

11-13-2007

Design of an Automated Book Reader as an Assistive Technology for Blind Persons

Lu Wang

Florida International University

DOI: 10.25148/etd.FI10022558

Follow this and additional works at: <https://digitalcommons.fiu.edu/etd>

Recommended Citation

Wang, Lu, "Design of an Automated Book Reader as an Assistive Technology for Blind Persons" (2007). *FIU Electronic Theses and Dissertations*. 222.

<https://digitalcommons.fiu.edu/etd/222>

This work is brought to you for free and open access by the University Graduate School at FIU Digital Commons. It has been accepted for inclusion in FIU Electronic Theses and Dissertations by an authorized administrator of FIU Digital Commons. For more information, please contact dcc@fiu.edu.

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

DESIGN OF AN AUTOMATED BOOK READER AS AN ASSISTIVE
TECHNOLOGY TOOL FOR BLIND PERSONS

A dissertation submitted in partial fulfillment of the

requirements for the degree of

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

by

Lu Wang

2007

To: Interim Dean Amir Mirmiran
College of Engineering and Computing

This dissertation, written by Lu Wang, and entitled Design of an Automated Book Reader as an Assistive Technology Tool for Blind Persons, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this dissertation and recommend that it be approved.

Jean H. Andrian

Naphtali Rishe

Armando Barreto

Malek Adjouadi, Major Professor

Date of Defense: November 13, 2007

This dissertation of Lu Wang is approved.

Interim Dean Amir Mirmiran
College of Engineering and Computing

Dean George Walker
University Graduate School

Florida International University, 2007

DEDICATION

I dedicate this thesis to my mom. Without her patience, understanding, support, and most of all love, the completion of this work would not have been possible.

ACKNOWLEDGMENTS

I wish to thank my advisor, Dr. Malek Adjouadi, for his support, patience, and good humor. His gentle but firm direction has been most appreciated. Dr. Malek Adjouadi was particularly helpful in guiding me toward a qualitative methodology. His interest in sense of competence was the impetus for my proposal. From the beginning, he had confidence in my abilities to not only complete a degree, but to complete it with excellence. I extend my gratitude to my committee members Dr. Armando Barreto, Dr. Naphtali Rishe and Dr. Jean Andrian for their valuable input.

I have found my coursework throughout the curriculum and instruction program at FIU to be stimulating and thoughtful, providing me with the tools with which to explore both past and present ideas and research issues.

The research infrastructure of the CATE center and the financial support provided by the National Science Foundation through grants HRD-0317692, CNS-0426125, CNS-0520811, CNS-0540592 and IIS-0308155 made of this research endeavor a successful one, just as the patent application on this topic acknowledges the support of the NSF.

ABSTRACT OF THE DISSERTATION

DESIGN OF AN AUTOMATED BOOK READER AS AN ASSISTIVE
TECHNOLOGY TOOL FOR BLIND PERSONS

by

Lu Wang

Florida International University, 2007

Miami, Florida

Professor Malek Adjouadi, Major Professor

This dissertation introduces a novel automated book reader as an assistive technology tool for persons with blindness. The literature shows extensive work in the area of optical character recognition, but the current methodologies available for the automated reading of books or bound volumes remain inadequate and are severely constrained during document scanning or image acquisition processes. The goal of the book reader design is to automate and simplify the task of reading a book while providing a user-friendly environment with a realistic but affordable system design. This design responds to the main concerns of (a) providing a method of image acquisition that maintains the integrity of the source (b) overcoming optical character recognition errors created by inherent imaging issues such as curvature effects and barrel distortion, and (c) determining a suitable method for accurate recognition of characters that yields an interface with the ability to read from any open book with a high reading accuracy nearing 98%. This research endeavor focuses in its initial aim on the development of an assistive technology tool to help persons with blindness in the reading of books and other bound volumes. But its secondary and broader aim is to also find in this design the

perfect platform for the digitization process of bound documentation in line with the mission of the Open Content Alliance (OCA), a nonprofit Alliance at making reading materials available in digital form. The theoretical perspective of this research relates to the mathematical developments that are made in order to resolve both the inherent distortions due to the properties of the camera lens and the anticipated distortions of the changing page curvature as one leafs through the book. This is evidenced by the significant increase of the recognition rate of characters and a high accuracy read-out through text to speech processing. This reasonably priced interface with its high performance results and its compatibility to any computer or laptop through universal serial bus connectors extends greatly the prospects for universal accessibility to documentation.

TABLE OF CONTENTS

CHAPTER	PAGE
CHAPTER 1 INTRODUCTION	1
1.1 General Statement of the Problem Area.....	1
1.2 Significance of the Study	2
1.3 Research Premise	4
1.4 Research Methodology.....	6
1.5 Summary	8
CHAPTER 2 SYSTEM DESIGN AND PRACTICAL IMPLICATIONS.....	12
2.1 Introduction	12
2.2 Situation of the Current Commercial Book Readers.....	12
2.3 Architecture of the book reader system.....	14
2.3.1 Hardware setup.....	15
2.3.2 Software architecture.....	18
2.4 Marketing Potential	20
CHAPTER 3 THEORETICAL DEVELOPMENTS AND DESIGN	
METHODOLOGY	23
3.1 Introduction	23
3.2 Automated image acquisition.....	24
3.3 Image restoration.....	25
3.3.1 Image distortion due to the camera lens.....	25
3.3.2 Mathematical Modeling of the Barrel Distortion.....	28
3.3.3 Mathematical modeling of the page curvature distortion.....	32
CHAPTER 4 INTEGRATION OF OCR AND TTS PLATFORMS	53
4.1 Introduction	53
4.2 Image classification and recognition.....	53
4.2.1 Image classification.....	53
4.2.2 Image recognition.....	55
4.3 Text-to-Speech synthesis.....	56
4.3.1 TTS Architecture.....	57
4.3.2 TTS development skill	58
CHAPTER 5 INTRODUCING A MATLAB INTERFACE FOR AUTOMATED	
BOOK READER SYSTEM	60
5.1 Introduction	60
5.2 GUI overview	60
5.3 GUI for automatic image acquisition.....	63
5.4 GUI for image restoration and OCR-TTS.....	64
5.4.1 Barrel distortion correction	65
5.4.2 Image segmentation.....	67

5.4.3	Curvature extraction and real height calculation.....	69
5.4.4	Page curvature distortion correction	71
5.4.5	Optical Character Recognition and Text-To-Speech	72
5.4.6	Fully automatic operations	79
CHAPTER 6 RESULTS ANALYSIS OF THE INTEGRATED BOOK READER		
	INTERFACE.....	80
6.1	Introduction	80
6.2	Results and Outcome Assessments	80
6.2.1	Computational Requirements.....	83
6.2.2	Recognition performance analysis	88
CHAPTER 7 CONCLUSION.....		
7.1	Major Outcomes.....	99
7.2	Prospective Research Endeavors.....	100
7.3	Summary	103
LIST OF REFERENCES		
VITA.....		
		110

LIST OF TABLES

TABLE	PAGE
Table 2-1 Comparison of the potential book reader with leading commercial product in the market	22
Table 6-1 Time consumed in seconds for image acquisition and distortion correction ...	83
Table 6-2 Time processing requirements comparison in OCR and Spell check by the proposed Book Reader Interface in Contrast to ABBYY and OmniPage	88
Table 6-3 Comparison of Interface WinOCR recognition rate of original and corrected images	90
Table 6-4 Comparison of ABBYY recognition rate of original and corrected image	91
Table 6-5 Recognition rate for glossy book with flash off	92
Table 6-6 Recognition rate for glossy book with flash on	92
Table 6-7 Recognition for different font family	95

LIST OF FIGURES

FIGURE	PAGE
Figure 2-1 Sketch of a commercial book scanner from Wikipedia	13
Figure 2-2 Book reader software architecture	14
Figure 2-3 System architecture of a novel book reader	15
Figure 2-4 Book reader system hardware setup.....	17
Figure 2-5 Software architecture of the book reader system	19
Figure 2-7 (a) 4DigitalBooks (b) BookEye book scanner (c) ATIZ BookDrive (high end) and (d) ATIZ BookDrive DIY (low end).....	21
Figure 3-1: Image Distortions Introduced by Photographic Lenses	26
Figure 3-2 Barrel distorted image compared with original image	27
Figure 3-3 Barrel Distorted Image and its correction	29
Figure 3-4 Sample of document image with page curvature distortion.....	32
Figure 3-5 Representation of the overhead and lateral images.....	35
Figure 3-6 Image segmentation and extraction of the boundaries of the book holder in the overhead camera image (left) and the lateral camera image (right)	36
Figure 3-7 Matching points from the overhead and lateral images	38
Figure 3-8: Results illustrating the curvature extraction process.....	40
Figure 3-9 Another approach to obtain the edge of the book curvature	42
Figure 3-10 Front view and lateral view of the whole system with size measure	45
Figure 3-11 Push operation for the overhead image	46
Figure 3-12 Illustration of pixel distance for images with (a) even height and (b) odd height.....	49
Figure 3-13 Section view of the page curvature	52

