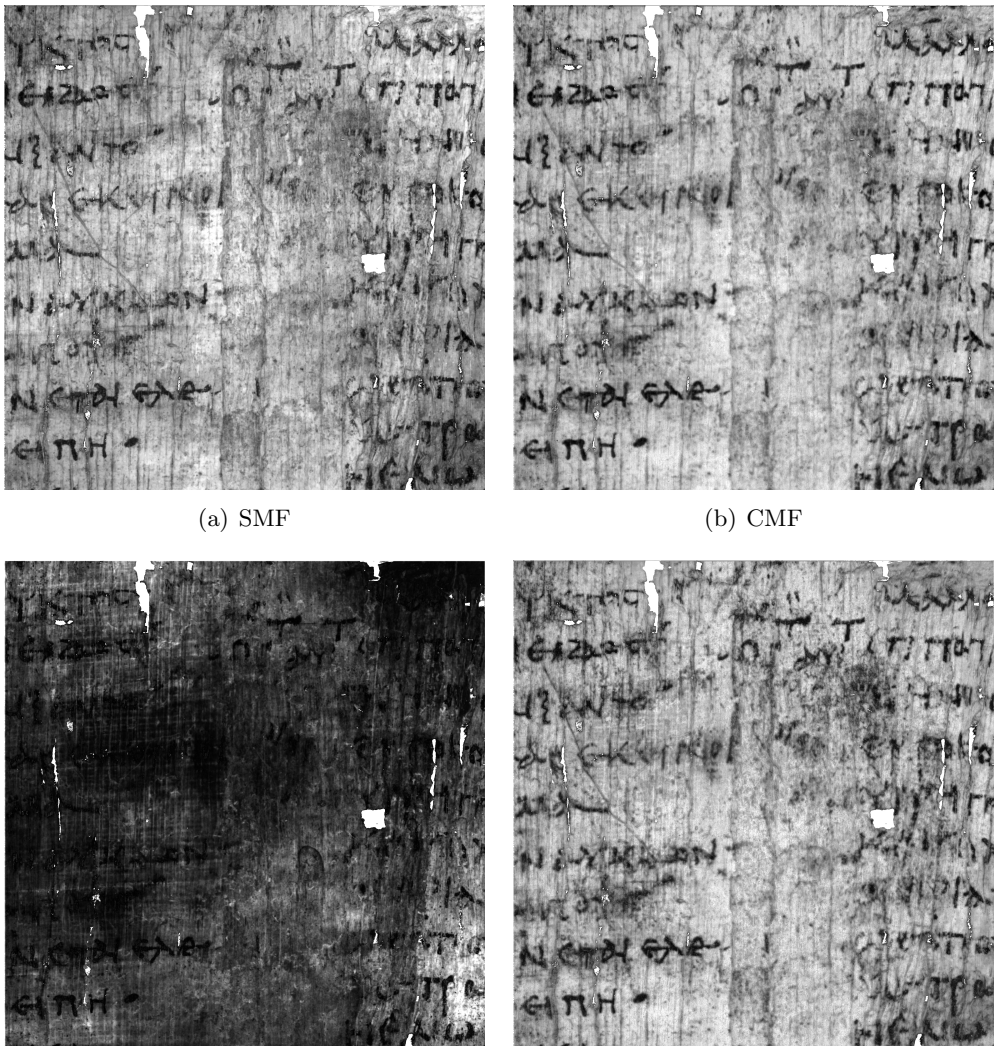
There are three pairs of equations that illustrate whether to use the covariance matrix or not: SMF-CMF, SAM-ACE, and CMF-CEM. The first two pairs, shown in Fig. 4.1, differ only by the use of the covariance matrix. Comparing both sets shows that the algorithm using the covariance matrix, CMF and ACE, were preferred to those that did not, SMF and SAM. Use of the covariance matrix yielded sharper images and gave a better visual representation of the ink while suppressing the background.

The third pair differs mainly by the use of the covariance and correlation matrices. Comparing these two equations showed that the correlation matrix was a sharper image but also contained more noise (see Fig. 4.2). The covariance matrix, although slightly blurred, was more visually appealing because of the removal of high frequency noise.

The LCMV and MSF equations are also similar while varying in the use of the covariance matrix. These equations have other differences and therefore were not included with the other analyzed pairs. It is interesting that comparing the LCMV and MSF also resulted in the covariance matrix providing the superior output. The covariance matrix is useful and, depending on the objective, may or may not be better than the correlation matrix.

4.3.3 Additional Observations

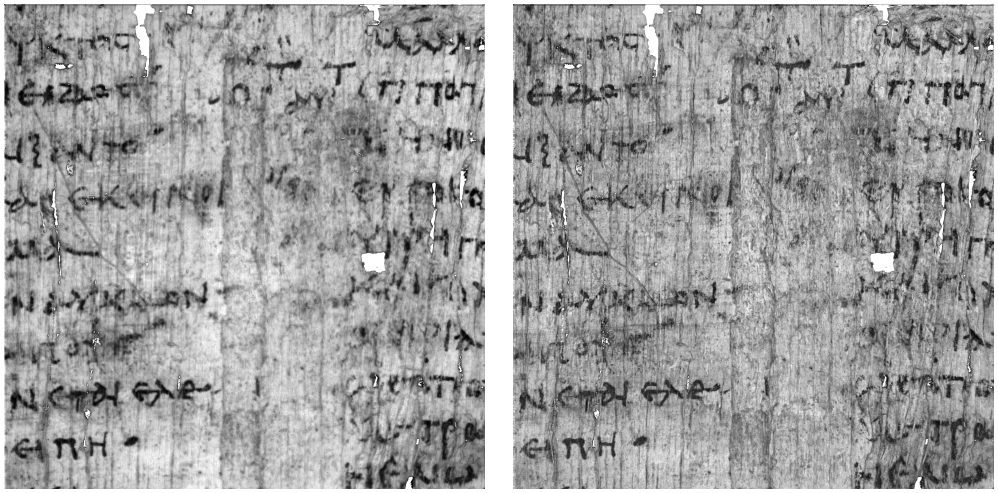
It was observed that the SMF is nearly identical to LDA. This is an odd observation because LDA is much more complex and yet does not outperform the simplest of all the filters, the SMF. The reason for their similarity has not been determined, but it is suspected that the angle between the background and the target is sufficiently close that the inner product with the matched filter is not effective. Also, the energy of the background seems to overpower the simple inner-product, dominating the result.



(a) SMF (b) CMF

(c) SAM (d) ACE

Fig. 4.1: Comparison of matched filters with and without a covariance matrix using MSI images of Oxyrhynchus fragment 1822 [7]. SMF and SAM do not use the covariance while CMF and ACE do. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.



(a) CMF (b) CEM

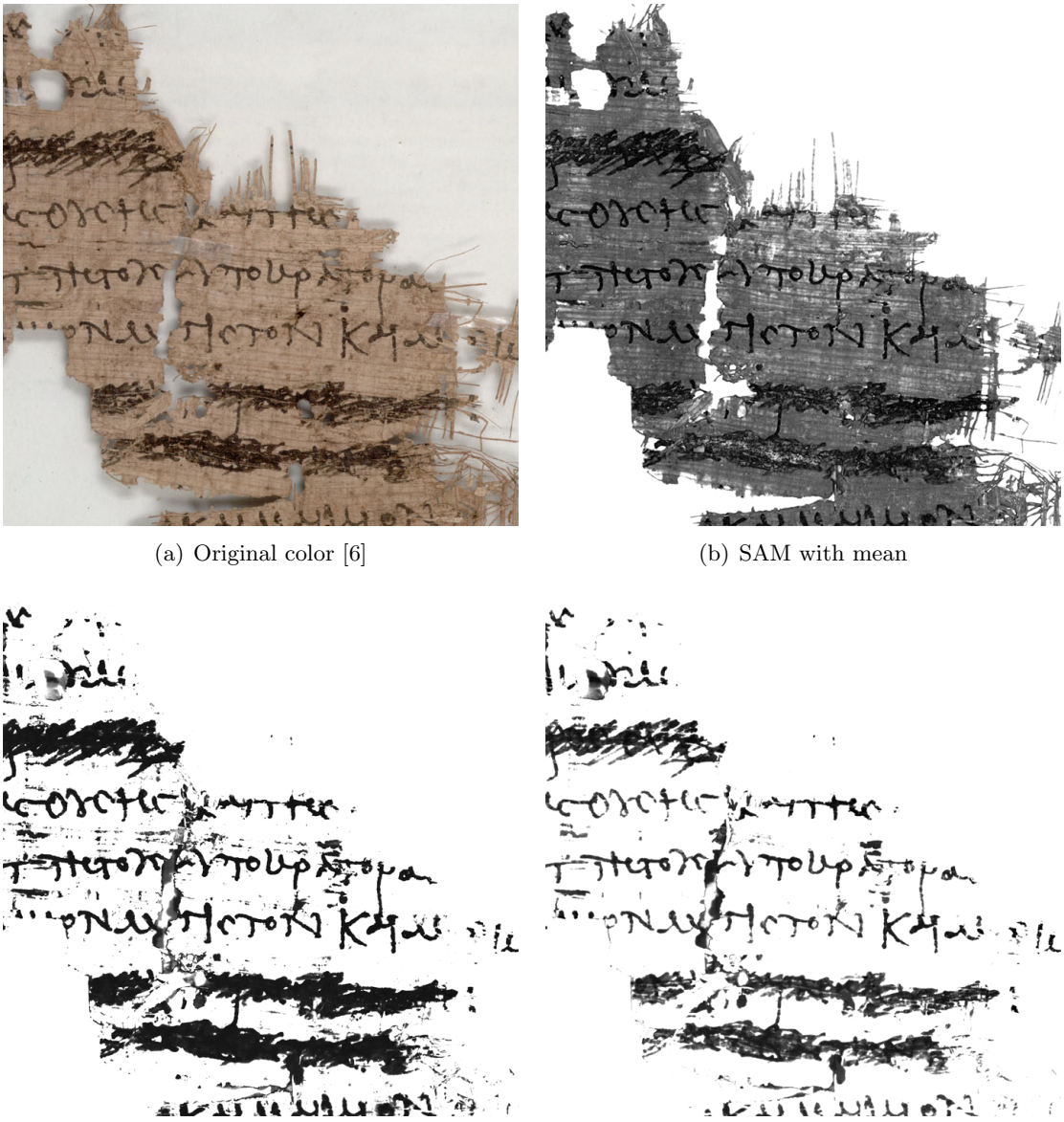
Fig. 4.2: Comparison of matched filters using a covariance or correlation matrix. CMF uses a covariance matrix while CEM uses a correlation matrix. Results were obtained using MSI images of Oxyrhynchus fragment 1822 [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

Another curious observation occurred when using matched filters to detect a word that had been scratched out, or covered over, by another ink. The best result was with the SAM matched filter with the mean removed from only the target spectrum, $\mathbf{s}$, used in the calculations. Figure 4.3 shows the original scratched out text and SAM results with different means removed. In this example, the target spectrum was able to discriminate between the two texts best with the cube mean removed.

4.4 Best Filter Methods

Each filter method was evaluated on the test documents with variations on mean removal and covariance/correlation matrices. The results give a better understanding of how best to use each method. These results are summarized as

- SMF–removing means is the same as keeping them,
- SAM–generally leave means in but remove the cube mean from the target to better suppress the background,
- CMF–remove the cube mean from the cube and target,



(a) Original color [6]

(b) SAM with mean

(c) SAM with background mean removed

(d) SAM with cube mean removed from target

Fig. 4.3: Matched filtering results for the scratched out text found in Oxyrhynchus fragment 3340 [7]. The SAM matched filter was used with three different mean subtraction methods: no mean removed (b), background mean removed from the target (c), and cube mean removed from the target (d). The SAM results have been contrast enhanced. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

- ACE–remove the cube mean from the cube and target,
- MSF–do not remove the means,
- t-statistic–remove the cube mean from the cube and target,
- F-statistic–leave the mean and use the correlation matrix,
- Quadratic–use the covariance of the cube data,
- DMF–use the covariance of the cube data,
- RX–was not useful in any of the tests,
- MVDR–no change was needed,
- CEM–remove the cube mean from the cube and the target,
- LCMV–remove the cube mean from the cube and the target,
- LDA–use the covariance of the cube data,
- QDA–no changes,
- PCA–no changes.

Table 4.3 displays the similarities of how the filters are best used.

All observations were made on the results after they were normalized as images. Image normalization includes subtracting the minimum value, dividing by the maximum, and multiplying by 65535 to form a 16-bit image ranging from 0 to 65535. This normalization effectively removes any of the constants that are present in the matched filter equations. It also places all the methods on the same viewing field, which is necessary due to the extreme range differences between equations.

After the changes listed in Table 4.3 were made to the equations in Table 4.2, many of the equations closely, if not exactly, resembled each other. The simplest equation similarity to see is with the CMF and MVDR equations, which originally differed by mean removal and a divisor constant removed by image normalization. When it was determined that the CMF was most effective with the mean removed, it became the same operation as the MVDR. Three other equations resulted in the exact same outcome after image normalization: LDA, Quadratic, and DMF. These filter equations, updated with Table 4.3 details, are written as

Table 4.3: Results of whether or not to remove means and whether to use the covariance or correlation matrix.

Remove Means	Leave Means	Cov	Corr
ACE		ACE	
CMF		CMF	
Quadratic		Quadratic	
MinVar		MinVar	
Matched		Matched	
	F-statistic		F-statistic
t-statistic		t-statistic	
CEM			CEM
LCMV			LCMV
SMF	MSF		
	SAM		

$$\begin{aligned} \text{DMF} &= (\mathbf{s}-\mathbf{b})^T \mathbf{R}^{-1} (\mathbf{x}-\mathbf{b}), \\ \text{LDA} &= (\mathbf{x}-\boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{s}-\boldsymbol{\mu}) - (\mathbf{x}-\boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{b}-\boldsymbol{\mu}), \\ \text{Quadratic} &= (\mathbf{x}-\mathbf{b})^T \mathbf{R}^{-1} (\mathbf{x}-\mathbf{b}) - (\mathbf{x}-\mathbf{s})^T \mathbf{R}^{-1} (\mathbf{x}-\mathbf{s}). \end{aligned} \tag{4.21}$$

Factoring the LDA equation gives

$$\text{LDA} = (\mathbf{x}-\boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{s}-\boldsymbol{\mu}-\mathbf{b}+\boldsymbol{\mu}). \tag{4.22}$$

Reducing terms and transposing gives

$$\text{LDA} = (\mathbf{s}-\mathbf{b})^T \mathbf{R}^{-1} (\mathbf{x}-\boldsymbol{\mu}), \tag{4.23}$$

which differs from the DMF equation only by subtracting $\boldsymbol{\mu}$ from $\mathbf{x}$ instead of $\mathbf{b}$.