Figure 4-1 Architecture of document image classification.....	54
Figure 4-2 Text-To-Speech Architecture.....	57
Figure 4-3 Microsoft TTS Software Development Kit 5.3.....	59
Figure 5-1 GUI for the book reader system.....	61
Figure 5-2 GUI for automatic image acquisition.....	63
Figure 5-3 Output options of the main GUI.....	64
Figure 5-4 Barrel distortion correction parameter configuration.....	65
Figure 5-5 GUI for Barrel correction result comparison	66
Figure 5-6 Segmentation parameters configuration.....	67
Figure 5-7 Segmentation window to display the segmented overhead and lateral image	68
Figure 5-8 Curvature Detection Filter section in the main GUI.....	69
Figure 5-9 Curvature extraction result.....	69
Figure 5-10 Plot of the height of each point on the curve to the overhead camera	70
Figure 5-11 Operation warning messages and page curvature distortion correction invoking button	71
Figure 5-12 Comparison between the original and corrected images.....	72
Figure 5-13 GUI for document image analysis toolkit	73
Figure 5-14 Training section of GUI for document image analysis	74
Figure 5-15 Feature extraction result of the image analysis	74
Figure 5-16 RLSA result for the sample document image	75
Figure 5-17 Training result	76
Figure 5-18 Simulation result for document image analysis	78
Figure 5-19 A win32 OCR and TTS embedded in the interface with processing messages	79

Figure 6-1 (a) Distorted image directly from camera (b) Corrected image of the proposed algorithm	81
Figure 6-2 BookEye scan result with cropping (a) original image and (b) book fold corrected image.....	81
Figure 6-3 BookEye Scan of (a) original image and (b) book fold correct image	82
Figure 6-4 Processing time for any image with dimension of 2304*3072	85
Figure 6-5 Processing time as a function of image size.....	86
Figure 6-6 Readout time versus the number of words in the 2 pages.....	87
Figure 6-7 Plot of Recognition rate of ABBYY and WinOCR	89
Figure 6-8 Overhead capture of the glossy book with flash on and the detail of the over-exposure part	93
Figure 6-9 Overhead capture of the glossy book with flash on and the detail of a part of the image.....	94
Figure 6-10 Original image (Left) and corrected image (Right) with multi-font and multi-size, including Helvetica Font, and New Roman.....	96
Figure 6-11 A document page with different gray scales parts of text.....	97

LIST OF SYMBOLS

SYMBOL	PAGE
θ Angle in the polar coordinate system	29
σ The standard deviation	41
\mathbf{C}_ℓ Vector of pixels between curvature and center line for the lateral image	46
\mathbf{C}_o Vector of pixels between curvature and center line for the overhead image	46
\mathbf{D}_{dist} Pixel Distance Vector	46
f Focal Length	47
\mathbf{f} Constant Vector with the value f	51
\mathbf{H}_c Vertical distance between each pixel on the curve and the overhead camera	46
$\mathbf{H}_{c\ell}$ Distance between each pixel on the curve and the center of lateral camera	46
\mathbf{H}_h Distance between the top of the book holder and the overhead camera	49
\mathbf{H}_ℓ Distance between the bottom of the book holder and the lateral camera	46
\mathbf{H}_o Distance between the bottom of the book holder and the overhead camera	46
I Original Image	39
I' Aligned Image	39
K_o Image height of the overhead image	41
K_ℓ Image height of the lateral image	41
L Real width of the book holder	40
M Image width of the aligned overhead image	40
N Image width of the aligned lateral image	40

P^C	Lens barrel distortion parameter vector	32
Q_{cor}	Matrix of vertical index for the corrected image	52
Q_{dist}	Matrix of vertical index for the distorted image	52
r	Length in the polar coordinate system	30
r_{dist}	Length in the polar coordinate system of the distorted image	31
r_{cor}	Length in the polar coordinate system of the distorted image	31
V	Constant Matrix	53
W	Modification weight vector	51
X	Real distance of any point to the left edge of the book holder	40
x_o	Pixel number of any point to the left edge of the overhead image	40
x_l	Pixel number of any point to the left edge of the overhead image	40
x_{dist}	Horizontal index of any point in the distorted image	33
x_{cor}	Horizontal index of any point in the distorted image	33
y_{cor}	Vertical index of any point in the corrected image	33
y_{dist}	Vertical index of any point in the corrected image	33
Z_l	Distance between the book holder and the lateral camera	47

CHAPTER 1

INTRODUCTION

1.1 General Statement of the Problem Area

This dissertation introduces an integrated design of a new automated book reader as an assistive tool for persons with blindness. The literature shows extensive work in the area of optical character recognition, but the current methodologies available for the automated reading of books or bound volumes in general are either inadequate or severely constrained during document acquisition. Standard scanners require human intervention with much pressure on the book to make the scanning image good enough for Optical Character Recognition (OCR). This is undoubtedly an inefficient way for image acquisition. Advanced image acquisition projects, like Project Gutenberg, Google Book Search, and the Open Content Alliance will scan books on a large scale with much more efficiency. However, the book curvature distortion remains an unresolved issue, which severely curtails the character recognition rate in using the OCR modules.

The goal of the book reader interface is to automate and simplify the task of reading a document or a book while providing a user-friendly environment with a realistic but affordable system design that takes a soft approach to page curvature and other image distortions. The interface also responds to the main concerns of (a) providing a method of image acquisition that maintains the integrity of the source (b) overcoming optical character recognition errors created by pertinent imaging issues such as curvature

effects and barrel distortion, and (c) determining a suitable method for accurate recognition of characters that yields an interface with the ability to read from any open book with a high reading accuracy nearing 98%.

1.2 Significance of the Study

It is estimated that 1,100,000 people are legally blind in the US and that 42,000,000 people are blind worldwide (Research to Prevent Blindness, National Institute of Special Education (NISE), and the National Science Foundation (NSF)). With the potential to affect accessibility to this large population of individuals, the significance of this study is related directly to the extent of which the functional capability of the blind user is augmented, and consequently serve as best practice for broadening access to reading materials and participation of persons with blindness in education. Such access will extend significantly the current technology which focuses primarily on scanned documents.

This research endeavor focuses in its initial aim on the development of an assistive technology tool to help persons with blindness in the reading of books and other bound volumes. But its secondary and broader aim is to also find in this design the perfect platform for the digitization process of bound documentation in line with the mission of the Open Content Alliance (OCA), a nonprofit Alliance at making reading materials available in digital form.

In this digital era, hundreds of millions of books adorn the shelves of many libraries around the world, with an unacceptably small amount of them available in

digital form for the blind as well as for the sighted. It is unconceivable therefore to stay indifferent and keep relying on standard scanning procedures as well as on the modern scanning devices, which are costly, time consuming and error prone to page curvature.

The first fully automatic book-scanning robotic device was produced by 4DigitalBooks in 2003 [Markoff, 2003]. This device is a first attempt at integrating the page turning, scanning and OCR functions in one system.

More recent advances have led to a sophisticated commercially available solution proposed with the ATIZ DIY scanner, which makes use of digital cameras. At its present state, the ATIZ scanner with its 55lb weight and a \$35,000 price is not yet an affordable option to an individual user. It is certainly a good start at bringing the technology together as one approach to the scanning of bound volumes [ATIZ, 2007]. Section 2.4 provides additional details on this and other types of scanners.

Google Book Search is introduced in 2004. As of March 2007, the New York Times reported that Google has already digitized one million volumes at an estimated cost of US\$5 million [Hafner K., 2007]. Google uses its own approach to scanning, OCR and proofreading high volume documents. But it is worthy to note that according to [Coyle, 2006] “In fact, the books currently available through the Google Book Project show quality control issues such as missing page images and blurred or unreadable images.”

Furthermore, Microsoft has its own online book search, but its access is limited and only select pages of each book are scanned for viewing purposes (<http://search.live.com/results.aspx?q=&scope=books#q=&filter=all>). Microsoft is also

one of the corporate partners in the Open Content Alliance (OCA). This constitutes an incredible effort at scanning all materials that are made available to this Alliance as open access material. Our intent is evidently in line with the mission of this Alliance.

Nonetheless, all scanning machines at this technological juncture require some form of intervention on the book to make the pages as flat as practically possible in order to obtain high quality images to be fed to the OCR. These interventions remain costly and with mixed results often contending with the lack of efficiency in the image acquisition process in order to overcome the problem of page curvature.

1.3 Research Premise

The research premise is to design a fully integrated system that is relatively fast and yet inexpensive and effective so that any book can be read with a high degree of accuracy. The primary focus of this research will be on reading text. Therefore, the research scope of this dissertation will not include such things as pictures, figures, maps and other type of graphics. Legends of these figures/maps/images can be read however through such a proposed design. The proposed interface will prove that the processes of image acquisition, image restoration, and automated text reading are all seamlessly integrated into an effective book reading design.

The main intent is to demonstrate that through the use of two inexpensive light weight cameras, a handily made book holder and regular lighting as available in an office, room, or library environment, the proposed integrated book reader design will be

able to address through an assimilation of hardware design and software development the following critical issues:

- Establishing the mathematical foundation of image distortion introduced by
 - (a) the curvature of the pages of an open book, and
 - (b) the barrel effect introduced by the image capture of the camera;
- Preprocessing images for finding lines and characters in a given image for appropriate image segmentation;
- Embedding an automatic word correction software based on the matching of words with the highest probability of occurrence; and
- Providing a reading output through synthesized voice to the blind person.

To properly integrate the software developments to the hardware design, the following issues are resolved:

- Registering the two images taken by the top and the lateral camera.
- Ascertaining the mathematical parameters for barrel distortion correction.
- Determining the depth between the overhead camera and each pixel of the curved pages through edge detection of the curvature.
- Correcting the perspective effect inherent to images.
- Achieving a high signal to noise ratio (de-noising) after image restoration.
- Hastening the process of linking the image acquisition process to those of image restoration.

- Text recognition while foregoing any neural network approach [Strouthopoulos and Papamarkos, 1998, Adjouadi et al. 2006] but using free and highly effective OCR software.

1.4 Research Methodology

In this project two cameras are used for image acquisition. So the first in determining matching points on the two images is a critical step for satisfying the uniqueness constraint and proceed with further processing. A simple and inexpensive book support base with solid colors is used as the only constraint for setting the book or any bound volume in the right orientation and so that the background is of uniform. The top and lateral views of the support base are thus easily extracted from the images taken by the cameras placed in the overhead and lateral positions, respectively. Since both views represent the same dimensions of a same object in the real world, pixel correspondence between the two views is easily achieved.

Pipeline processing is used as an efficient approach for this type of high volume image processing. The key point was to asses what kind of pipeline algorithm was to be used to yield a high throughput between image acquisition and text processing. In this project, every acquired image consists of two pages of the bound volumes, and four operations are scheduled in pipeline in the following order: (1) barrel correction, (2) curve edge detection, (3) curvature correction and (4) image de-noising. For each operation, a sub-pipeline level can be employed to improve the performance further.

Since we use digital camera for image acquisition, the lens distortion is the first issue considered. There are four parameters used for barrel distortion correction which are intrinsic property for each camera. Therefore, the property of the lens of the cameras used in this dissertation had to be defined from the cameras' manufacturers.

The theoretical perspective of this research relates to the mathematical developments that are made in order to resolve both the inherent distortions due to the properties of the camera lens and the anticipated distortions of the changing page curvature as one leafs through the book from page to page.

For these reasons, the main tasks include:

Barrel Distortion Correction: A step undertaken at restoring the inherent lens effect which tends to produce a spherical like distortion; an effect more pronounced at extremities of the image.

Page Curvature Correction: This important step involves two interrelated issues, which are:

- **The X and Y axes transformation:** These are (a) projection transformation to eliminate the distortions in the Y direction, and (b) character extension in order to eliminate distortions in the X direction.
- **Interpolation:** Linear interpolation will be used here to eliminate the noise introduced through the geometric projection of text into a flat surface in order to improve the reading accuracy.

At this level, the automated readout could be initiated on the text part. The initial image segmentation is achieved through the detection of discontinuities using high pass filters. This is in order to delineate text from such things as pictures, figures, maps and other type of graphics. It should be noted that segmentation of a nontrivial image is one of the most difficult tasks in image processing.

1.5 Summary

The entire book reader system consisted of the a red painted wooden book holder with a piece of black cloth behind it serving as a uniform background, and two cameras taking pictures simultaneously from overhead and lateral sides, feeding the images to the computer through Universal Serial Bus (USB) connectors. The pre-processed results are then connected to the character recognition module. Image acquisition and processing were implemented on a DSP platform, even though the prototype model is still running using a MATLAB program on a PC platform.

The data analysis includes the acquired image data and the different error rates between the original image and the processed image all the way through the character recognition module. As part of the data analysis, document placement errors, different document curvatures, and varying font sizes, and different paper quality were considered in the implementation.

In Chapter 1, it is revealed that the technology for automated book readers is recent and newest developments have not come until 2003, despite the notable advances made in the digital processing field. The problem related to the technology of

automatically turning pages, combining mechanics and aerodynamic theories, has been regarded as a solved problem. Optical character recognition (OCR) for printed fonts and text to speech (TTS) are established technologies with impressive results. However, a key factor influencing the results of OCR and TTS are in the aforementioned inherent image distortions, involving barrel distortions and most importantly page curvature distortions. Very few commercial products that have been released to the market attempting to resolve these issues still face economic and technological barriers in terms of price, weight and the burden imposed by the curvature of bound volumes.

The leading companies in the automated book scanning field, such as BookDrive DIY from ATIZ Company and BookEye from ImageWare have created their unique methods in image acquisition. These type of new functionalities as well as the economic factors are compared with the proposed book reader system in Chapter 2.

All the newly developed theories and mathematical modeling of the proposed system are provided in Chapter 3. The lens distortion is the first one addressed. An inverse solution is used to overcome this distortion and improve consequently the efficiency of the program's performance. The next and most pertinent distortion that needed to be addressed is the distortion caused by the page curvature. The root cause of this distortion is the uneven distance between each point on the book surface and the overhead camera. To quantify these distances is the key for this corrective step. Another innovative idea that is introduced is to use another camera to take a lateral image at the same time as the overhead camera in order to capture the book curvature information. Difficulty of using this lateral image to recover the distance between the overhead camera

and to co-register this information with the overhead camera is resolved with some crucial preprocessing steps involving image alignment, curvature extraction and image data matching. Based on perspective transformation, practical algorithms to correct the distortions have been demonstrated comprehensively.

Chapter 4 introduces the methodology for integrating into the book reader design the OCR and the TTS platforms. In this process, it is evident that the objectives are to ensure a high reading accuracy by optimizing the recognition rate of characters. In this way, when the text is fed into the TTS process, the read out will be optimal.

Chapter 5 proposes a user-friendly GUI, in which all the practical functions are provided. With this GUI, the user can control the two cameras with real time view or image capture. The GUI not only provides every processing step described in chapter 3 for monitoring purposes and for comparing the results acquired at each step, but endows the user with fully automated image acquisition, processing and OCR-TTS with just one click. With the prototype setup and the GUI, an integrated system has been built and with the ultimate goal of this dissertation basically attained. This system had been tested by 4 blind persons from Lighthouse of Broward and its Executive Director and they unanimously acclaimed that this would be one of the easiest concept they could conceive for a book reader and that the results were very effective with realistic and practical processing time.

Chapter 6 provides the results for different cases and for all the intermediate steps of the process, involving different book curvature angles, different book paper materials

and different font families and font sizes. The recognition rate for the corrected image is dramatically increased to nearly 98% compared to less than 60% for the original image.

Chapter 7 summarizes the findings of this dissertation and proposes new avenues for research to extend further the prospects for universal accessibility to documentation. In retrospect, this research establishes a new platform for the automated reading of books, with their simultaneous digitization viewed as a significant corollary.

The significance of this work is further amplified considering that in the field of book digitization both Google and Microsoft are devoting great efforts and spending millions of dollars in recent years as part of the Open Content Alliance project to which this proposed project adheres fully and identifies to the noteworthy mission of creating virtual libraries with open access to all.

CHAPTER 2

SYSTEM DESIGN AND PRACTICAL IMPLICATIONS

2.1 Introduction

This chapter describes the general architecture of a novel book reader system, including its hardware setup and software architecture. The main qualities sought of this design are: (1) Inexpensive design affordable to all individual users, (2) Portability, (3) Processing time within practical bounds, with a waiting time for read outs that are acceptable by the user and are comparable to what it would take a sighted user to read a book; and (4) a high reading accuracy. It should be added that the potential for marketing the book reader as proposed is great, necessitating no special hardware requirements.

2.2 Situation of the Current Commercial Book Readers

A typical commercial book reader usually consists of 1) an advanced book scanning device, and 2) an optical character recognition (OCR) with text-to-speech (TTS) functionality.

In reference to Wikipedia, an advanced book scanner achieves the image acquisition through a high quality digital camera with a light source preferably on each side of it as shown in Figure 2-1. The book can be placed on the prepared table with a white base. Such a configuration allows the user to flip the pages of the book with ease and without each time pressing on the pages or on the hood of the scanner in order to

acquire the images in a satisfactory manner. Thus the concept of the advanced book scanner is to provide some form of protection to valuable books which are often vulnerable to tear. However we are left to contend with the page curvature under this configuration.