To understand the Quadratic detector, the equation is factored in the form of

$$\mathbf{a}^T\mathbf{a} - \mathbf{v}^T\mathbf{v}, \tag{4.24}$$

where $\mathbf{a}$ is $\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{b})$ and $\mathbf{v}$ is $\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{s})$. Further manipulating gives

$$(\mathbf{a}+\mathbf{v})^T(\mathbf{a}-\mathbf{v}). \tag{4.25}$$

Substituting back in for $\mathbf{a}$ and $\mathbf{v}$ gives

$$\left(\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{b})+\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{s})\right)^T\left(\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{b})-\mathbf{R}^{-1/2}(\mathbf{x}-\mathbf{s})\right). \tag{4.26}$$

Combining terms gives

$$\left(\mathbf{x}-\frac{\mathbf{b}+\mathbf{s}}{2}\right)^T\mathbf{R}^{-1}(\mathbf{s}-\mathbf{b}). \tag{4.27}$$

Transposing (4.27) gives

$$(\mathbf{s}-\mathbf{b})^T\mathbf{R}^{-1}(\mathbf{x}-(\mathbf{b}+\mathbf{s})/2), \tag{4.28}$$

which differs from the DMF equation only by subtracting the average of $\mathbf{s}$ and $\mathbf{b}$ instead of $\boldsymbol{\mu}$. Writing the three equations after manipulations we get

$$\begin{aligned}\text{DMF} &= (\mathbf{s}-\mathbf{b})^T\mathbf{R}^{-1}(\mathbf{x}-\mathbf{b}),\\ \text{LDA} &= (\mathbf{s}-\mathbf{b})^T\mathbf{R}^{-1}(\mathbf{x}-\boldsymbol{\mu}),\\ \text{Quadratic} &= (\mathbf{s}-\mathbf{b})^T\mathbf{R}^{-1}(\mathbf{x}-(\mathbf{b}+\mathbf{s})/2).\end{aligned} \tag{4.29}$$

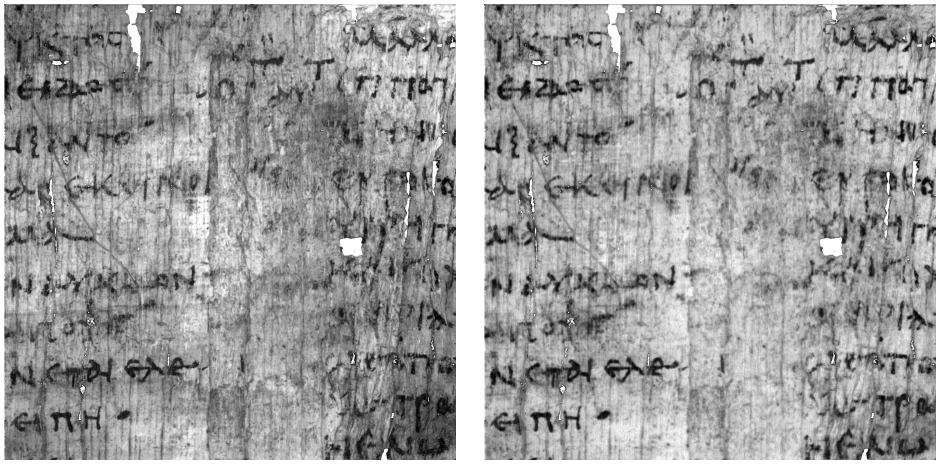
4.5 Conclusion

With documents, the ink usually has a relatively small influence on the mean of the data cube. This results in the spectral mean of the cube and background being of very similar shape. The difference of the three equations is in the selection of the mean to subtract from the cube. Since all of the means used are of similar shape, the matched filter output is similar. After normalizing the output into an image, it is reasonable that the three equations in (4.29) give the same result. The CMF and MVDR equations only differ from (4.29) by subtracting $\boldsymbol{\mu}$ from $\mathbf{s}$ in the first term instead of $\mathbf{b}$. Following the same reasoning it is easy to see that these two equations are very similar to the DMF, LDA, and Quadratic detectors.

Reducing the similar equations and eliminating the ineffective ones gave a list of six methods that were effective with the test set of documents. Table 4.4 contains the list of effective methods and the equations used. These equations simplify down to inner products using either the direction of largest variance as the target or a target spectrum supplied by the user. The supplied spectrum is often found by averaging target pixel spectrums from across the image. Adding a covariance/correlation matrix in the middle of the inner product is an effort to reduce the effect of the background in the calculation. SAM and ACE divide out the energy or length of $\mathbf{x}$ in an effort to remove the effect of the large pixel value difference between target and background. Figure 4.4 shows the results of the six useful matched filters listed in Table 4.4 when applied to the Oxyrhynchus fragment 1822.

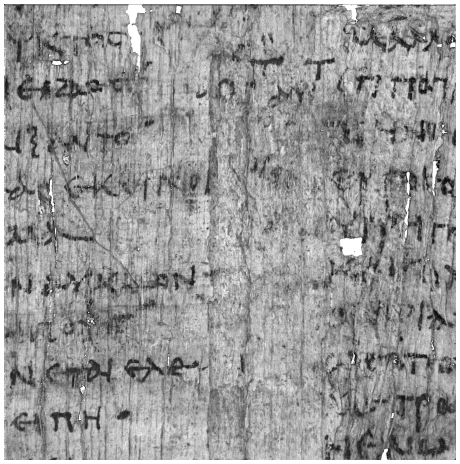
Table 4.4: Condensed list of useful matched filters, where the hat ($\hat{}$) indicates that the cube mean has been removed from the vector.

Filter Name	Equation
PCA	$\Lambda_x^{-1/2} V_x^T \hat{\mathbf{x}}$
CMF/LDA	$\hat{\mathbf{s}}^T \mathbf{R}^{-1} \hat{\mathbf{x}}$
CEM	$\hat{\mathbf{s}}^T \mathbf{\Gamma}^{-1} \hat{\mathbf{x}}$
SAM	$\frac{\mathbf{s}^T \mathbf{x}}{\sqrt{\mathbf{x}^T \mathbf{x}}}$
ACE	$\frac{\hat{\mathbf{s}}^T \mathbf{R}^{-1} \hat{\mathbf{x}}}{\sqrt{\hat{\mathbf{x}}^T \mathbf{R}^{-1} \hat{\mathbf{x}}}}$
LCMV	$\mathbf{x}^T \mathbf{\Gamma}^{-1} \mathbf{U} \left(\mathbf{U}^T \mathbf{\Gamma}^{-1} \mathbf{U} \right)^{-1} \mathbf{c}$



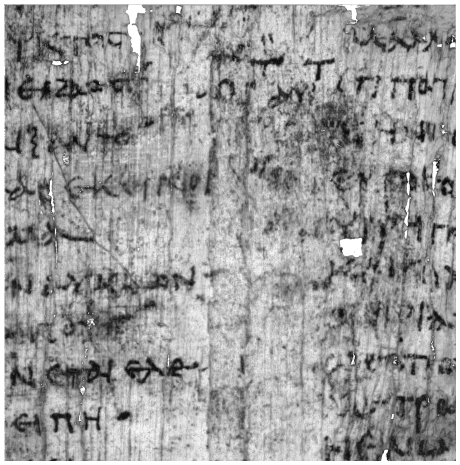
(a) PCA

(b) CMF

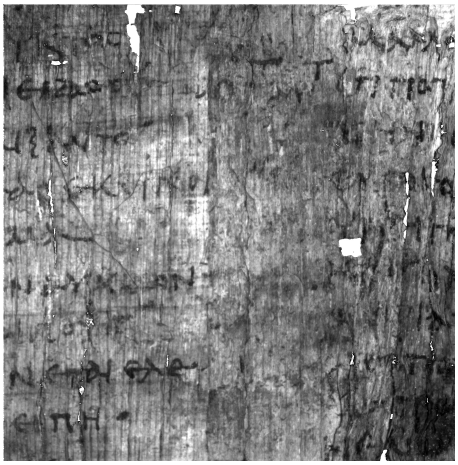


(c) CEM

(d) SAM



(e) ACE



(f) LCVM

Fig. 4.4: Comparison of matched filters using MSI images of Oxyrhynchus fragment 1822 [7]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

Chapter 5

Bleed-Through Removal

This chapter is an expansion of a paper presented at the Asilomar Conference on Signals, Systems, and Computers Nov. 2010 [50] and portions are reprinted here with permission from the IEEE. The original article is found in Appendix B.

5.1 Introduction

A problem that may be dealt with in the digital imaging of documents is bleed-through text, in which the text from the verso (back) side is seen from the recto (front) and vice versa. In the literature, a number of algorithms exist to remove the bleed-through text in both ancient and modern documents. The algorithms can be categorized as either non-blind or blind separations, using information from both sides or only one side, respectively.

Non-blind separation techniques compare the text on the recto and verso sides of the document in order to remove the bleed-through from each side. Registration of the recto and verso images is necessary in order to utilize all the information simultaneously. Before the registration, and one of the sides, commonly the verso side, must be horizontally reversed. This allows the bleed-through on the recto image to be registered with the text of the verso image and vice versa. Registration both is difficult and time-consuming, and is the major penalty for using information from both sides. Non-blind techniques have identified bleed-through text by adaptive linear filtering [51], thresholding [52], pixel matching [53], and joint histograms [54].

Once the bleed-through is identified, it may be replaced by an approximation of the background, or is smeared using an iterated wavelet transform technique [55]. A model for ink diffusion has been used to reverse the diffused ink causing bleed-through [56] while a

Bayesian source separation was recently proposed [57]. The non-blind algorithms use two grayscale images, one from each side of the document.

Blind separation has the advantage of not requiring a document image of the verso side. Computational savings are gained by not registering images, but often these algorithms are quite complex and time-consuming. Techniques that use only one grayscale image include directional wavelet transforms [58], global thresholding [59], and pixel matching [60]. When information in different wavelength bands is available, such as color images, principal component analysis (PCA) and blind source separation (BSS) techniques have been utilized [61–64]. Recursive K-means clustering following PCA was implemented by Drira and Emptoz [65]. A nonlinear BSS solution has also been applied [66].

Multispectral imaging (MSI) has not been previously employed for bleed-through removal by any of the algorithms proposed in the literature. Some use the three wavelength bands associated with color images but nothing more. MSI provides additional spectral information of the subject which can be used for distinguishing the bleed-through from the recto text and removing it. Having MSI data from both sides allows fast and unsupervised techniques to remove the data rather than costly alternatives. This chapter addresses how MSI data from both the recto and verso sides of documents might be utilized for bleed-through correction by combining the blind PCA and BSS techniques with non-blind architecture.

5.2 Methods

The MSI data for this thesis contains images at b specific wavelengths between 400 and 1000 nm. The image cube is reshaped from three to two dimensions with each row being an image and each column representing the spectral vector of each pixel. This reshaping has dimensions $b \times N$ where N is the number of pixels in each image. A model for bleed-through correction using the $b \times N$ version of the data, can be written as a linear mixing of the sources to form each image as

$$\mathbf{x}(t) = A\mathbf{s}(t) \quad t = 1, 2, ..., N, \tag{5.1}$$

where $\mathbf{x}(t)$ is the measured spectral vector at each pixel location, A is a $b \times M$ mixing matrix (where M is the number of sources) and $\mathbf{s}(t)$ is the spectral response of each of the sources that make up the image.

Since neither the sources, $\mathbf{s}$, nor the mixing matrix, A, are known, this problem falls within the BSS framework. As has been noted by Tonazzini *et al.* [64], the linear framework is too simple for the complex problem found in ink seepage, but has still been shown to produce good results. In their research, a three band color image was used, but it was noted that the same process could also be used with MSI data. This thesis tests the utility of the decorrelation techniques of PCA and the source separation techniques of independent component analysis (ICA) for correcting bleed-through degradation. These two techniques are well documented in the signal processing literature [67, 68]. Also tested is a recently proposed symmetric orthogonalization method, similar to PCA but with an added rotation to the whitening matrix [62]. PCA can be defined as

$$\text{PCA} = W^T\mathbf{x}, \tag{5.2}$$

where $\mathbf{x}$ is the $b \times N$ data matrix with images as row vectors, W^T is the $b \times b$ dimensional whitening matrix defined as $\Lambda_x^{-1/2}V_x^T$ with Λ_x and V_x being the eigenvalues and eigenvectors of the autocorrelation matrix of $\mathbf{x}$, respectively. The symmetric orthogonalization transformation is defined as

$$\text{Orthogonal} = W_s^T\mathbf{x}, \tag{5.3}$$

where W_s is defined as $V_x\Lambda_x^{-1/2}V_x^T$. The additional eigenvector matrix in front provides a rotation to the common whitening matrix W [62]. All ICA calculations were performed using the fastICA algorithm [67].

All transformations are performed on the two-dimensional image cube labeled here as $\mathbf{x}$. Each transformation returns a matrix the same size as $\mathbf{x}$ with output images as rows. In order to view the output images, they must be transformed back into the original $m \times n$ dimensions of the input images used to form the image cube. They must also be normalized

to contain values consistent with grayscale images. The process used to normalize the images can be written as

$$Y = \frac{y - \min(y)}{\max(y) - \min(y)} \times 65535, \tag{5.4}$$

where y is the data vector, Y is the normalized vector, and 65535 is used for 16-bit images as the maximum possible pixel value. Outputs from the three transformations are only unique to scaling and often must be inverted in order to be viewed correctly.

The ICA and symmetric orthogonalization transformations are derived to extract the independent components of the data. This may be ideal for many cases, but with documents there is no reason to assume that the bleed-through and recto text are independent. PCA separates the data components by projecting the data onto an orthogonal basis using the eigenvectors matrix. The first principal component (PC) will have most of the variance in the data and, in the case of bleed-through, contains both recto and bleed-through texts. The second PC will begin to separate the texts due to the difference in spectral characteristics caused by the verso text being filtered through the parchment.

Bleed-through can be removed from both the recto and verso texts using the first two PCs by subtracting/adding the first two PCs pixel by pixel. Subtraction provides the recto text and addition the verso, or vice versa. This arithmetic operation can be performed on the PCA output vectors directly or on the normalized grayscale images. A direct subtraction/addition of the PC data vectors was effective in eliminating the bleed-through, but a scaling of the subtracted image could be useful with other data. It is not advised to scale the subtraction of PCA data vectors, since they are orthogonal and scaled by the eigenvalues.

Two methods are proposed to find an appropriate scaling for the normalized PCA images, denoted PC_1 and PC_2 (converted into vector format for calculations). The first is simply the least squares solution to

$$\min_a \|\mathrm{PC}_1 - a \times \mathrm{PC}_2\|_2^2, \tag{5.5}$$

which is known to be

$$a = \frac{\langle \mathrm{PC}_1, \mathrm{PC}_2 \rangle}{\langle \mathrm{PC}_2, \mathrm{PC}_2 \rangle}, \tag{5.6}$$

where $\langle\ ,\rangle$ indicates the standard Euclidean inner product. This method finds the best scaling, in the least squares sense, to make PC_2 as close to PC_1 as possible.