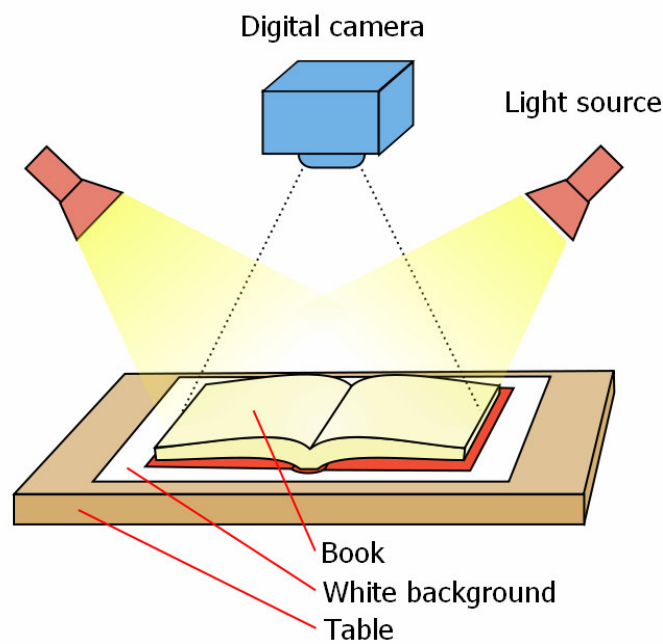


Figure 2-1 Sketch of a commercial book scanner from Wikipedia

Another issue with a commercial book reader is the requirements for OCR with TTS, in which the images obtained from the advanced book scanner are subsequently processed with an embedded layout analysis distinguishing text from other graphical representations. Character recognition and spell checking to transform the image into text and then feeding this text for a read out with a synthetic voice through TTS is the final goal. These two major steps are however linked through the image restoration tool,

consisting of several newly developed algorithms in order to correct for all potential distortions. This entire procedure is as illustrated Figure 2-2.

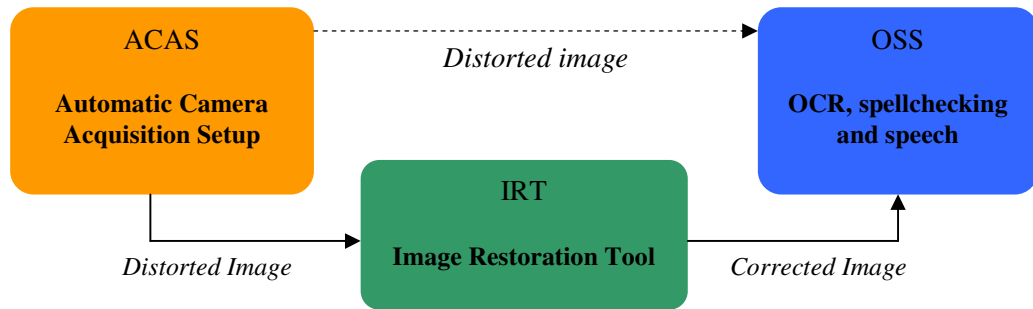


Figure 2-2 Book reader software architecture

The current OCR technology still can not handle the distortions introduced by the inherent design of the lens and the page curvature of bound volumes. In other words the IRT component of the design is essential.

2.3 Architecture of the book reader system

The integrated hardware-Software system architecture is as depicted in Figure 2.3. The novelty here is that the page curvature is addressed through the introduction of an additional lateral camera to the requirements of the advanced book scanner shown earlier in 2.1. This translates into a soft approach to resolving the page curvature issue. The two images are consequently entered into the image restoration module or tool to generate an undistorted image as means to significantly improve the performance rate of the OCR/TTS module.

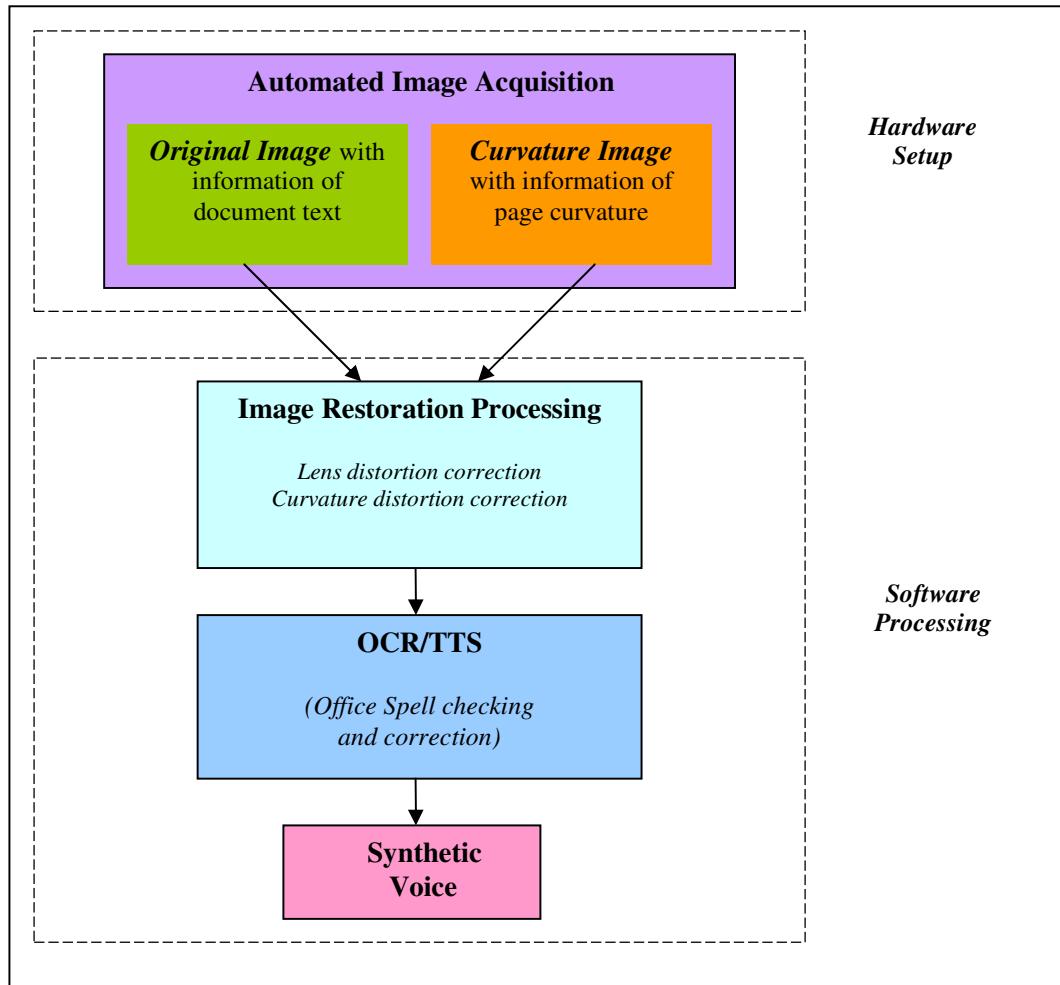
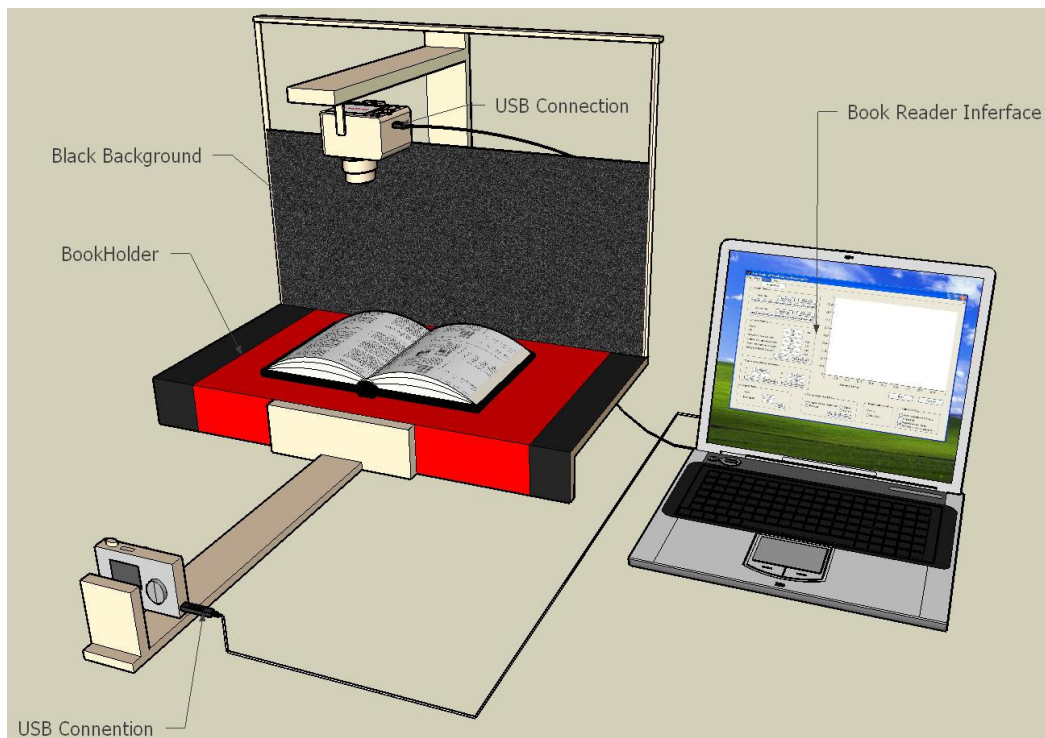