The least squares method has the possibility of over-correcting, or removing too much, of the bleed-through. When over-corrected, the removed bleed-through leaves a white shadow rather than blending into the background as desired. Another method for calculating the scaling parameter a is to make the result of the subtraction as close to the background as possible. This should help remove the effects of over-correcting, but requires training information indicating where bleed-through and background occurs in the images. This is obtained by having the user tag pixels by hand. The scaling is then found by

$$\min_a \|\mathrm{Cov}(\mathbf{P}_1 - a\mathbf{P}_2) - \mathrm{Cov}(\mathbf{background})\|_2^2 \tag{5.7}$$

where $\mathbf{P}_1$ and $\mathbf{P}_2$ are vectors containing the values of pixels tagged as bleed-through in the first and second PC images, respectively, and **background** is a vector of the values of pixels tagged as background. This can then be written as

$$\min_a \left\| E[(\mathbf{P}_1 - a\mathbf{P}_2)(\mathbf{P}_1 - a\mathbf{P}_2)^T] - E[\mathbf{P}_1 - a\mathbf{P}_2]E[\mathbf{P}_1 - a\mathbf{P}_2]^T - R_b \right\|_2^2, \tag{5.8}$$

where R_b, the covariance of the background pixels, is a scalar. Multiplying out gives

$$\min_a \left\| A - 2aB + a^2C - \mu_1^2 + 2a\mu_1\mu_2 - a^2\mu_2^2 - R_b \right\|_2^2, \tag{5.9}$$

where $A = E[\mathbf{P}_1\mathbf{P}_1^T], B = E[\mathbf{P}_1\mathbf{P}_2^T]$, and $C = E[\mathbf{P}_2\mathbf{P}_2^T]$ are scalars. Taking the derivative gives an equation that can be solved for a,

$$4a^3K^2 + 6a^2JK + 2a(J^2 + 2KH) + 2JH = 0, \tag{5.10}$$

where $J = 2\mu_1\mu_2 - 2B, K = C - mu_2^2$, and $H = A - R_b$ are constant scalars. Due to the need for a third order numeric solver to find a, this method will be referred to as the solver method. This could possibly give three solutions to a.

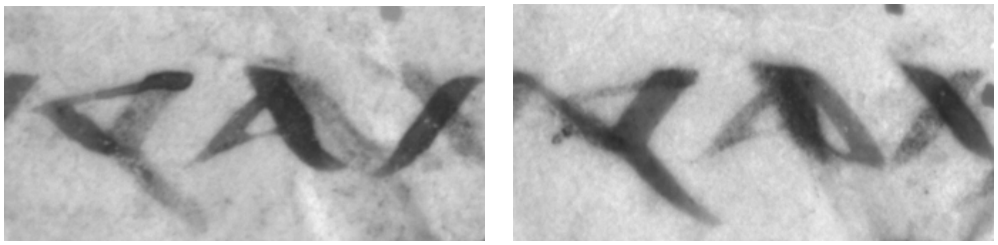
5.3 Test Results

Two text samples were selected for analysis. One is a severely deteriorated parchment scroll which contains bleed-through text, folds in the parchment, stains, holes, and large burned sections. The other is a 19^{th} century document with very slight bleed-through. Three non-blind separation techniques were implemented and compared in this thesis: PCA, ICA, and symmetric orthogonalization. These three techniques were previously shown to be effective with one-sided color images [62]. The algorithms are tested to determine their utility using a stacked image cube formed from multispectral images of both the recto and verso sides of the document. To form the stacked cube, the verso text was horizontally flipped and registered to the recto.

5.3.1 Ancient Scroll

The cropped section of the scroll, seen in Fig. 5.1, shows the text with strong bleed-through on both sides of the parchment. The results of PCA, ICA, and the symmetric orthogonalization transformations using a stacked image cube are shown in Fig. 5.2. These methods were also applied to the individual front and back image cubes but were less accurate than the results with the stacked images in all cases. It is apparent that the ICA and symmetric orthogonalization methods were not able to accurately eliminate the bleed-through text. PCA did have promising results due to its ability to separate the two texts into opposite grayscale values in the second PC. PCA was also easier to use due to its ability to consistently place the most important images at the first and second place in the output order. ICA and symmetric orthogonalization do not have this feature, requiring the user to determine which of all the output images are useful.

Direct subtraction of the PCA data vectors, as well as scaled subtraction of the normalized PC images were used to remove the bleed-through text. Figure 5.3 shows the results



(a) Recto (b) Verso

Fig. 5.1: A cropped section of the uncorrected recto (a) and horizontally flipped verso (b) scroll fragment A [7] at 700 nm. The dark characters are the recto/verso text with hazy, gray characters as bleed-through. Each cropped section corresponds to the same registered area in both (a) and (b). Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

of the bleed-through removal on the recto side of the ancient scroll. All of the methods are similar in this case. Figure 5.4 shows the results of bleed-through removal on the verso side. For this case, the subtraction was simply changed to addition without changing the scaling parameter a. The results of the solver method were poor in this case. To improve the solver method's results, it is necessary to re-run the solver algorithm with $\mathbf{P}_1$ correlating to the verso and $\mathbf{P}_2$ to the recto. This flipping is needed because the algorithm needs to match the subtraction to the background of the verso side, not the recto. The result of this method is shown in Fig. 5.5. Re-running the least squares method with PC_1 and PC_2 flipped did not change the result.

5.3.2 19^{th} Century Document

The 19^{th} century document does not contain strong bleed-through and has a very uniform background compared to the ancient scroll fragment. A section of original text is seen in Fig. 5.6. Similar to the ancient scroll findings, ICA and symmetric orthogonalization are not able to remove the bleed-through using the stacked image cube of the 19^{th} century document. PCA was able to separate the texts as expected. Figure 5.7 shows results of the PCA subtraction bleed-through removal techniques. The least squares method over-corrected the bleed-through, leaving white characters in the output image, but the solver

method was able to calculate a scaling parameter that made the bleed-through blend in with the background.

Applying the transformations to the recto text image cube, without information from the verso, gave significantly better results. If the stacked cube has dimensions $m \times n \times 2b$, the recto cube will be the $m \times n \times b$ formed from the recto images. Using the recto image cube, ICA and symmetric orthogonalization were able to remove the bleed-through but tended to be noisy while PCA was still able to separate the two texts in the second PC. The results of each transformation are shown in Fig. 5.8. Results from adding the PC data vectors, the normalized PC images, and the addition scaled by parameter a in (5.6) are shown in Fig. 5.9. All are shown here to demonstrate the output differences between the addition methods. The scaling parameter a was calculated to be 2.3835 using the normalized PC_1 and PC_2 images. The best visual output in this case is obtained by adding the normalized PC images. Scaling the addition of the normalized PC images is very similar to directly adding the PC data vectors.

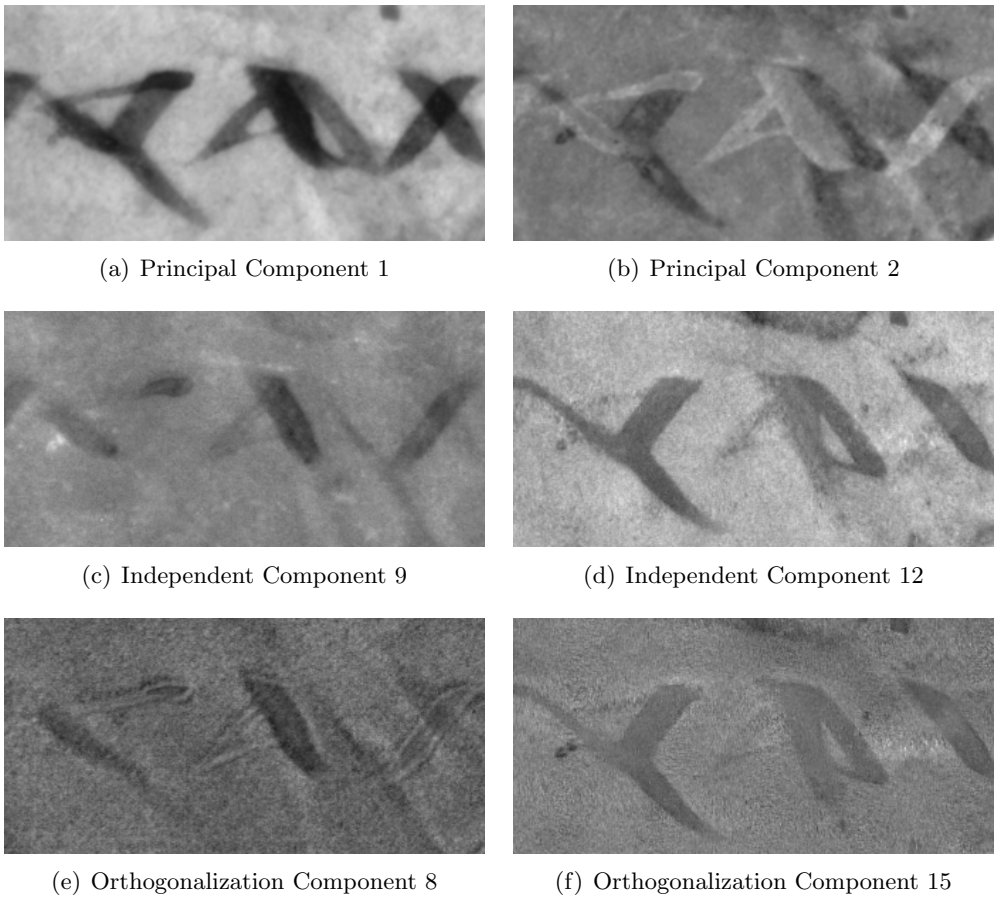
5.4 Conclusion

PCA was found to be the most effective method for removing bleed-through text from both the ancient scroll and a 19^{th} century document. The first and second PCs were able to separate the bleed-though from the recto and verso texts. Subtracting the two PCs resulted in an image where the bleed-through has been removed. This can be performed on both the PC data vectors or the normalized PC images with good results.

The algorithm worked best for the ancient scroll data using the stacked image cube containing information from both sides of the parchment. This gave the needed information to accurately remove the severe bleed-through deterioration from both the recto and verso text simultaneously. The 19^{th} century fragment, with much less bleed-though present, produced good results using either the stacked or recto image cubes. If the stacked cube is used, the solver method was much more effective than the least squares method for calculating the subtraction scaling parameter.

ICA and symmetric orthogonalization transformations were unable to remove the bleed-though in the ancient scroll data. They produced improved images with the recto image cube of the 19^{th} century document, but the results were not superior to that obtained using the PCA approach. PCA was able to maintain the visual characteristics of the original document while the other two transformations changed the look dramatically and added noise to the image.

These results show that, when available, MSI data allows bleed-through to be removed from simple and complicated documents. The large amount of data facilitates a fast unsupervised algorithm which is much simpler than those methods proposed for bleed-through removal without MSI data.



(a) Principal Component 1

(b) Principal Component 2

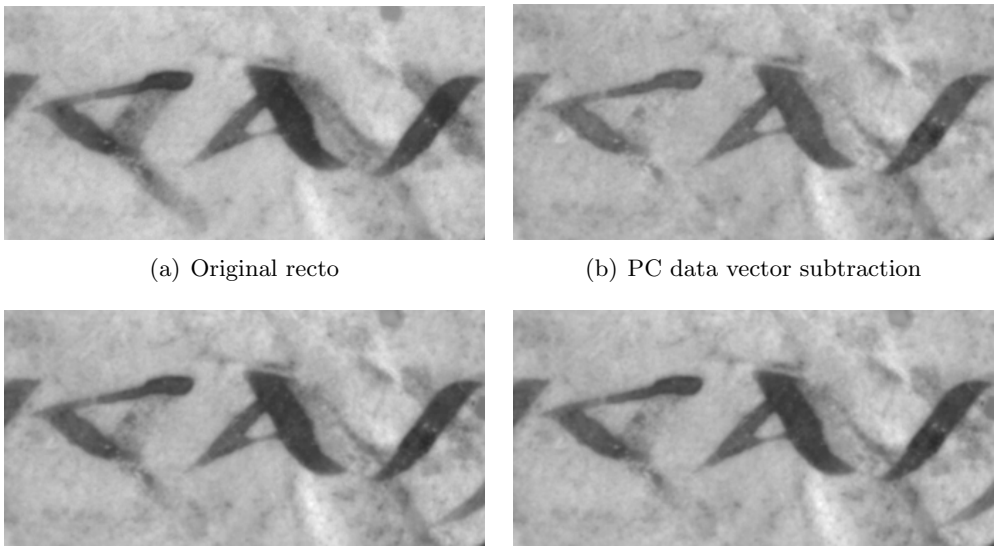
(c) Independent Component 9

(d) Independent Component 12

(e) Orthogonalization Component 8

(f) Orthogonalization Component 15

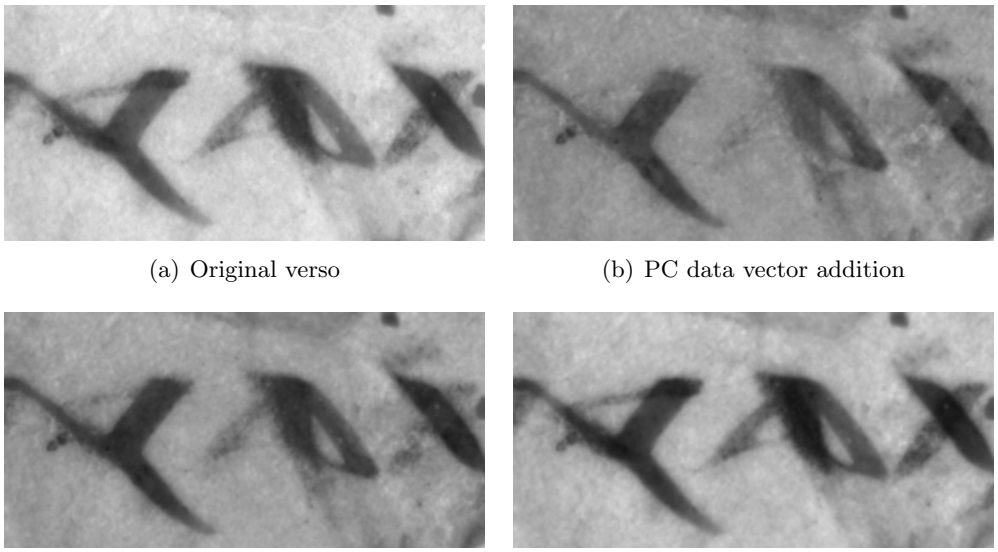
Fig. 5.2: Scroll results for PCA, ICA, and symmetric orthogonalization using a stacked image cube of fragment A [7]: Principal components 1 and 2, independent components 1 and 12, and symmetric orthogonalization components 8 and 15. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.



(a) Original recto (b) PC data vector subtraction

(c) Least squares method (d) Solver method

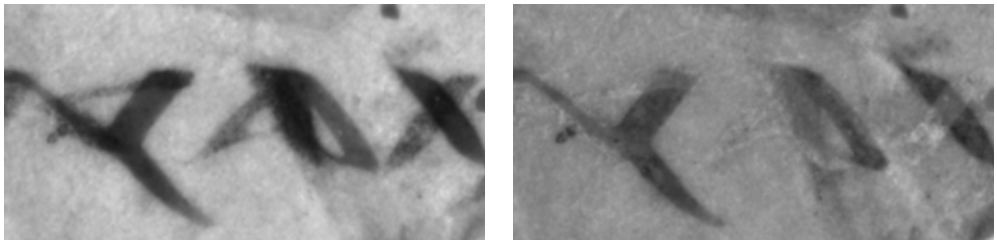
Fig. 5.3: Original fragment A recto [7] with bleed-through removed using direct PC data vector subtraction, least squares method with a=1.0831, and the solver method with a=.4513. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.



(a) Original verso (b) PC data vector addition

(c) Least squares method (d) Solver method

Fig. 5.4: Original fragment A verso [7] with bleed-through removed using direct PC data vector addition, least squares method with a=-1.0831, and the solver method with a=-.4513. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.