Figure 2-3 System architecture of a novel book reader

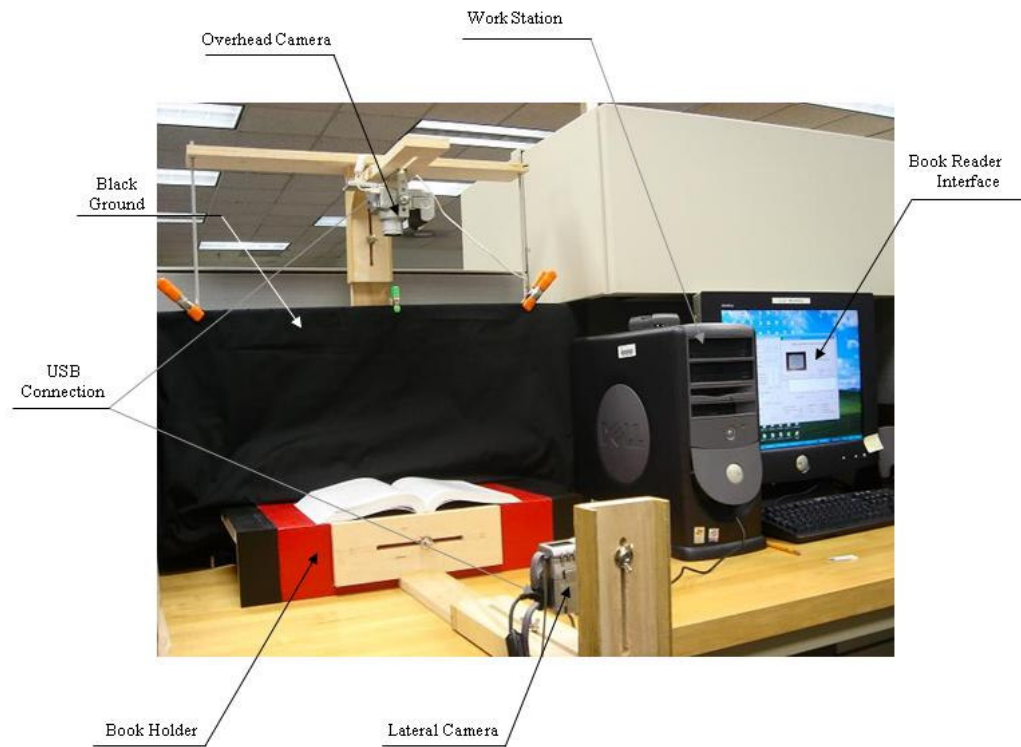
2.3.1 Hardware setup

In retrospect to the requirements established for the system shown in Figure 2.1, the proposed book reader hardware setup is as shown in Figure 2-4. The whole system consists of:

- A laptop, which runs the software to control the image acquisition and processing the software part as depicted in Figure 2-3.
- A book holder setup with a black background constraint.
- Two cameras that are mounted such that (a) one is to take an overhead picture of two pages simultaneously to generate a read out, and (b) a lateral camera such that the page curvature is addressed and its co-registration with the overhead image is attained. The black background constraint is useful in this case to facilitate the edge extraction of the curvature.
- The connectivity between the two cameras and the computer is accomplished through the USB ports.



(a) Graphical rendering



(b) Actual design

Figure 2-4 Book reader system hardware setup

It is noted that the size of the book holder is designed to fit most of the current books and the specified color on the book holder is designed to facilitate the task of segmentation for both images from overhead and lateral cameras. Adjustment of the positions of the both cameras provides the possibility for users to take most appropriate images. The size of the book holder could be made bigger if needed.

2.3.2 Software architecture

The image restoration processing module constitutes the entire software architecture as shown in Figure 2-5, with the following actions considered:

- Barrel distortion correction for both images. Barrel distortion, which is a type of image distortion caused by the camera lens as detailed in Chapter III, precedes all other distortion correction processes.
- Image segmentation for both images. The common part of the two images must be extracted through image segmentation algorithms, to facilitate co-registration.
- Curvature information extraction is performed on the lateral camera image.
- Once the curvature is extracted, height calculations to the overhead camera are performed so that the pixels on the curved pages are now placed in their appropriate position as if the pages were flattened.

The OCR/TTS module, which is in deep blue color in Figure 2-5, consists of:

- Document image analysis. This is to separate the image segments containing text from other image segments that contain any other graphical representation in the overhead document image.
- The image segments containing the text part is now fed to OCR and for spell checking. This OCR component is to transform the image part of the text into readable text. The image processing part is such as to maintain the image part of the text environment with the same layout as the original image.

- The TTS component is to transform the readable text to speech so that a synthetic voice can read out the text content to the user.

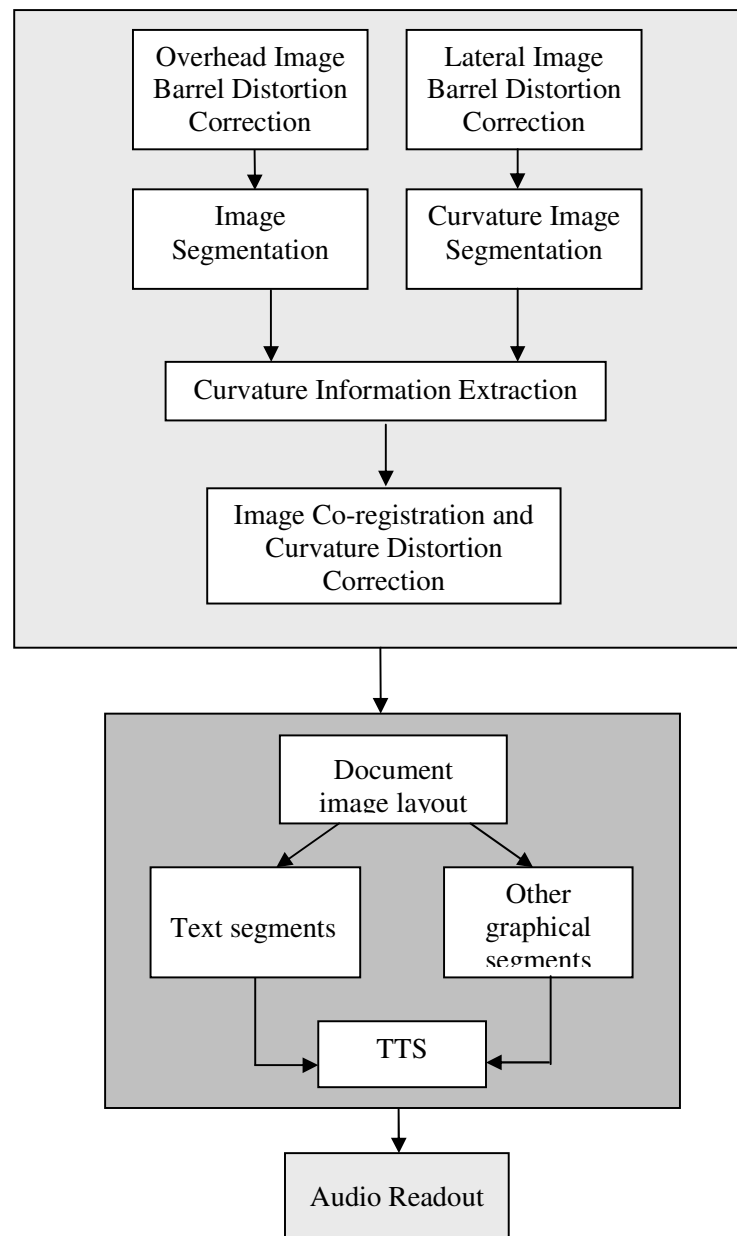


Figure 2-5 Software architecture of the book reader system

2.4 Marketing Potential

The 4DigitalBooks, which was earlier described in Chapter 1, is the world first fully automated book scanner. Its dimensions are 310x150x220cm, as shown in Figure 2-6(a). The methods for book flattening are partial or pressed. Its productivity is 3000 pages/hour. It is produced in Switzerland,.

There are two separate products related to the technology of automated book reader that are commercially available here in the United States. One is the advanced book scanner, of which the leading company is ATIZ, and the other is OCR software, of which is OmniPage remains the leading company.

ImageWare Bookeye GS400, as shown in Figure 2-6(b), utilizes two overhead lenses together to scan the book placed on the book holder. Its dimensions are 88 x 68.5 x 67.3 cm. Its results are mixed and are assessed in Chapter 5.

ATIZ BookDrive is the high end book scanner product with dimensions of 86 x 57 x 32cm, as shown in Figure 2-6(c). Its productivity is 500 pages/hour. Its heavy weight and high price are due in large part in the hard approach taken for curvature correction and for automating the page turning mechanism. Its results are also assessed in Chapter 5.