(a) Negated subtraction scaling (b) Recalculated solver scaling

Fig. 5.5: Bleed-through removed from fragment A [7] using the negated subtraction scaling and the recalculated solver scaling a=.3844. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

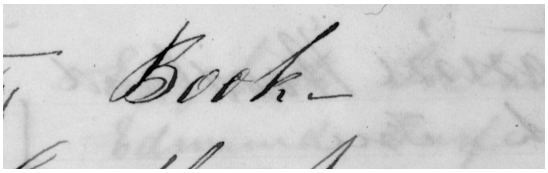
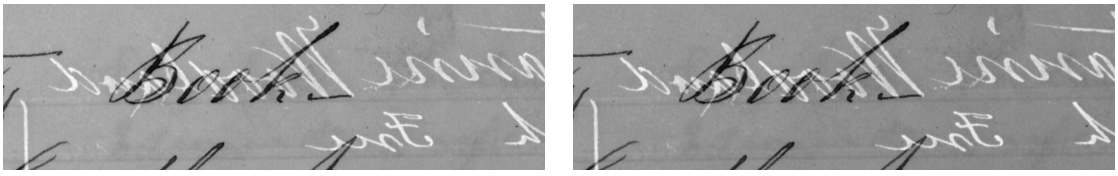
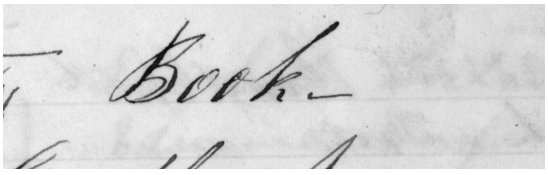


Fig. 5.6: 19^{th} century uncorrected document recto at 650 nm. The word *Book* is the overtext and the hazy letters filling the rest of the frame are bleed-through.

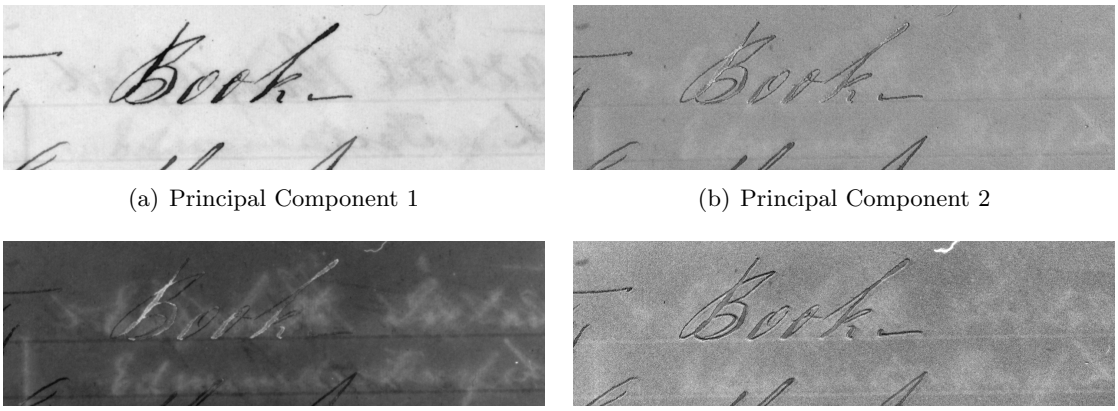


(a) PC data vector addition (b) Least squares method



(c) Solver method

Fig. 5.7: Bleed-through removed from the stacked image cube of the 19^{th} century document by adding the PC data vectors, the least squares method, and the solver method.



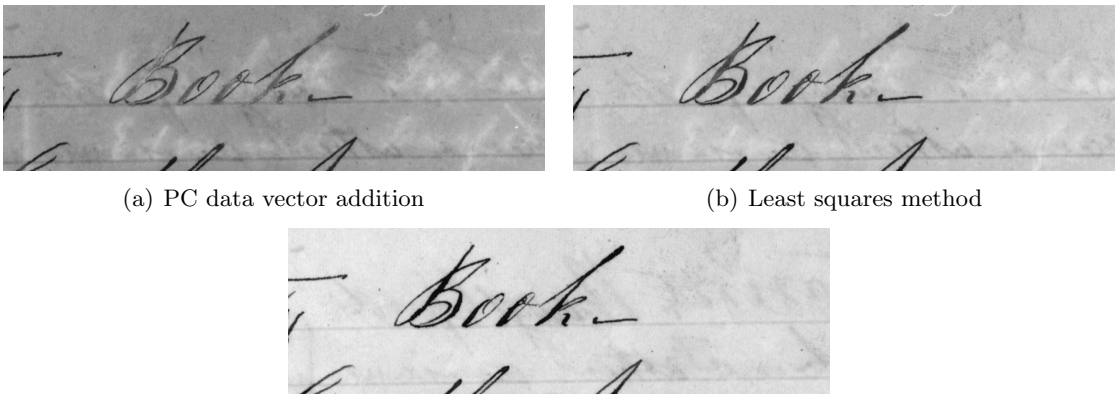
(a) Principal Component 1

(b) Principal Component 2

(c) Independent Component 6

(d) Orthogonalization Component 12

Fig. 5.8: 19^{th} century document results for PCA, ICA, and symmetric orthogonalization using the recto text image cube: Principal components 1 and 2, independent component 6, and symmetric orthogonalization component 12.



(a) PC data vector addition

(b) Least squares method

(c) Solver method

Fig. 5.9: Bleed-through removed from recto image cube of the 19^{th} century document using PC data vector addition, the least squares method with a=2.3835, and the solver method with a=.3069.

Chapter 6

Shadow Removal

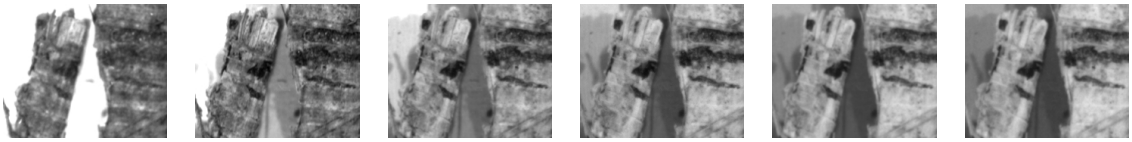
Many of the documents used in this thesis are encased in glass for preservation purposes. While adequately protecting the fragment from harm, this glass raises the document up from the viewing table, thus causing shadows in the imagery. In most cases, the camera is directly above the document and, therefore, the lights must be placed at an angle. With two to four lights being used, a number of shadows develop along the directions of the light axes as seen in the Oxyrhynchus fragment 2383 image of Fig. 6.1.

Fig. 6.1: Oxyrhynchus fragment 2383 at 800 nm as an example of shadows in MSI imagery. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

In grayscale MSI images, shadows are often the same shade of gray as the document and can be difficult to distinguish from the document. This is particularly true for the infrared images which have more pronounced shadows at longer wavelengths of light. The problem is further complicated when holes in the middle of the document create dark regions that can be confused with text. For these reasons, removing shadows caused by holes and edges can significantly improve legibility.

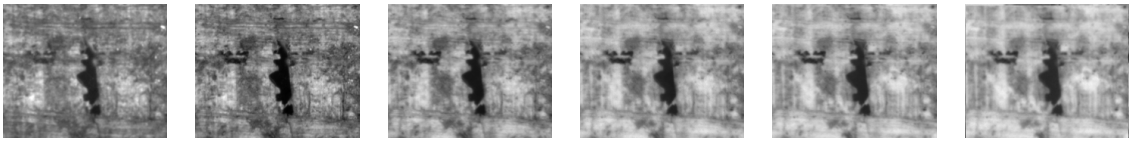
Shadows must be identified before they can be removed. This is a detection problem where a known target must be separated from the background. Each of the thirteen matched filter equations listed in Table 4.2 was tested for its ability to identify the shadows. The results identified two classes of shadows which are called here light and dark. Light shadows are regions which have some illumination (see Fig. 6.2) and dark shadows have little to none (see Fig. 6.3). Most of the matched filters detect the light shadows due to their spectral characteristics being significantly different from anything in the document. The dark shadows, however, are much more difficult to detect due to the similarity of the ink's spectral curve with that of the shadow's.

It is necessary to detect light and dark shadows separately since they have distinct spectral characteristics. Training information must be provided by identifying pixel locations of known ink, dark shadows, and parchment/papyrus without ink. For light shadows, the matched filters are trained with ink as the target and the parchment/papyrus as the background. For dark shadows, ink remains the target but with the dark shadows as the background.



(a) 450 nm (b) 550 nm (c) 650 nm (d) 750 nm (e) 850 nm (f) 950 nm

Fig. 6.2: An example from Oxyrhynchus fragment 2838 [7] demonstrating how a light shadow changes at different wavelengths. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.



(a) 450 nm (b) 550 nm (c) 650 nm (d) 750 nm (e) 850 nm (f) 950 nm

Fig. 6.3: An example from Oxyrhynchus fragment 2383 [7] demonstrating how a dark shadow changes at different wavelengths. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

Using this approach, QDA was determined to be the most effective in distinguishing the dark shadows from the ink and parchment. When using training data to detect light shadows, QDA is able to identify the location of the document, effectively eliminating the shadows and background surface surrounding the fragment. Training QDA for light shadows does not, however, perform well in regions of dark shadow, which are typically identified as part of the document. An example of the light shadow detection result for Oxyrhynchus fragment 2383 is given in Fig. 6.4.

Training QDA to detect the black holes results in a thresholded image identifying the location of the dark holes. These two QDA outputs, light and black hole detection, can then be combined to create an image with the document clearly distinguished from the shadows as illustrated by Fig. 6.5. In difficult subjects, combining the thresholded PCA and LDA images with the QDA thresholded images can more effectively eliminate the shadows.

Using QDA to remove shadows is shown to be effective in Fig. 6.5, but there are areas where errors did occur. It is possible to have ink falsely identified as shadow and removed, while dark shadows might be considered ink and not removed. These two errors can be corrected while thresholding the QDA images, but it is difficult to remove both simultaneously. For instance, if the threshold is placed so that all of the dark holes are removed, some of the text might be removed as well. If the threshold is placed so that none of the text is removed, some of the dark holes might remain. In most cases these errors are very small when considered in the overall image. In practice, the threshold is placed somewhere in the middle of these extremes so that very little text is removed and almost

(a) Thresholded light shadow QDA

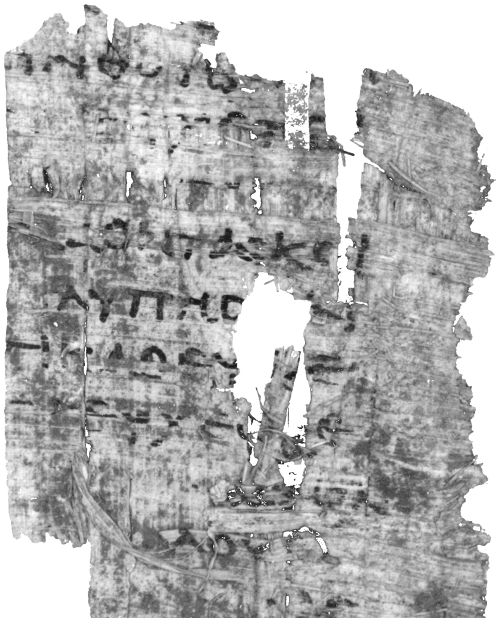
(b) Masked light shadows

(c) Thresholded dark shadow QDA

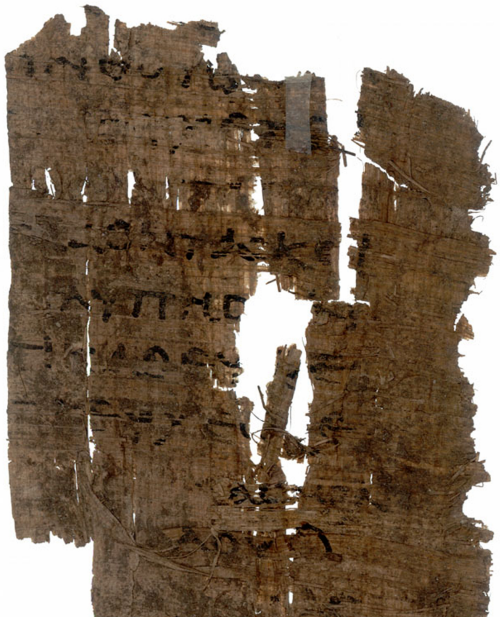
(d) Masked dark shadows

Fig. 6.4: Thresholded QDA result for light shadow detection (a), which is used to mask off the light shadows in Oxyrhynchus fragment 2383 (b) [7]. Thresholded QDA result for dark shadow detection (c), which is used to mask off the dark shadows (d). Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

(a) Thresholded QDA



(b) Masked original MSI



(c) Masked original color

Fig. 6.5: Thresholded QDA result for light and dark shadow detection combined (a), which is used to mask off both light and dark shadows in Oxyrhynchus fragment 2383 (b) [7]. This is compared with the original color image (c) [6]. Oxyrhynchus images © The Oxford Imaging Papyri Multi-spectral Laboratory and the Egyptian Exploration Society.

all the dark shadows are removed. In the example of Fig. 6.5, the threshold was placed so that nearly all of the shadows are removed while part of one character is removed.

It should be noted that this technique requires significant user input both in training the input to QDA and in thresholding the output. It is also a quadratic function which can become computationally intensive and slow with large images. With this considered, it does provide an excellent statistical means for identifying the shadows. Thresholding QDA yields a mask image containing information as to which dark regions might be shadows instead of ink, thus aiding the analyst in transcription.

Chapter 7

Summary and Conclusion

This thesis presents signal processing techniques that may be used to enhance multispectral images of ancient documents. Image fusion and matched filtering are employed to combine the information contained in a large number of MSI images into a single image. A method for removing distracting bleed-through and shadows is also presented. Combining images may enhance significant detail and remove distractions, thus giving the papyrologist studying the text of the document additional tools for identification of previously unreadable texts.

7.1 Results

A set of ten Oxyrhynchus papyri and one Derveni papyrus fragments were analyzed using multiple matched filter and image fusion techniques. Originally, it was desired to better understand which methods were the best for various surface characteristics found in each of the test documents. After analysis, the fragments were grouped into four categories:

1. normal text,
2. fain text,
3. burned areas,
4. scratched out text.

The faint text category is comprised of documents with dirt, stains, or possibly abrasions which obscure the text. In addition to these categories, texts with bleed-through and shadow-causing holes were analyzed. A summary of the more useful methods, with their strengths and weaknesses when applied to the document test set, is given in Table 7.1.

In order to ascertain which of the methods was best for each of the fragment categories, the enhancement results for various matched filters were judged by a papyrologist who

Table 7.1: Listing of useful enhancement methods.

Method	Strengths	Weaknesses
Principal Component Analysis (PCA)		
	Produces sorted image outputs which combine information together and separate sources.	Does not use trained data, so it is not adaptable to different desired results.
Clutter Matched Filter (CMF)		
	Similar to the first PCA output but tends to be slightly blurred which can help in removing noise, making characters easier to read.	Too much blurring can occur.
Constrained Energy Minimization (CEM)		
	Correlation is used instead of covariance which tends to contain fine detail.	Background detail is also enhanced, which may be distracting.
Spectral Angle Map (SAM)		
	Strongly separates ink from background which helps eliminate gray areas.	Dark background features are commonly grouped with ink.
Adaptive Coherence Estimator (ACE)		
	Detects faint text and makes it darker.	Some dark features of the background are also made darker.
Matched Subspace Filter (MSF)		
	Provides images that are slightly overexposed, so that few pixels are light saturated.	The details can be difficult to distinguish in dark areas of the image.
Linearly Constrained Minimum Variance (LCMV)		
	Lightens the background and does well in preserving the background texture.	Tends to enhance background more than ink.
Quadratic Discriminant Analysis (QDA)		
	Able to distinguish the fragment from shadows.	Output must be thresholded and the algorithm can be slow for large images.

specializes in the study of papyrus documents. Prior to this input, the helpful results were considered to be those that could suppress the background while keeping the ink dark, thus increasing the contrast between the text and papyrus. After the papyrologist's input, it became clear that background suppression is not desired, since in many cases it provides useful clues in deciphering the text. Rather than suppress any of the detail, all should be enhanced, including the background and papyrus fibers. With this taken into consideration, a summary of the most effective matched filters for each of the four problem categories is given in Table 7.2.

7.2 Discussion

Principal Component Analysis proved the most useful method used in this thesis. PCA was helpful in analyzing every one of the test fragments. It is not coincidental that it is listed as an image fusion, matched filter, and bleed-through removal tool. The simple PCA algorithm provides the user with component images where each image contains additional information about the subject. The first component combines the majority of the information, which is the goal of image fusion. Subsequent images separate the various independent components of the data, which is helpful in detecting ink and other substances. Due to its

Table 7.2: Most effective matched filters for different texts, listed in order of quality.

Text Group	Useful Methods
Normal	CMF PCA CEM
Faint	CEM ACE SAM
Burned	ACE CMF LCMV
Scratched Out	PCA CEM ACE

simplicity and broad range of applications, it has become the most widely applied MSI enhancement method. If only one process were to be selected to enhance or analyze an MSI image cube, PCA should be that process.

Matched filters proved effective for enhancing specific areas of text. While PCA is excellent at enhancing the document in general, matched filters provide a better means for enhancing specific characteristics. Even with the list of *best* filters given in Table 7.2, if possible, it is still recommended that all of the matched filter equations be evaluated for each fragment, as each may enhance a different section of text.

In most cases, the papyrologist was unable to definitively say that a particular method was preferable. Rather, each of the methods were useful in its own specific way. For instance, one matched filter might enhance a few characters on the bottom right-half of the document while another enhanced the top left, and yet another enhanced all of the text except these two areas. All three of the outputs are useful in transcribing the text. Multiple methods occurred for nearly all of the fragments presented to the papyrologist. This revealed the benefit of having the suite of matched filters presented in this thesis. With all of the results available, the papyrologist typically will transcribe most of the text from one image and then use the remaining images to help determine difficult characters.

The methods that were useful for a given papyrus fragment proved to be difficult to combine into a single image using image fusion techniques. The result from a matched filter that enhanced a small portion of the text often had very undesirable effects on other portions of the image. Image fusion tended to combine both good and poor areas making the fused image less useful than the source images.

An investigation was made into an image technique to provide papyrologists the ability to review many images of the text at once. One effective method found for viewing multiple images is to utilize software with layering capabilities such as Adobe Photoshop [26]. This capability allows the layers to quickly be flashed on and off, providing the user with an easy way to view the changes between images. Viewing the images this way was found to be far superior to the traditional side-by-side viewing method.

Though the enhancement obtained by the methods discussed in this thesis may be ever-so-slight to the untrained eye, it can be a very significant contribution to those who have dedicated significant amounts of time studying the text. Even allowing only one or two additional characters to be recognized can lead to very substantial findings.

Future Work

The methods presented here are the more straightforward signal processing techniques that could be applied to MSI data. This thesis considered only methods that are applied pixel-by-pixel. A future area of research is to investigate the utilization of the two-dimensional information contained in the image. Rather than performing calculations on individual pixels, it may be more effective to utilize the information from a group of surrounding pixels. Commonly referred to as neighborhood processing, this is particularly useful for text enhancement since ink pixels are likely to be found in groups. It is possible to harness the known ink shapes in this processing since it is usually known which letters/shapes are possible.

Another area of focus is to delve further into statistical signal detection techniques, utilizing Bayes theory to include prior knowledge about the ink and/or the papyrus statistics. The only statistics used in this thesis are the linear and quadratic discriminate equations used as matched filters. These contain prior probabilities on how often ink pixels occur in the image as compared to background. This thesis made these probabilities equal, essentially removing them from the equation. If these probabilities were known, they could provide added information to the detection problem.

References

[1] Derveni Papyrus fragment C4. Multispectral images taken at the Thessaloniki Archaeological Museum, Thessalonike, Greece, October 2-10, 2006, by Gene A. Ware in conjuction with the Ancient Textual Imaging Group at Brigham Young University.

[2] B. Zuckerman, "Bringing the Dead Sea Scrolls back to Life: a new evaluation of photographic and electric imaging of the Dead Sea Scrolls," *Dead Sea Discoveries*, vol. 3, no. 2, pp. 178–207, 1996.

[3] G. A. Ware, D. M. Chabries, R. W. Christiansen, J. E. Brady, and C. E. Martin, "Multispectral analysis of ancient Maya pigments: implications for the Naj Tunich corpus," in *IEEE International Geoscience and Remote Sensing Symposium, IGARSS*, vol. 6, pp. 2489–2491, 2000.

[4] G. A. Ware, D. M. Chabries, R. W. Christiansen, and C. E. Martin, "Multispectral document enhancement: ancient carbonized scrolls," in *IEEE International Geoscience and Remote Sensing Symposium, IGARSS*, vol. 6, pp. 2486–2488, 2000.

[5] R. L. Easton and W. Noel, "Infinite possibilities: Ten years of study of the Archimedes Palimpsest," *Proceedings of the American Philosophical Society*, vol. 154, no. 1, pp. 50–76, 2010.

[6] "POxy: Oxyrhynchus Online," Mar. 2011 [Online]. Available: `http://www.papyrology.ox.ac.uk/POxy/`.

[7] Oxyrhynchus Papyri. Fragments from a group of papyri within the Oxyrhynchus collection in the Sackler Library, Oxford, selected for experimentation by the Oxford Imaging Papyri Multi-spectral Laboratory, Sept. 15, 2010.

[8] MATLAB R2009b, The MathWorks Inc., Natick, MA, 2009.

[9] G. A. Ware, "MSI data report," Imaging Papyri Project, Brigham Young University, Technical Report, 2008:6, 2006.

[10] C. Pohl and J. L. Van Genderen, "Multisensor image fusion in remote sensing: concepts, methods and applications," *International Journal of Remote Sensing*, vol. 19, no. 5, pp. 823–854, 1998.

[11] A. A. Goshtasby and S. Nikolov, "Image fusion: advances in the state of the art," *Information Fusion*, vol. 8, no. 2, pp. 114–118, 2007.

[12] M. Ehlers, "Multisensor image fusion techniques in remote sensing," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 46, no. 1, pp. 19–30, 1991.

[13] C. M. Chen, G. F. Hepner, and R. R. Forster, "Fusion of hyperspectral and radar data using the IHS transformation to enhance urban surface features," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 58, no. 1-2, pp. 19–30, 2003.

[14] L. Jun and Z. Baoming, "An accelerated IHS transform fusion of remote sensing image data based on GPU," in *International Conference on Environmental Science and Information Application Technology, ESIAT*, pp. 492–496, 2009.

[15] C. R. Meyer, J. L. Boes, B. Kim, P. H. Bland, K. R. Zasadny, P. V. Kison, K. Koral, K. A. Frey, and R. L. Wahl, "Demonstration of accuracy and clinical versatility of mutual information for automatic multimodality image fusion using affine and thin-plate spline warped geometric deformations," *Medical Image Analysis*, vol. 1, no. 3, pp. 195–206, 1997.

[16] L. Yang, B. Guo, and W. Ni, "Multimodality medical image fusion based on multiscale geometric analysis of contourlet transform," *Neurocomputing*, vol. 72, no. 1-3, pp. 203–211, 2008.

[17] Y. Yang, D. Park, S. Huang, and N. Rao, "Medical image fusion via an effective wavelet based approach," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 44:1–44:13, 2010.

[18] A. A. Goshtasby, *2-D and 3-D Image Registration: for Medical, Remote Sensing, and Industrial Applications*. New York: Wiley-Interscience, 2005.

[19] G. Pajares and M. de la Cruz, "A wavelet-based image fusion tutorial," *Pattern Recognition*, vol. 37, no. 9, pp. 1855–1872, 2004.

[20] R. Constable and D. Spencer, "Composite image formation in Z-shimmed functional MR imaging," *Magnetic Resonance in Medicine*, vol. 42, no. 1, pp. 1522–2594, 1999.

[21] A. Waxman, M. Aguilar, D. Fay, D. Ireland, and J. Racamato, "Solid-state color night vision: fusion of low-light visible and thermal infrared imagery," *Lincoln Laboratory Journal*, vol. 11, no. 1, pp. 41–60, 1998.

[22] P. Steele and P. Perconti, "Part task investigation of multispectral image fusion using gray scale and synthetic color night-vision sensor imagery for helicopter pilotage," in *Proceedings of SPIE*, vol. 3062, pp. 88–100, 1997.

[23] J. Yang and R. S. Blum, "A statistical signal processing approach to image fusion using Hidden Markov Models," in *Multi-Sensor Image Fusion and Its Applications*, Z. Liu and R. S. Blum, Eds. London: CRC Press, 2006, pp. 265–287.

[24] N. Mitianoudis and T. Stathaki, "Pixel-based and region-based image fusion schemes using ICA bases," *Information Fusion*, vol. 8, no. 2, pp. 131–142, 2007, special Issue on Image Fusion: Advances in the State of the Art.

[25] J. Yang and R. Blum, "Image fusion using the expectation-maximization algorithm and a hidden Markov model," in *IEEE 60th Vehicular Technology Conference, VTC2004-Fall*, vol. 6, 2004.

[26] Adobe Photoshop CS4, Adobe Systems Inc., San Jose, CA, 2010.

[27] J. Gruschel, "Blend modes," Jan. 2011, `http://www.pegtop.net/delphi/articles/blendmodes/`.

[28] Y. Zheng, E. Essock, B. Hansen, and A. Haun, "A new metric based on extended spatial frequency and its application to DWT based fusion algorithms," *Information fusion*, vol. 8, no. 2, pp. 177–192, 2007.

[29] D. Scribner, J. Schuler, P. Warren, G. Howard, and R. Klein, "Melding Images for Information," *SPIE's OE Magazine*, pp. 24–26, Sept. 2002 [Online]. Available: `http://dx.doi.org/10.1117/2.5200209.0004`.

[30] P. Chavez, S. Sides, and J. Anderson, "Comparison of three different methods to merge multiresolution and multispectral data- Landsat TM and SPOT panchromatic," *Photogrammetric Engineering and Remote Sensing*, vol. 57, no. 3, pp. 295–303, 1991.

[31] A. Pellemans, R. Jordans, and R. Allewijn, "Merging multispectral and panchromatic SPOT images with respect to the radiometric properties of the sensor," *PE & RS-Photogrammetric Engineering and Remote Sensing*, vol. 59, no. 1, pp. 81–87, 1993.

[32] K. Amolins, Y. Zhang, and P. Dare, "Wavelet based image fusion techniques–An introduction, review and comparison," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 4, pp. 249–263, 2007.

[33] D. Yocky, "Image merging and data fusion by means of the discrete two-dimensional wavelet transform," *Journal of the Optical Society of America A*, vol. 12, no. 9, pp. 1834–1841, 1995.

[34] W. Yao and C. Zhang, "Multi-spectral image fusion method based on wavelet transformation," in *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 1261–1266, Part B7, 2008 [Online]. Available: `http://www.isprs.org/proceedings/XXXVII/congress/7_pdf/6_WG-VII-6/37.pdf`.

[35] H. Li, B. Manjunath, and S. Mitra, "Multisensor image fusion using the wavelet transform," *Graphical models and image processing*, vol. 57, no. 3, pp. 235–245, 1995.

[36] Z. Zhang and R. Blum, "A categorization of multiscale-decomposition-based image fusion schemes with a performance study for a digital camera application," *Proceedings of the IEEE*, vol. 87, no. 8, pp. 1315–1326, 1999.

[37] O. Rockinger, "Metapix: solutions beyond the pictures," Jan. 2011, `http://www.metapix.de/`.

[38] A. Toet, J. Valeton, and L. van Ruyven, "Merging thermal and visual images by a contrast pyramid," *Optical Engineering*, vol. 28, no. 7, pp. 789–792, 1989.

[39] A. Toet, "Image fusion by a ratio of low-pass pyramid," *Pattern Recognition Letters*, vol. 9, no. 4, pp. 245–253, 1989.

[40] E. Hall, "Computer image processing and recognition," *NASA STI/Recon Technical Report A*, vol. 81, p. 13069, 1979.

[41] W. Morrow, R. Paranjape, R. Rangayyan, and J. Desautels, "Region-based contrast enhancement of mammograms," *IEEE Transactions on Medical Imaging*, vol. 11, no. 3, pp. 392–406, 1992.

[42] T. Pu and G. Ni, "Contrast-based image fusion using the discrete wavelet transform," *Optical Engineering*, vol. 39, no. 8, pp. 2075–2082, 2000 [Online]. Available: `http://link.aip.org/link/?JOE/39/2075/1`.

[43] A. Gillespie, A. Kahle, and R. Walker, "Color enhancement of highly correlated images. I. Decorrelation and HSI contrast stretches," *Remote Sensing of Environment*, vol. 20, no. 3, pp. 209–235, 1986.

[44] R. Haydn, G. Dalke, J. Henkel, and J. Bare, "Application of the IHS color transform to the processing of multisensor data and image enhancement," in *Proceedings of the International Symposium on Remote Sensing of Arid and Semi-Arid Lands, Cairo, Egypt*, pp. 599–616, 1982.

[45] T. Tu, S. Su, H. Shyu, and P. Huang, "A new look at IHS-like image fusion methods," *Information Fusion*, vol. 2, no. 3, pp. 177–186, 2001.

[46] B. R. Foy, "Overview of target detection algorithms for hyperspectral data," Los Alamos Report LA-UR-09-00593, Technical Report, June 2009.

[47] D. Manolakis, "Taxonomy of detection algorithms for hyperspectral imaging applications," *Optical engineering*, vol. 44, pp. 66 403–1–11, 2005.

[48] C. Chang, *Hyperspectral Data Exploitation: Theory and Applications*. New York: Wiley-Blackwell, 2007.

[49] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2001.

[50] T. A. Griffiths, G. A. Ware, T. K. Moon, and J. H. Gunther, "Bleed-through removal using multispectral image data," *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, 2010, [In Press].

[51] G. Sharma, "Show-through cancellation in scans of duplex printed documents," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 736–754, 2001.

[52] E. Dubois and A. Pathak, "Reduction of bleed-through in scanned manuscript documents," in *Proceedings of the IS&T Image Processing, Image Quality, Image Capture Systems Conference*, pp. 177–180, Montreal, 2001.

[53] Q. Wang and C. L. Tan, "Matching of double-sided document images to remove interference," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1084–1089, 2001.

[54] M. R. Khan, H. Imtiaz, and M. K. Hasan, "Show-through correction in scanned images using joint histogram," *Signal, Image and Video Processing*, vol. 4, pp. 337–351, 2010.

[55] C. L. Tan, R. Cao, and P. Shen, "Restoration of archival documents using a wavelet technique," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1399–1404, 2002.

[56] R. F. Moghaddam and M. Cheriet, "Low quality document image modeling and enhancement," *International Journal on Document Analysis and Recognition*, vol. 11, no. 4, pp. 183–201, 2009.

[57] L. T. Duarte, C. Jutten, and S. Moussaoui, "Bayesian source separation of linear and linear-quadratic mixtures using truncated priors," *Journal of Signal Processing Systems*, in press DOI 10.1007/s11265-010-0488-3.