BookDrive DIY is a lower end book scanner using a V-shaped glass to flatten the book and applies two cameras on both sides to capture both pages of the book, as shown in Figure 2-6(d). Its dimension is 86x53x140cm, and its productivity is 700 pages/hour.



(a)



(b)



(c)



(d)

Figure 2-6 (a) 4DigitalBooks (b) BookEye book scanner (c) ATIZ BookDrive (high end) and (d) ATIZ BookDrive DIY (low end)

A comparison for the different features is listed in the Table 2-1 including the cost of these commercially available systems.

Table 2-1 Comparison of the potential book reader with leading commercial product in the market

Company	Features					
	Advanced book scanner	Automated book page turning	Page curvature distortion correction	OCR with document layout analysis	TTS	Approximate Price (\$)
ATIZ BookDrive	X	X				35,000
ImageWare BookEye	X		X			10,000
OmniPage				X		150-1500
ABBYY				X	X	200- 1500
Book reader interface	X		X	X	X	Cost 1500-2000

CHAPTER 3

THEORETICAL DEVELOPMENTS AND DESIGN

METHODOLOGY

3.1 Introduction

The central objective of the automated book reader is to overcome the need for scanning flattened pages of bound materials or any other type of documentation prior to having access to an automatic read out. Since cameras are used in the proposed design, and since the reading of bound documents such as books is what is at stake, this chapter describe the necessary steps it would take to read the pages of a book without flattening the pages. Therefore, the type of image distortions that are introduced by the camera lens and the distortion resulting from page curvature are systematically addressed. Also, the required image preprocessing steps needed are devised in order to establish appropriate mathematical modeling that will undo the effect of such distortions.

Consequently, Chapter 3 introduces the mathematical foundation and the design methodology of the fully integrated book reader design. This design integrates the following modules:

- Automated image acquisition,
- Barrel distortion correction,
- Image preprocessing for curvature extraction,
- Mathematical modeling of page curvature,

- Image segmentation for extracting text from other object in the book pages, and
- Text to audio read out.

3.2 Automated image acquisition

The automated image acquisition process is viewed as an important part of the book reader design in order to free the user from any unnecessary burden, and provide him/her with a user-friendly system and a simplified human-computer interface. The following options are deemed necessary in order to achieve this first automated step.

- Remote image preview control, which provides the user with a preview before the document image is captured.
- Remote image adaptation control, which provides the possibility for user to adjust the camera setting remotely, such as focal length, shutter speed and aperture in order to get a high quality image.
- Remote image capture control, which makes it possible to capture images remotely.
- Remote image transfer control, which transfers images from the storage media in the camera into the hard disk of the laptop running the software.

Specific Canon camera models like Canon G6 and PowerShot S60, which are used in the proposed book reader design, provide a software development kit (SDK) for users to implement any or all of the aforementioned four requirements.

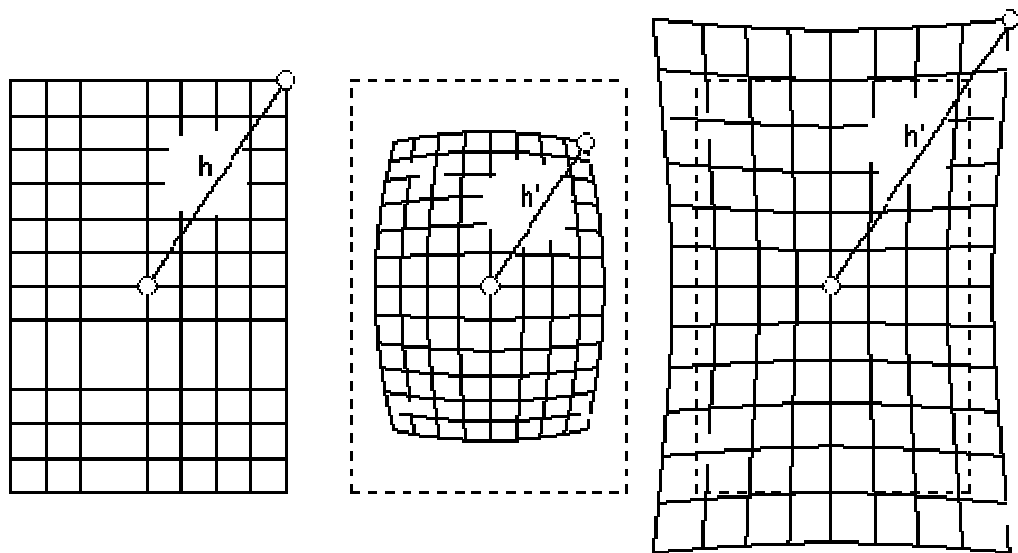
3.3 Image restoration

The image restoration process involves key intrinsic distortions introduced mainly by the camera lens and the page curvature. Given the design elements of the book reader, the overhead camera lens introduces the so-called barrel distortion which requires that a mathematical model be established and that the parameters of the overhead camera lens be determined. The second distortion, which is more pronounced, is due to page curvature, which in our view constitutes the most difficult part of the image restoration process. To resolve the page curvature problem, both image preprocessing to extract the shape of the curvature, and consequently, the mathematical modeling of the curvature need to be well accomplished so as to restore the distorted text and make it appear as if it were set in a flattened page.

3.3.1 Image distortion due to the camera lens

In a distortion-free image acquisition system, a straight line in the real-world environment should project as a straight line in the output or acquired image, yielding a rectilinear transformation or projection. However, due to the inherent design constraints of cameras, two types of magnification effects typically distort the output image. These are known as (1) the *barrel distortion effect*, in which magnification is inversely proportional to distance in reference to the optical axis (lesser magnification with increased distance from the optical axis), and (2) the *pincushion distortion*, in which magnification is now proportional to distance in reference to the optical axis (higher magnification with increased distance from the optical axis). These effects are

represented in Figure 3-1. Note that the barrel distortion, also referred to as a spherical distortion, resembles projecting a square-shaped grid as shown in Figure 3-1.a into a spherical-shaped object which would indeed look like the case shown in Figure 3-1.b. Since pincushion distortion is more of a side effect of a telephoto-type lens, which is not the case of the book reader design, this step of the process contends solely with the effects of barrel distortion. Since barrel distortion is a side effect of wide-angle-type lens, the top-view camera used in the book reader design, although not quite a wide-angle lens within the context of this experimental setup, does introduce such effects.



a. Square grid b. Barrel distortion effects on (a) c. Pincushion distortion effects on (a)

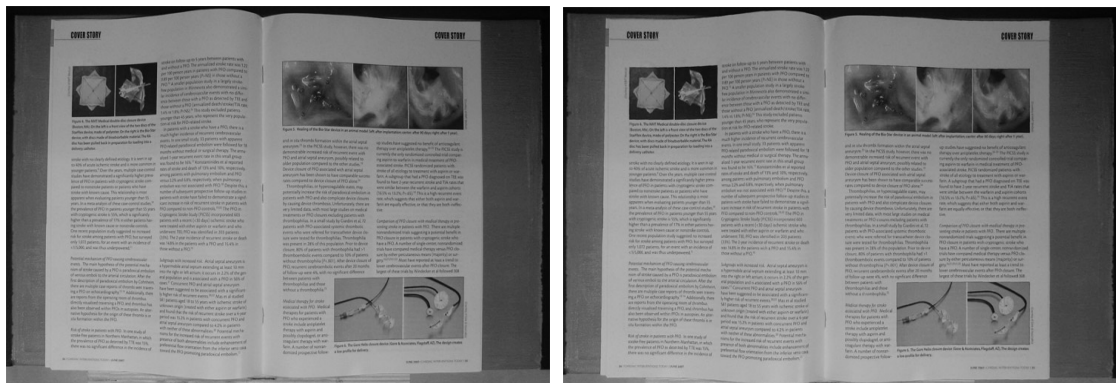
Figure 3-1: Image Distortions Introduced by Photographic Lenses

Illustrations of the barrel distortion are given as examples in Figure 3-2 for (a) a real world scene where the distortion is pronounced and for (b) a book page with text, where the distortion is more subtle. It is forewarned that within the context of this

dissertation, that although the barrel distortion is subtle for text input, it nonetheless introduces errors that affect the reading accuracy.



a. Barrel distortion effects on a real-world scene (courtesy of Wikipedia)



b. Barrel distortion effects on text

Figure 3-2 Barrel distorted image compared with original image

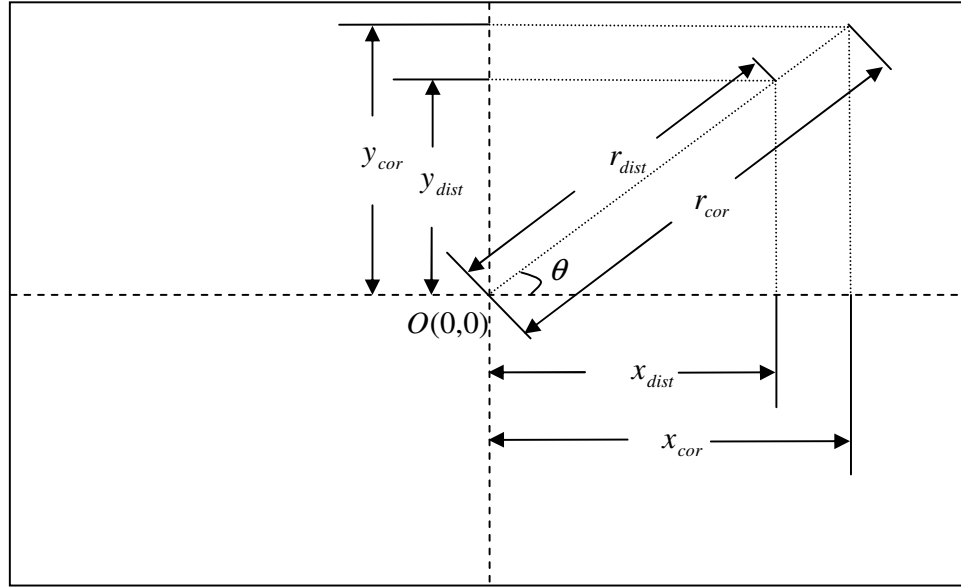
3.3.2 Mathematical Modeling of the Barrel Distortion

To correct for the barrel distortion, we are in effect trying to make of this spherical distortion a rectilinear projection. To do so, a geometric model together with the related mathematical formulae is established to address this phenomenon.