[58] Q. Wang, T. Xia, L. Li, and C. L. Tan, "Document image enhancement using directional wavelet," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, p. 534. Los Alamitos, CA, USA: IEEE Computer Society, 2003.

[59] G. Leedham, S. Varma, A. Patankar, and V. Govindarayu, "Separating text and background in degraded document images–a comparison of global threshholding techniques for multi-stage threshholding," in *International Workshop on Frontiers in Handwriting Recognition*, pp. 244–249. Los Alamitos, CA, USA: IEEE Computer Society, 2002.

[60] J. Wang, M. S. Brown, and C. L. Tan, "Accurate alignment of double-sided manuscripts for bleed-through removal," in *IAPR International Workshop on Document Analysis Systems*, pp. 69–75. Los Alamitos, CA, USA: IEEE Computer Society, 2008.

[61] A. Tonazzini, L. Bedini, and E. Salerno, "Independent component analysis for document restoration," *International Journal on Document Analysis and Recognition*, vol. 7, no. 1, pp. 17–27, 2004.

[62] A. Tonazzini, E. Salerno, and L. Bedini, *Bleed-through Removal from Degraded Documents Using a Color Decorrelation Method*, ser. Lecture Notes in Computer Science, vol. 3163/2004, pp. 229–240. Berlin/Heidelberg: Springer, 2004.

[63] A. Tonazzini, L. Bedini, and E. Salerno, "A Markov model for blind image separation by a mean-field EM algorithm," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 473 –482, 2006.

[64] A. Tonazzini, E. Salerno, and L. Bedini, "Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique," *International Journal on Document Analysis and Recognition*, vol. 10, no. 1, pp. 17–25, June 2007.

[65] F. Drira and H. Emptoz, "A recursive approach for bleed-through removal," in *First International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2005)*, D. S. D. Koichi Kise, Ed., pp. 119–126, Aug. 2005.

[66] F. Merrikh-Bayat, M. Babaie-Zadeh, and C. Jutten, "A nonlinear blind source separation solution for removing the show-through effect in the scanned documents," in *Proceedings of the 16th European Signal Processing Converence, EUSIPCO*, Aug. 2008.

[67] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, May 1999.

[68] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. New York: Wiley, 2002.

Appendices

Appendix A

MSI Enhancement Functions for MATLAB

A.1 Creating an Image Cube

```
function [cube,m,n,b] = makeCube(names)
%creates an image cube from a cell array of image names rather than a text
%file listing the image names.  If there is not an input, a cube will be
%formed from all of the images in the current directory.  If a string is
%recieved all images starting with the string will be used.
%
%names can be obtained using:
%list = dir('pic*.tif');   will select all images starting with 'pic'
%names = {list.name};
%[cube,m,n,b] = makeCube(names);
%

if(~nargin)
    list = dir('*.tif');
    names = {list.name};
elseif(ischar(names))
    list = dir([names,'*.tif']);
    names = {list.name};
elseif(isstruct(names))
    names = {names.name};
end

b = length(names);
pic = imread(char(names(1)));
[m,n] = size(pic);
cube = zeros(m,n,b);
cube(:,:,1) = pic;

for i = 2:b
    cube(:,:,i) = imread(char(names(i)));
end
```

A.2 Image Fusion Functions

The function IMBLEND was written to perform image blending as well as color composites. The image fusion toolbox available at www.metapix.de [37] was used for wavelet and pyramid image fusion.

PCA

```
function [pca,white,E,D] = fastpca(cube,pcs)
%perfoms PCA the same way that fastICA does, pcs is the number of
%principal components to keep.  CUBE can be in image or vector formats.
%PCS indicates how many princial components to keep.

[m,n,b] = size(cube);
if b>1
    cube = reshape(cube,m*n,b);
elseif m<n
    cube = cube';
end

[vectors,means] = remmean(cube'); %removes the mean
R = cov(vectors',1);
if(nargin>1)
    [E,D] = eigs(R,[],pcs);
else
    [E,D] = eig(R);
end
[d,indx] = sort(diag(D)); %sort the eigenvalues from greatest to least

white = inv(sqrt(d))*E(:,indx)'; %whitening matrix
unwhite = E*sqrt(D); %unwhitening matrix
pca = white*vectors; %pca vectors sorted from first to last
```

Blending

```
function pic = imblend(base,blend,method,class)
%Performs image fusion on the two input images (not in vector format).
%
%inputs:
%   BASE - the base image to be fused with BLEND.  If BASE is a stacked
%       image cube (the third dimension is >1) then each image in the cube
% will be fused with BLEND using the selected METHOD.  A stacked cube
%       cannot be used when the RGB color composite methods are selected.
%   BLEND - the image to be fused with BASE.
%   METHOD - the fusion rule to be used during the fusion.  METHOD can be
% any of the following:
```

```
%             'ave','min','max','multiply','screen','overlay','hardlight',
%             'softlight','stamp','rootsum'
%   CLASS - indicates whether to interpret the images as 8 or 16 bits,
%           which determines what the maximum value is.
% This is an optional input that can be either 'uint8' or
% 'uint16'.  The default is 'uint16'.
% outputs:
%   PIC - has the same dimensions as BASE, unless 'red','blue',or 'green'
%       are selected as the fusion METHOD.  In this case the output image
%       is a three band composite image.

[m,n,b] = size(base);

%find out what class the images are
if(nargin<4);
    limit = 65535;
    class = 'uint16';
elseif(strcmp(class,'uint8'))
    limit = 255;
elseif(strcmp(class,'uint16'))
    limit = 65535;
else
    error('CLASS must be either uint8 or uint16');
end

%find if RGB composite mothods are selected, allocate output
rgb = 0;
if(strcmp(method,'red') || strcmp(method,'blue') || strcmp(method,'green'))
    rgb = 1;
    pic = zeros(m,n,3);
else
    pic = zeros(m,n,b);
end

%recursively perform blending on image cubes
if (b>1)
    if(rgb)
        error('RGB composites cannot be perfomed on image cubes.\n ...
        Base must be one image');
    end
    pic(:,:,2:end) = imblend(base(:,:,2:end),blend,method,class);
end

%vectorize image
base = double(reshape(base(:,:,1),m*n,1));
blend = double(reshape(blend,m*n,1));
```

```
mid = round(limit/2);
p = zeros(m*n,1);

rgb = 0;
switch method
    case 'ave'
        p = (blend+base)/2;
    case 'min'
        D = (base<blend);
        p = base.*D+blend.*(~D);
    case 'max'
        D = (base>blend);
        p = base.*D+blend.*(~D);
    case 'multiply'
        p = base.*blend;
    case 'screen'
        p = limit-(limit-base).*(limit-blend);
    case 'overlay'
        ind = find(base>mid);
        ind2 = find(base<=mid);
        p(ind,1) = limit-((limit-base(ind)).*
         (limit-blend(ind)))/mid;
        p(ind2,1) = base(ind2).*blend(ind2)/mid;
    case 'hardlight'
        ind = find(base<mid);
        ind2 = find(base>=mid);
        p(ind,1) = limit-((limit-base(ind)).*
         (limit-blend(ind)))/mid;
        p(ind2,1) = base(ind2).*blend(ind2)/mid;
    case 'softlight'
        temp = base.*blend/limit;
        p = temp+base.*(limit-((limit-base).*
         (limit-blend)/limit)-temp)/limit;
    case 'stamp'
        p = base+2*blend - limit;
        p(p<0) = 0;
        p(p>limit) = limit;
    case 'rootsum'
        base = base-mean(base);
        blend = blend-mean(blend);
        p = sqrt(base.^2+blend.^2);
    case 'red'
        pic = cat(3,blend,base,base);
        rgb = 1;
    case 'green'
```

```
        pic = cat(3,base,blend,base);
        rgb = 1;
    case 'blue'
        pic = cat(3,base,base,blend);
        rgb = 1;
    otherwise
        error('The selected method is not available. ...
        Choose another.');
end

if(rgb)
    pic = reshape(pic,m,n,3);
else
    pic(:,:,1) = reshape(p,m,n);
end
```

A.3 Matched Filtering

```
function out = matched(cube,back,ink,method)
%Performs matched filter detection using an image cube and training data
%indicating the coordinates of known background and ink pixels.
%
%   CUBE - an image cube in any format
%   BACK - coordinates of background pixels
%   INK - coordinates of ink (or target) pixels
%   METHOD - indicates which matched filter method to use, can be any of
%          of the following: 'smf', 'sam', 'cmf', 'tstat', 'fstat', 'ace',
%                            'msf', 'cem', 'lcmv', 'lda', 'qda', 'pca'
%
%Example: out = matched(cube,back,ink,'cem');
%

%flatten the cube, with pixels as columns (wide matrix)
[m,n,b] = size(cube);
if b>1
    flat = reshape(cube,m*n,b)';
elseif m>n
    flat = cube';
else
    flat = cube;
end

mu = mean(flat,2); %cube mean
F = flat-repmat(mu,1,m*n); %cube with mean removed

Nt = size(ink,1);
```

```
Nb = size(back,1);
T = zeros(b,Nt);
B = zeros(b,Nb);

for i = 1:Nt %get ink spectral vectors from the tagged ink coordinates
T(:,i) = squeeze(cube(ink(i,2),ink(i,1),:));
end

for i = 1:Nb %get background spectral vectors from the tagged coordinates
B(:,i) = squeeze(cube(back(i,2),back(i,1),:));
end

mu_t = mean(T,2); %target mean
mu_tm = mu_t-mu;  %target mean with cube mean removed
mu_b = mean(B,2); %background mean

switch method
    case 'smf'
        %Simple Matched Filter
        %with cube mean removed from the the target and cube
        smf = mu_tm'*F;
        out = reshape(smf,m,n);

    case 'sam'
        %Spectral Angle Map
        temp1 = zeros(1,m*n);
        nor = sqrt(mu_t'*mu_t);
        for i = 1:m*n
            temp1(i) = sqrt(flat(:,i)'*flat(:,i));
        end

        %Spectral Angle Map with means NOT removed
        sam = mu_t'*flat./(nor*temp1);
        out = reshape(sam,m,n);

    case 'cmf'
        R = cov(flat');
        Rinv = inv(R);%cov of cube without mean

        %Clutter Matched Filter
        cmf = mu_tm'*Rinv*F;
        out = reshape(cmf,m,n);

    case 'tstat'
        R = cov(flat');
        Rinv = inv(R);%cov of cube without mean
```

```
    %T-statistic
    qm = (mu_t-mu)'*Rinv/(sqrt((mu_t-mu)'*Rinv*(mu_t-mu)));
    tstat = zeros(1,m*n);
    for i = 1:m*n
        tstat(i) = (qm*F(:,i))/sqrt(F(:,i)'*Rinv*F(:,i)-(qm*F(:,i))^2);
    end
    out = reshape(tstat,m,n);

case 'fstat'
    G = corr(flat');
    Ginv = inv(G);

    %F-statistic
    sqr2 = zeros(1,m*n);
    %rho = 1/sqrt((mu_t-mu)'*Rinv*(mu_t-mu));
    rho2 = 1/sqrt((mu_t)'*Ginv*(mu_t));
    for i = 1:m*n
        sqr2(i) = flat(:,i)'*Ginv*flat(:,i);
    end
    fstat2 = (mu_t'*Ginv*flat).^2./(sqr2-rho2^2*(mu_t'*Ginv*flat).^2);
    out = reshape(fstat2,m,n);

case 'ace'
    R = cov(flat');
    Rinv = inv(R);%cov of cube without mean

    %Adaptive Coherence Estimator
    nor = sqrt(mu_t'*Rinv*mu_t);
    temp2 = zeros(1,m*n);
    for i = 1:m*n
     %cov cube without mean
        temp2(i) = sqrt(F(:,i)'*Rinv*F(:,i));
    end
    ace3 = mu_tm'*Rinv*F./(nor*temp2);
    out = reshape(ace3,m,n);

case 'msf'
    %Matched Subspace Filter, means are NOT removed
    P = (T'*T)\T'; %T is the stacked target spectral vectors
    P = T*P;
    msd = zeros(m*n,1);
    for i = 1:m*n
        msd(i) = flat(:,i)'*P*flat(:,i);
    end
    out = reshape(msd,m,n);
```

```
case 'cem'
    G = corr(flat');
    Ginv = inv(G);

    %Constrained Energy
    y = mu_tm'*Ginv*F/(mu_tm'*Ginv*mu_tm);
    out = reshape(y,m,n);

case 'lcmv'
    G = corr(flat');
    Ginv = inv(G);

    c = [ones(1,Nt),zeros(1,Nb)]';

    w = (Mm'*Ginv*Mm)\c;
    w = Ginv*Mm*w;
    lcmv = w'*F;
    out = reshape(lcmv,m,n);

case 'lda'
    Rb = cov(B');
    Rbinv = inv(Rb);%cov of background

    dtf = flat'*Rbinv*mu_t-.5*mu_b'*Rbinv*mu_b;
    dbf = flat'*Rbinv*mu_b-.5*mu_t'*Rbinv*mu_t;
    out = reshape(dtf-dbf,m,n);

case 'qda'
    Rb = cov(B');
    Rbinv = inv(Rb);%cov of background
    Rt = cov(T');
    Rtinv = inv(Rt);%cov of target

    [~,ET] = eig(Rt);
    [~,EB] = eig(Rb);
    covdt = .5*sum(log(diag(ET)));
    covdb = .5*sum(log(diag(EB)));
    fppl = log(Nt/(m*n));
    fpbl = log(1-(Nt/(m*n)));
    for i = 1:m*n
        dpfq(i) = fppl-covdt-.5*(flat(:,i)-mu_t).'*
         Rtinv*(flat(:,i)-mu_t);
        dbfq(i) = fpbl-covdb-.5*(flat(:,i)-mu_b).'*
         Rbinv*(flat(:,i)-mu_b);
    end
```

```
        out = reshape(dpfq-dbfq,m,n);

    case 'pca'
        R = cov(flat');
        [V,D] = eig(R);
        [d,indx] = sort(diag(D));
        %return the first principal component
        out = 1/sqrt(d(1))*V(:,indx(1))'*F;
        out = reshape(out,m,n);
    otherwise
        fprintf('Not a valid matched filter method, please try again.');
        out = [];
end
```

A.4 Bleed-Through Removal

```
function out = bleedRemove(one,two,method,bleed,back)
%Removes the bleed through from the first principal component by
%subtracting a scaled version of the second.  Two ways are available to
%calculate the scaling parameter.  The training data files BLEED and BACK
%are not necessary when performing the least squares method.
%
%   ONE - the first principal component image
%   TWO - the second principal component image
%   METHOD - selects which subtraction method to use,
%               1=least squares, 2=solver method
%   BLEED - coordinates to bleed-through pixels.  The matrix should have
%       two columns with the y coordinate as the first and the x as the
% second column.
%   BACK - coordinates to background pixels in the same format as BLEED
%
%examples: out = bleedRemove(pic1,pic2,1);
%          out = bleedRemove(pic1,pic2,2,bleed,back);
%

switch method

    case 1 %least squares method
        [m,n] = size(one);
        %make the images vectors
        one = one(:);
        two = two(:);
        a = one'*two/(two'*two);
        out = reshape(one-a*two,m,n);

    case 2
```

```
        bl = size(bleed,1);
        ba = size(back,1);

        recto = zeros(1,bl);
        verso = zeros(1,bl);

        %select pixels where the bleed-through has been tagged
        for i = 1:bl
        recto(i) = one(bleed(i,2),bleed(i,1));
        verso(i) = two(bleed(i,2),bleed(i,1));
        end

        %select pixels where there is only background
        bb = zeros(1,ba);
        for i = 1:ba
        bb(i) = one(back(i,2),back(i,1));
        end
        Rb = cov(bb);%calculate the covariance of tagged background pixels


        %calculate: min|| cov(recto-a*verso) - Rb ||_2

        A = mean(recto.*recto);
        B = mean(recto.*verso);
        C = mean(verso.*verso);

        mu1 = mean(recto);
        mu2 = mean(verso);

        J = 2*(mu1*mu2-B);
        K = C-mu2^2;
        H = A-Rb;

        syms x
        a = solve( 4*x^3*K^2 + 6*x^2*J*K + 2*x*(J^2+2*K*H) + 2*J*H);
        out = one-eval(a(1))*two;

    otherwise
        fprintf('Not a valid subtraction method: enter 1 or 2.');
end
```

A.5 Shadow Removal Functions

Functions were written to perform QDA and LDA as well as masking their results.

QDA

```
function [qimage,qdata] = myqda(C,targ,back)
%QDA Quadratic Discriminant Analysis for hyperspectral data
%  QDA(C,TARGET,BKGRND,LOOP,PERCENT)
%        C:   Full hyperspectral cube
%   TARG:   Subscripts of the target pixels (y coordinates are in the first
%     column and x coordinates are in the second)
%   BACK:   Subscripts of the background pixels
%
%  PERCENT:   Percent of the data to use for training. Between 0 and 1.
%
% this code was parced from the USU HAT gui written by Cameron Grant


[xdim,ydim,ldim] = size(C);
p = length(targ);
b = length(back);
qimage = zeros(xdim,ydim);
ldata=struct;

N = p+b; %number of samples in training set
Nf = xdim*ydim; %number of pixels in image

%correct the dimensions of the inputs
[x,y] = size(targ);
if (x < y) && (y > 2)
    targ = targ.';
end

[x,y] = size(back);
if (x < y) && (y > 2)
    back = back.';
end

    %Estimate mean for plume
    Cp = zeros(ldim,p);
    for i = 1:p Cp(:,i) = squeeze(C(targ(i,2),targ(i,1),:)); end
    mp = mean(Cp,2);

    %Estimate mean for background
    Cb = zeros(ldim,b);
    for i = 1:b Cb(:,i) = squeeze(C(back(i,2),back(i,1),:)); end
```

```
    mb = mean(Cb,2);

    %Estimate covariance for plume
    Cpr = Cp - repmat(mp,1,p);
    Rp = (Cpr*Cpr.')/(p-1);

    %Estimate covariance for background
    Cbr = Cb - repmat(mb,1,b);
    Rb = (Cbr*Cbr.')/(b-1);

    %maybe this will help with the memory issue
    Rpi = inv(Rp);
    Rbi = inv(Rb);
    %Eigen Decomposition of these
    [~,EP] = eig(Rp);
    [~,EB] = eig(Rb);

    %Since it's a loop, calculate all possible calculations here
    covdtp = .5*sum(log(diag(EP)));
    covdtb = .5*sum(log(diag(EB)));

    %calculate priors
    fppl = log(p/Nf);
    fpbl = log(1-(p/Nf));

    %Construct Xf, the entire image in data array format
    Xf = reshape(shiftdim(C,2),ldim,xdim*ydim); %Flatten

    %Create mean-removed data.  Use same old means
    Xfp = Xf - repmat(mp,1,Nf);
    Xfb = Xf - repmat(mb,1,Nf);

    %Run QDA functions on entire image
    dpfq = zeros(1,Nf);
    dbfq = zeros(1,Nf);
    for i = 1:Nf
        dpfq(i) = fppl-covdtp-.5*Xfp(:,i).'*Rpi*Xfp(:,i);
        dbfq(i) = fpbl-covdtb-.5*Xfb(:,i).'*Rbi*Xfb(:,i);
    end

    %save the image output
    qimage = shiftdim(reshape(dpfq-dbfq,1,xdim,ydim),1);

%save the data in a structure that can be used to
%apply this QDA to another document
qdata.Rpi = Rpi;
```

```
qdata.Rbi = Rbi;
qdata.covdtp = covdtp;
qdata.covdtb = covdtb;
qdata.fppl = fppl;
qdata.fpbl = fpbl;
qdata.mp = mp;
qdata.mb = mb;
```

LDA

```
function [limage,ldata] = mylda(C,targ,back)
%performs LDA on an input image using the coordinates of target and
%background pixels.
%
%     C - input multispectral image cube
%     targ - coordinates to target pixels (y coordinates are in the first
%     column and x coordinates are in the second)
%     back - coordinates to background pixels
%
% this code was parced from the USU HAT gui written by Cameron Grant


[xdim,ydim,ldim] = size(C);
t = length(targ);
b = length(back);
limage = zeros(xdim,ydim);
ldata=struct;

N = t+b; %number of target and background vectors
Nf = xdim*ydim; %number of pixels in the image

%correct the dimensions of the inputs
[x,y] = size(targ);
if (x < y) && (y > 2)
    targ = targ.';
end

[x,y] = size(back);
if (x < y) && (y > 2)
    back = back.';
end

%Estimate mean for target
Cp = zeros(ldim,t);
for i = 1:t Cp(:,i) = squeeze(C(targ(i,2),targ(i,1),:)); end
mp = mean(Cp,2);
```

```
%Estimate mean for background
Cb = zeros(ldim,b);
for i = 1:b Cb(:,i) = squeeze(C(back(i,2),back(i,1),:)); end
mb = mean(Cb,2);

%Calculate priors
pt = t/N;
pb = b/N;

%Common Covariance Matrix
T = [targ;back];
Ct = zeros(ldim,N);
for i = 1:N Ct(:,i) = squeeze(C(T(i,2),T(i,1),:)); end
mt = mean(Ct,2);

Ctr = Ct - repmat(mt,1,N);
Rt = (Ctr*Ctr.')/(N);
Rti = inv(Rt);

%Construct Xf, the entire image in data array format
Xf = reshape(shiftdim(C,2),ldim,xdim*ydim); %Flatten

fpt = t/Nf;
fpb = 1-fpt;

fptl = log(fpt);
fpbl = log(fpb);

%Run Linear Discriminant Functions
dtf = Xf.'*Rti*mp-.5*mp.'*Rti*mp+fptl;
dbf = Xf.'*Rti*mb-.5*mb.'*Rti*mb+fpbl;
limage = shiftdim(reshape(dtf-dbf,1,xdim,ydim),1);

%save the data in a structure that can be used to apply this LDA to another
%similar image.
ldata.Rti = Rti;
ldata.mp = mp;
ldata.mb = mb;
ldata.fptl = fptl;
ldata.fpbl = fpbl;
```

Masking

```
function masking(mask,name)
%Masks out, or makes white, all unwanted pixels in the input image.  The
%new masked image is saved with the same name as the input image, but with
%the word 'masked' added to the end.
%
%   MASK - a black and white image with white pixels indicating those that
%       should be changed to white in the input image.  All pixels in the
%       mask image that are higher than 10,000 will be masked.
%   NAME - the file name of the tiff image that is to be masked.
% It should not include the .tif extension of the file.
%
%Example: masking(maskpic,'image one');
%

[m,n] = size(mask);
mask = reshape(mask,m*n,1);
pic = imread([name,'.tif']); %read in the image to be masked

ind = find(mask>10000); %find white pixels in the mask
pic = reshape(pic,m*n,1); %reshape the input image
pic(ind) = 65535; %change all pixels that are to be masked to white
pic = reshape(pic,m,n);

%save the image with 'masked' added to the name
imwrite(pic,[name,' masked.tif']);
```

Appendix B
IEEE Asilomar Paper

Bleed-Through Removal Using Multispectral Image Data

Trace A. Griffiths*, Gene A. Ware, Todd K. Moon, Jacob H. Gunther
Utah State University
Old Main Hill, 84322-4120
Logan, UT, USA
*Email: t.griff@aggiemail.usu.edu

***Abstract*—Removing bleed-through texts found in digitally scanned or imaged documents is a current need. Many algorithms have been presented to correct bleed-through using grayscale and/or color images. Recent application of multispectral imaging techniques to documents increases the signal information available for bleed-through correction. We utilize the multispectral information from both sides of the document and principal component analysis to correct bleed-through texts in a fast, unsupervised algorithm.**

I. Introduction

Document imaging has become a widespread practice to preserve texts and allow worldwide availability. A common problem that must be corrected in the digital imagery of documents is bleed-through text, in which the text from the verso (back) side is seen from the recto (front) and vice versa. In the literature, many algorithms exist to remove the bleed-though text in both ancient and modern documents. The algorithms can be categorized as either non-blind or blind separations, using information from both sides or only one side, respectively.

Non-blind separation techniques compare the text on the recto and verso sides of the document in order to remove the bleed-through from each side. Registration of the recto and verso images is necessary in order to utilize all the information simultaneously. One of the sides, commonly the verso side, must be horizontally reversed before registration. This allows the bleed-through on the recto image to be registered with the text of the verso image and vice versa. Registration is difficult and time consuming and is the major penalty for using information from both sides. Non-blind techniques have identified bleed-through text by adaptive linear filtering [1], thresholding [2], pixel matching [3], and joint histograms [4].

Once the bleed-though is identified, it is replaced by an approximation of the background, or has been smeared using an iterated wavelet transform technique [5]. A model for ink diffusion has been used to reverse the diffused ink causing bleed-through [6]. A Bayesian source separation was recently proposed [7]. The non-blind algorithms use two grayscale images, one from each side of the document.

Blind separation has the advantage of not requiring an image of the verso side of the document. Computational savings is gained by not registering images but often these algorithms are quite complex and time consuming. Techniques that use only one grayscale image include directional wavelet transforms [8], global thresholding [9], and pixel matching [10]. When information in different wavelength bands is available, such as color images, principal component analysis (PCA) and blind source separation (BSS) techniques have been used [11]–[14]. Recursive K-means clustering following PCA was used in [15]. A non-linear BSS solution has also been used [16].

A common scientific imaging technique called multispectral imaging (MSI), which has also been used for document [17] and painting analysis [18], has not been previously used for bleed-through removal by any of the algorithms proposed in the literature. Some use the three wavelength bands associated with color images but nothing more. MSI provides more spectral information of the subject which could be used for distinguishing the bleed-through from the recto text and removing it. Having MSI data from both sides allows fast and unsupervised techniques to remove the data rather than costly alternatives. We investigate the use of MSI data from both the recto and verso sides of documents for bleed-through correction, combining the blind PCA and BSS techniques with non-blind architecture.

II. Methods

Our MSI data provides b images at specific wavelengths between 400 and 1000 μm. These images can be aligned using image registration so that each pixel corresponds to the same location. Stacking these images in three dimensions creates an image cube with each pixel forming a vector of length b, representing the spectral response of the subject. If each multispectral image has dimension $m \times n$, the image cube will have dimension $m \times n \times b$. An illustration of an image cube is given in Fig. 1.

An image cube can be reshaped from three to two dimensions with each row being an image and each column representing the spectral vector of each pixel. This reshaping has the dimensions $b \times N$ where $N = mn$. A model for bleed-through correction using the $b \times N$ version of the data, can be written as a linear mixing of the sources to form each image:

$$\mathbf{x}(t) = A\mathbf{s}(t) \quad t = 1, 2, ..., N \tag{1}$$

where $\mathbf{x}(t)$ is the measured spectral vector at each pixel location, A is a $b \times M$ mixing matrix where M is the number of sources, and $\mathbf{s}(t)$ as the spectral response of each of the sources that make up the image.

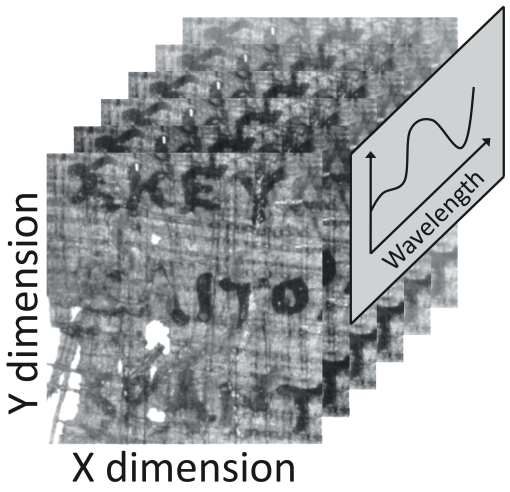


Fig. 1. Image cube representation of multispectral data

Since neither the sources, $\mathbf{s}$, nor the mixing matrix, A, are known, this problem falls into the BSS framework. As has been noted by Tonazzini *et. al.* [14] the linear framework is too simple for the complex problem found in ink seepage, but has still been shown to produce good results. In their research, a three band color image was used, but it was noted that the same process could also be used with MSI data. We test the utility of the decorrelation techniques of PCA and the source separation techniques of independent component analysis (ICA) for correcting bleed-through degradation. These two techniques are well documented in the signal processing literature [19], [20]. We also test a recently proposed symmetric orthogonalization method, similar to PCA but with an added rotation to the whitening matrix [12]. PCA can be defined as:

$$\text{PCA} = W^T\mathbf{x} \tag{2}$$

where $\mathbf{x}$ is the $b \times N$ data matrix with images as row vectors, W^T is the $b \times b$ dimensional whitening matrix defined as $\Lambda_x^{-1/2}V_x^T$ with Λ_x and V_x being the eigenvalues and eigenvectors of the autocorrelation matrix of $\mathbf{x}$, respectively. The symmetric orthogonalization transformation is defined as:

$$\text{Orthogonal} = W_s^T\mathbf{x} \tag{3}$$

where W_s is defined as $V_x\Lambda_x^{-1/2}V_x^T$. The additional eigenvector matrix in front provides a rotation to the common whitening matrix W [12]. All ICA calculations were performed using the fastICA algorithm [19].

All transformations are performed on the two dimensional image cube we will call $\mathbf{x}$. Each transformation returns a matrix the same size as $\mathbf{x}$ with output images as rows. In order to view the output images, they must be transformed back into the original $m \times n$ dimensions of the input images used to form the image cube. They must also be normalized to contain values consistent with grayscale images. The process we used to normalize the images can be written as:

$$Y = (y - \min(y))\,(\max(y) - \min(y)) \times 65535 \tag{4}$$

where y is the data vector, Y is the normalized vector, and 65535 is used for 16-bit images as the maximum possible pixel value. Outputs from the three transformations are only unique to scaling and often must be inverted in order to be viewed correctly.

The ICA and symmetric orthogonalization transformations are derived to extract the independent components of the data. This is ideal for many cases, but with documents there is no reason to assume that the bleed-through and recto text are independent. PCA separates the data components by projecting the data onto an orthogonal basis using the eigenvectors matrix. The first principal component (PC) will have most of the variance in the data and, in the case of bleed-through, contains both recto and bleed-through texts. The second PC will begin to separate the texts due to the difference in spectral characteristics caused by the verso text being filtered through the parchment medium.