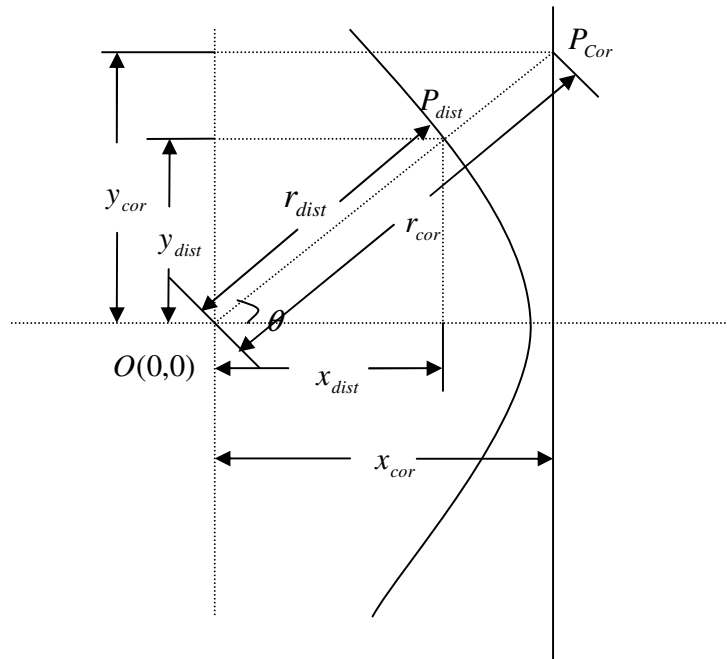
In these mathematical derivations, the image space is transformed from the Cartesian coordinate system to the Polar coordinate, where any point position in the image plane is now represented by its distance (radial coordinate r) and angle (angular coordinate θ) with respect to the point of origin $O(0,0)$. Taking the center point of the image as the point of origin, length r and angle θ of the Polar coordinate system are as defined in the common relation 3.1.

$$r = \sqrt{x^2 + y^2} \quad \text{and} \quad \theta = \arctan \frac{y}{x} \quad (3.1)$$

In reference to Figure 3-3, the mathematical expressions for resolving the effects of the barrel distortion can now be established.



(a) Nomenclature of the barrel distortion geometry



(b) Projection of a barrel-distorted line into a straight line in the corrected plane

Figure 3-3 Barrel Distorted Image and its correction

In this geometry, the relationship that exist between the distance of the origin $O(0,0)$ to the corrected image plane r_{cor} and the distance $O(0,0)$ to the distorted image plane r_{dist} , can be established using a fourth order vector multiplication as defined in (3.2)

$$r_{dist} = P^C \cdot R_{cor}^T \quad (3.2)$$

where, T indicates the transpose operation of R and C is to define the inherent parameters of the barrel distortion for a given camera that is being used in the experiment. These measurements are normalized with respect to the maximum of half the height and width of the image in both the corrected (destination) and distorted (source) image. Angles θ_{cor} , θ_{dist} defined as the angles in the corrected image and distorted image, respectively, with the fact that $\theta_{dist} / \theta_{cor} = 1$.

Defining vector $P^C = [P_1^c \ P_2^c \ P_3^c \ P_4^c]$, in which P_1^c , P_2^c , P_3^c , and P_4^c are parameters that are inherent to the properties of each camera lens that is being used. In this case $[P_1^c \ P_2^c \ P_3^c \ P_4^c] = [0.016 \ -0.042 \ 0 \ 1.026]$ with the upper script C defining the Canon G6 overhead camera; with $R_{cor} = [r_{cor}^4 \ r_{cor}^3 \ r_{cor}^2 \ r_{cor}^1]$ being a fourth order vector with respect to r_{cor} , the correction of the image distortion can be thought as being based on the fourth order polynomial given by (3.3), which prescribes an r_{dist} value for each r_{cor} point [Vass G. and Perlaki T., 2003].

$$r_{dist} = P_1^c \cdot r_{cor}^4 + P_2^c \cdot r_{cor}^3 + P_3^c \cdot r_{cor}^2 + P_4^c \cdot r_{cor}^1 \quad (3.3)$$

However, since the pixel tessellation is known at the destination image, given the discrete nature of the image and its known size, each r_{cor} position is known, it is only a matter of selecting the gray scale of each pixel value for the computed point r_{dist} and placing it in the known position r_{cor} which yielded r_{dist} . In other words, the gray scale of $Pixel(x_{cor}, y_{cor})$ becomes the gray scale of $Pixel(x_{dist}, y_{dist})$. It is noted that the (x, y) coordinates in the source image relate to the (x, y) coordinate system in the corrected image using the following relationships:

$$x_{dist} = x_{cor} \cdot r_{dist} / r_{cor} \quad y_{dist} = y_{cor} \cdot r_{dist} / r_{cor} \quad (3.4)$$

Above is the all the procedures to correct the image distortion caused by the camera lens. Actually it is not easy, at least taking too much time, to resolve equation 3.3 to get the r_{cor} through this fourth order polynomial. And choosing the correct one from the four resolutions will also take a lot of time. So I solve this problem reversely, which means find out the polar distance r_{dist} in the distorted image according to every polar distance r_{cor} in the corrected image.

Thus we have to solve equation 3.3 with $r_{dist} = \max (r_{dist})$, which in this case would be half of the diagonal distance of the image, and choose the solution r_{src} which is the closest to r_{src} to obtain the size of the corrected image in advance. Or we can use the size of the distorted image as the size of the corrected one if the information of in the margin of the image is not important enough.

3.3.3 Mathematical modeling of the page curvature distortion

Image capture of bound documents introduce distortions due to the page curvature, and this distortion is more pronounced as the curvature is higher, as in the case of book pages bound in thick volumes. Furthermore, the curvature of a given page on one side or the other of the book changes increases/decreases in an indeterminate manner as the pages are flipped from the beginning of the book to its end. Since the recognition rate of the current optical character recognition (OCR) systems, which is usually good on flat documents, is dramatically decreased while processing this kind of distorted document images. This particular problem is consequential and must be addressed before these book pages are fed as images to any OCR system.

As an illustration of this type of distortion, Figure 3-4 shows an example of a portion of document image affected by the page curvature.

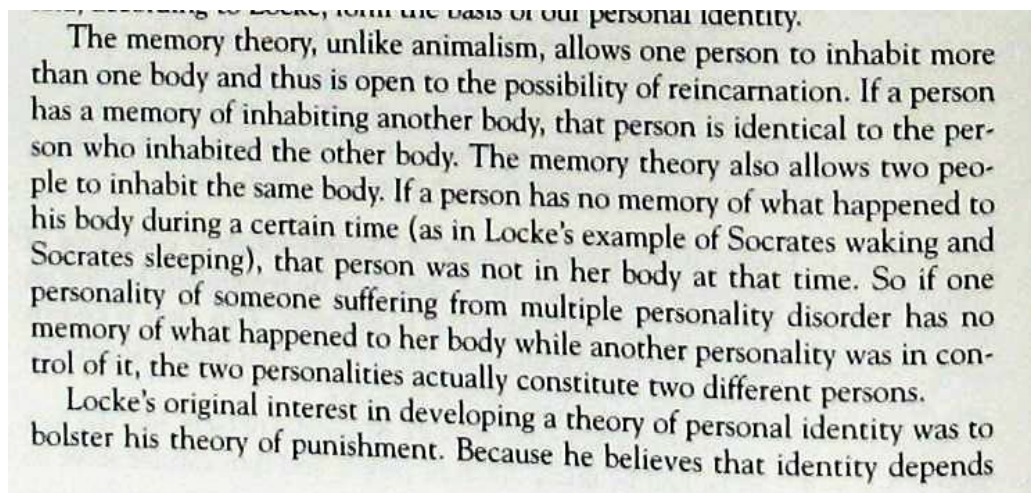


Figure 3-4 Sample of document image with page curvature distortion

Recently, a number of restoration techniques of document image distortion caused by page curvature have been proposed through "hard" approaches [Pilu M., 2001; Brown M. S. and Seales W. B., 2001] and "soft" approaches [Agam G. and Wu C. H., 2001; Liang J., et al., 2005], between which the difference is whether there is auxiliary hardware involved in dealing with the restoration process [Lu S. and Tan C. L., 2006; Pavic D. et al., 2006].