Bleed-through can be removed from both the recto and verso texts using the first two PCs by subtracting/adding the first two PCs pixel by pixel. Subtraction provides the recto text and addition the verso, or vice versa. This arithmetic operation can be performed on the PCA output vectors directly or on the normalized grayscale images. A direct subtraction/addition of the PC data vectors was effective in eliminating the bleed-through, but a scaling of the subtracted image could be useful with other data. To find an ideal scaling of the subtracted image, the least squares solution to

$$\min_a \|\text{PC}_1 - a \times \text{PC}_2\|_2^2\,, \tag{5}$$

$$a = \frac{\langle \text{PC}_1, \text{PC}_2 \rangle}{\langle \text{PC}_1, \text{PC}_1 \rangle} \tag{6}$$

can be used, where $\langle\ ,\rangle$ indicates the standard Euclidean inner product. Since the PC data vectors are orthogonal, the inner product of the two PCs is approximately zero. This means the scaling parameter, a, can only be used with the normalized images formed from the PC data vectors using (4) (normalized images must be in vector format for the calculation).

III. Test Results

Two text samples have been selected for analysis. One is a severely deteriorated scroll parchment which contains bleed-through text, folds in the parchment, stains, holes, and large burned sections[1]. The other is a 19^{th}-century document with very slight bleed-through. Three non-blind separation techniques were implemented and compared in this study: PCA, ICA, and symmetric orthogonalization. These three techniques were previously shown to be effective with one sided color images [12]. We verify the algorithms' utility using a stacked image cube formed from multispectral images of both the recto and verso sides of the document. To form the stacked cube, the verso text was horizontally flipped and registered to the recto.

[1]Fragment A, from a group of papyri from the Oxyrhynchus collection in the Sackler Library, Oxford selected for experimentation by the Oxford Imaging Papyri Multi-spectral Laboratory.

A. Ancient Scroll

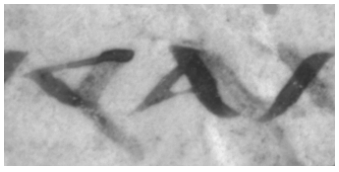
The cropped section of the scroll, seen in Fig. 2, shows the text with strong bleed-through on both sides of the parchment. The results of PCA, ICA, and the symmetric orthogonalization transformations using a stacked image cube are seen in Fig. 6. These methods were also applied to the individual front and back image cubes but were less accurate than the results with the stacked images in all cases. It is apparent that the ICA and symmetric orthogonalization methods were not able to accurately eliminate the bleed-through text. PCA did have promising results due to its ability to separate the two texts into opposite grayscale values in the second PC. PCA was also easier to use due to its ability to consistently place the most important images at first and second of the output order. ICA and symmetric orthogonalization do not have this feature, requiring the user to determine which of all the output images are useful.

The bleed-through removed fragment obtained by directly subtracting/adding the PC data vectors is shown in Fig. 3. The scaling parameter a for the scroll data was calculated to be approximately one and, therefore, did not visually change the result.

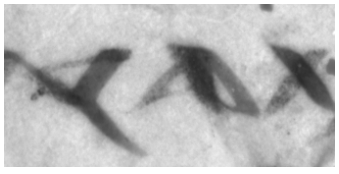
The bleed-through removed fragment obtained by directly subtracting/adding the PC data vectors is shown in Fig. 3. The scaling parameter a for the scroll data was calculated to be approximately one and, therefore, did not visually change the result.

B. 19^{th}-century document

The 19^{th}-century document does not contain strong bleed-through and has a very uniform background compared to the ancient scroll fragment. A section of original text is seen in Fig. 4. Applying PCA, ICA, and the symmetric orthogonalization transformations to the stacked image cube gave similar results as the scroll fragment, but subtracting the PC images did not effectively remove the bleed-through.

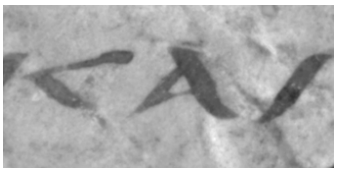


(a)

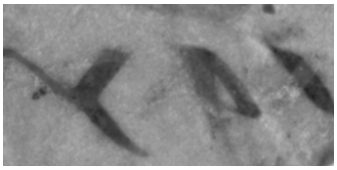


(b)

Fig. 2. A cropped section of the uncorrected recto 2(a) and horizontally flipped verso 2(b) scroll fragment at 700 μm. The dark characters are the recto/verso text with hazy, gray characters as bleed-through. Each cropped section corresponds to the same registered area in both 2(a) and 2(b).



(a)



(b)

Fig. 3. Recto 3(a) and verso 3(b) texts after Bleed-through was removed from scroll by subtracting and adding the first two stacked PC data vectors, respectively.

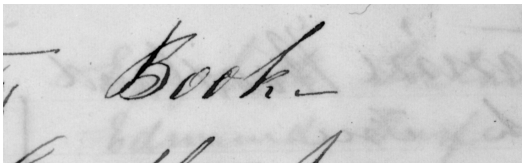


Fig. 4. 19^{th}-century uncorrected document recto at 650 μm. The word *Book* is the overtext and the hazy letters filling the rest of the frame are bleed-through.

Instead, the bleed-through was changed from faint dark text to light text as shown in Fig. 5.

Applying the transformations to the recto text image cube, without information from the verso, gave significantly better results. If the stacked cube had dimensions $m \times n \times 2b$, the recto cube would be the $m \times n \times b$ formed from the recto images. Using the recto image cube, ICA and symmetric orthogonalization were able to remove the bleed-through but tended to be noisy while PCA was still able to separate the two texts in the second PC. The results of each transformation are shown in Fig. 7. Results from adding the PC data vectors, the normalized PC images, and the addition scaled by parameter a in (6) are shown in Fig. 8. All are shown here to demonstrate the output differences between the addition methods. The scaling parameter a was calculated to be 2.3835 using the normalized PC 1 and 2 images. The visually best output in this case is obtained by adding the normalized PC images. Scaling the addition of the normalized PC images is very similar to directly adding the PC data vectors.

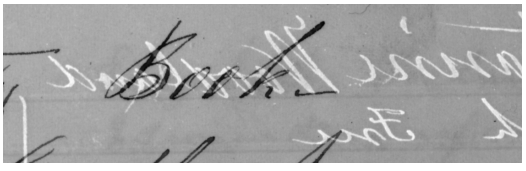
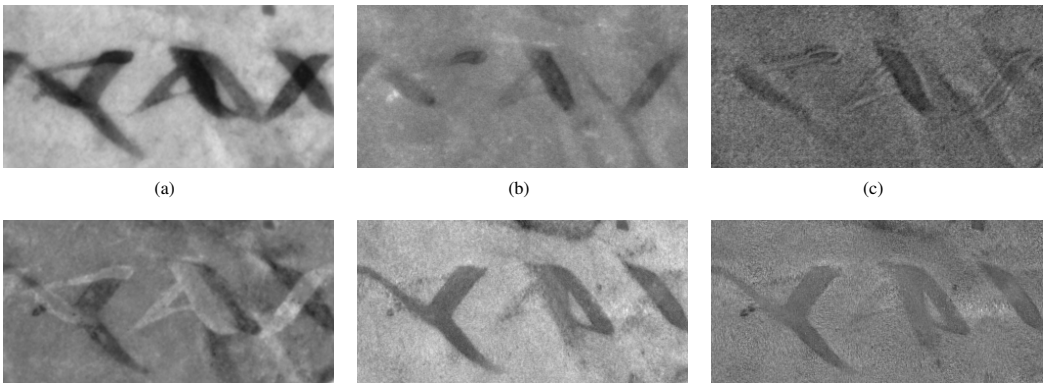


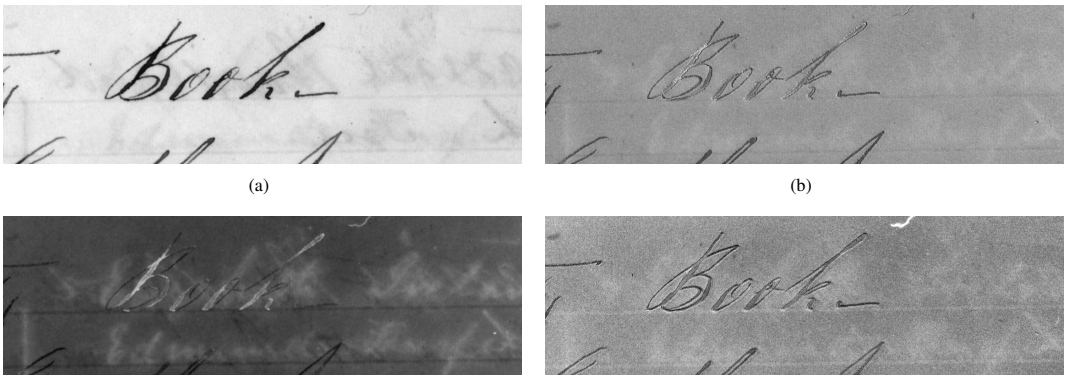
Fig. 5. Bleed-through removed by adding the PC data vectors from the 19^{th}-century document using the stacked image cube.



(a) (b) (c)

(d) (e) (f)

Fig. 6. Scroll results for PCA, ICA, and symmetric orthogonalization using a stacked image cube: Principal components 1 6(a) and 2 6(d), independent components 1 6(b) and 12 6(e), and orthogonalization components 8 6(c) and 15 6(f).



(a) (b)

(c) (d)

Fig. 7. 19^{th}-century document results for PCA, ICA, and symmetric orthogonalization using the recto text image cube: Principal components 1 7(a) and 2 7(c), independent component 6 7(b), and symmetric orthogonalization component 12 7(d)

IV. Conclusion

PCA was the most effective method for removing bleed-through text from both the ancient scroll and 19^{th}-century documents. The first and second PCs were able to separate the bleed-though from the recto and verso texts. Subtracting/adding the two PCs resulted in an image where the bleed-through has been removed. This can be performed on both the PC data vectors or the normalized PC images with good results.

The algorithm worked best for the ancient scroll data using the stacked image cube containing information from both sides of the parchment. This gave the needed information to accurately remove the severe bleed-through deterioration from both the recto and verso text simultaneously. The 19^{th}-century fragment, with much less bleed-though present, produced better results when only using the images of the recto side. When using the images of both sides, the algorithm over-corrected and caused the bleed-through to become white. Using only the recto images gave a better result in this case.

ICA and symmetric orthogonalization transformations were unable to remove the bleed-though in the ancient scroll data. They produced improved images with the 19^{th}-century document but the results were not superior to that obtained using the PCA approach. PCA was able to maintain the visual characteristics of the original document while the other two transformations changed the look dramatically and added noise to the image.

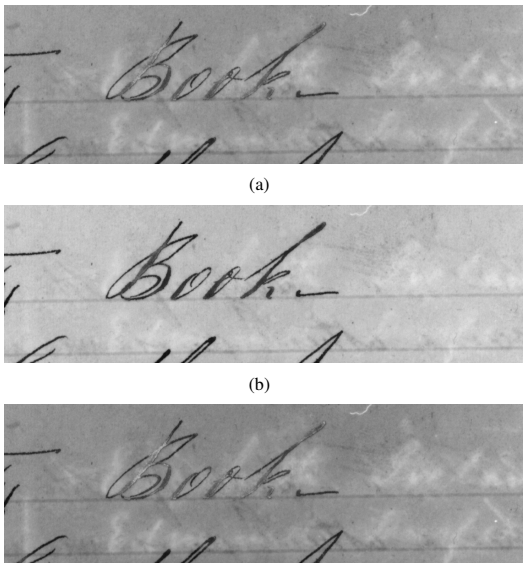
This study shows that when available, MSI data allows bleed-through to be removed from simple and complicated documents. The large amount of data facilitates a fast unsupervised algorithm which is much simpler than those methods proposed for bleed-through removal without MSI data.

References

[1] G. Sharma, "Show-through cancellation in scans of duplex printed documents," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 736–754, 2001.

[2] E. Dubois and A. Pathak, "Reduction of bleed-through in scanned manuscript documents," in *Proceedings of the IS&T Image Processing, Image Quality, Image Capture Systems Conference*, Montreal, 2001, pp. 177–180.

[3] Q. Wang and C. L. Tan, "Matching of double-sided document images to remove interference," in *in IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 1084–1089.



(a)

(b)

(c)

Fig. 8. Bleed-through removed from 19^{th}-century document using three different addition techniques: PC data vector addition 8(a), normalized PC image addition 8(b), and scaled image addition using PC 1 + a(PC 2) with a=2.3835 8(c).

[4] M. R. Khan, H. Imtiaz, and M. K. Hasan, "Show-through correction in scanned images using joint histogram," *Signal, Image and Video Processing*, vol. 4, pp. 337–351, 2010.

[5] C. L. Tan, R. Cao, and P. Shen, "Restoration of archival documents using a wavelet technique," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1399–1404, 2002.

[6] R. F. Moghaddam and M. Cheriet, "Low quality document image modeling and enhancement," *International Journal on Document Analysis and Recognition*, vol. 11, no. 4, pp. 183–201, 2009.

[7] L. T. Duarte, C. Jutten, and S. Moussaoui, "Bayesian source separation of linear and linear-quadratic mixtures using truncated priors," *Journal of Signal Processing Systems*, in press DOI 10.1007/s11265-010-0488-3.

[8] Q. Wang, T. Xia, L. Li, and C. L. Tan, "Document image enhancement using directional wavelet," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, p. 534, 2003.

[9] G. Leedham, S. Varma, A. Patankar, and V. Govindarayu, "Separating text and background in degraded document images–a comparison of global threshholding techniques for multi-stage threshholding," *Frontiers in Handwriting Recognition, International Workshop on*, vol. 0, pp. 244–249, 2002.

[10] J. Wang, M. S. Brown, and C. L. Tan, "Accurate alignment of double-sided manuscripts for bleed-through removal," *Document Analysis Systems, IAPR International Workshop on*, vol. 0, pp. 69–75, 2008.

[11] A. Tonazzini, L. Bedini, and E. Salerno, "Independent component analysis for document restoration," *International Journal on Document Analysis and Recognition*, vol. 7, no. 1, pp. 17–27, 2004.

[12] A. Tonazzini, E. Salerno, and L. Bedini, *Bleed-through removal from degraded documents using a color decorrelation method*, ser. Lecture Notes in Computer Science. Berlin/Heidelberg: Springer, 2004, vol. 3163/2004, pp. 229–240.

[13] A. Tonazzini, L. Bedini, and E. Salerno, "A Markov model for blind image separation by a mean-field EM algorithm," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 473 –482, 2006.

[14] A. Tonazzini, E. Salerno, and L. Bedini, "Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique," *International Journal on Document Analysis and Recognition*, vol. 10, no. 1, pp. 17–25, June 2007.

[15] F. Drira and H. Emptoz, "A Recursive Approach For Bleed-Through removal," in *First International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2005)*, D. S. D. Koichi Kise, Ed., Aug. 2005, pp. 119–126.

[16] F. Merrikh-Bayat, M. Babaie-Zadeh, and C. Jutten, "A nonlinear blind source separation solution for removing the show-through effect in the scanned documents," in *Proc. EUSIPCO*, 2008.

[17] G. Ware, D. Chabries, R. Christiansen, and C. Martin, "Multispectral document enhancement: ancient carbonized scrolls," in *IEEE 2000 International Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000*, 2000, pp. 2486–2488.

[18] S. Baronti, A. Casini, F. Lotti, and S. Porcinai, "Multispectral imaging system for the mapping of pigments in works of art by use of principal-component analysis," *Appl. Opt.*, vol. 37, no. 8, pp. 1299–1309, 1998.

[19] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, May 1999.

[20] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. Wiley, New York, 2002.