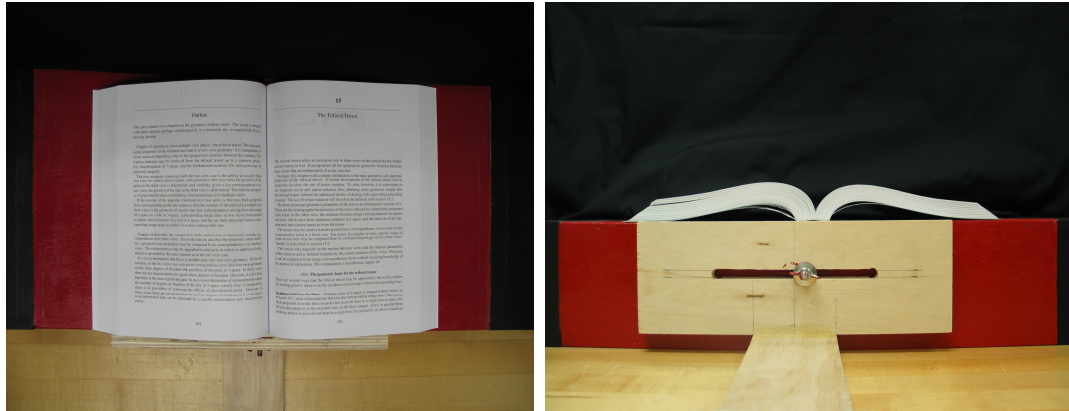
In [Pilu M., 2001; Pollard and Pilu, 2005], a special laser device is used to acquire 3D information and book curls as they are referred to in the articles are addressed through the projection of structured lighting involving 2-D light patterns and polygonal mesh approximations. The performance is very slow and rather difficult to converge, since this is a contentious dilemma between the sparsity of the 3-D maps with the applicability of the mesh surface. This type of work could be more useful for framing books in their different orientations rather than for reading text that is free of the curvature distortion [Pollard et al., 2000; Pilu, 2001 and 2002, Pollard et al., 2000, Isgro and Pilu, 2004; Clark and Mirmehdi, 2003].

In [Agam and Wu, 2001], a technique called 3D mesh manipulation, similar to that proposed by [Pollard and Pilu, 2005], is used to remove the curvature distortion instead of using any auxiliary device. This method is based on stereo disparity, which requires taking pictures from different viewpoints and cause image acquisition to be inefficiently performed, and stereo vision is known to be computationally taxing with also sparse results.

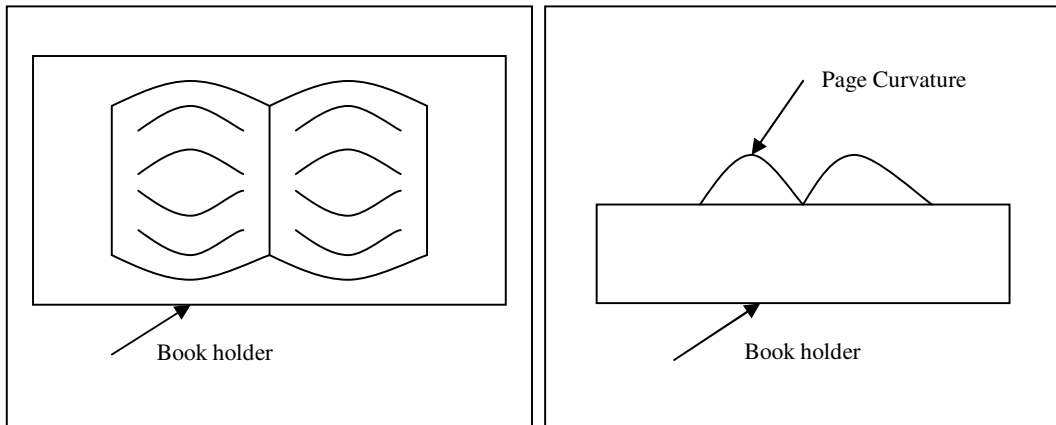
A novel and computationally-fast algorithm, classified as a combination of "hard/soft" approach, is presented in this section in order to eliminate the distortion caused by the book page curvature. In terms of auxiliary hardware, a single laterally-placed camera is used for the curvature detection. Recall that the second camera, the overhead camera is used to capture the content of the book pages which are fed to the OCR system. Images captured from both cameras are fed into the computer through USB links for further analysis. Figure 3-5 provides an illustration the technique to capture the both images from overhead and lateral side.

To determine the level of curvature at each pixel point in the lateral image, few preprocessing steps are considered prior to assessing the distortion resulting from page curvature, and these are:

- Image alignment and point matching to satisfy the uniqueness constraint
- Curvature extraction and height calculations of every pixel point in the lateral image to assess and correct the distortion effect of page curvature.



(a) Actual overhead camera image (left) and actual lateral camera image (right)



(b) Rendering of the overhead camera image (left) and lateral camera image (right)

Figure 3-5 Representation of the overhead and lateral images

3.3.2.1 Image alignment and matching of points in the lateral and overhead images

Since the overhead and lateral acquired images are taken of a same object (book or any bound documentation in this case), image segmentation is very important preprocessing first step to determine the matching points of what they share in common in order to satisfy the uniqueness constraint. This is not a stereo imaging problem, requiring matching stereo pairs and determining depth (in relation to page curvature in

this case). This option is not considered since the computational time is considered excessive for such an application, and yet the results are still parse and not accommodating for accurate audio read-outs of text [Adjouadi M. and Candocia F., 1994, Adjouadi et al., 1996]. But this would take a long time and not precisely enough. Another simple solution is to provide a reference, whose size is totally independent with the book size. Here I use the book holder as the reference for segmentation on both images. Figure 3-6 shows this procedure, where the red lines are the boundaries for the segmented images.

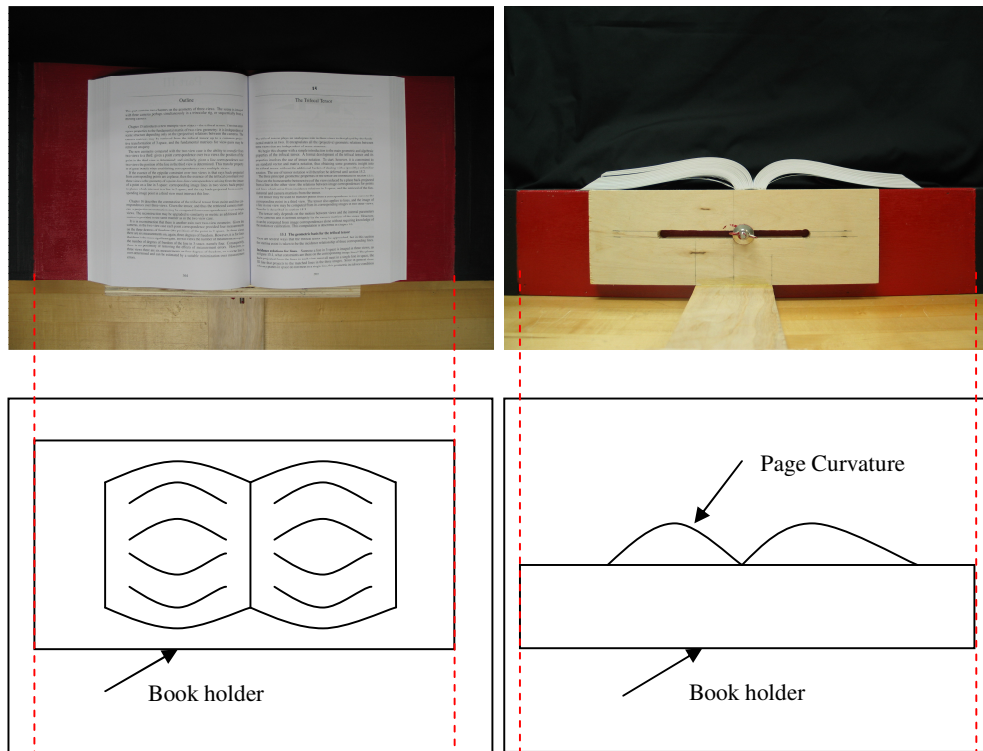


Figure 3-6 Image segmentation and extraction of the boundaries of the book holder in the overhead camera image (left) and the lateral camera image (right)

A simple convolution with the first-order derivative operator in the space domain, as given in equation 3.6, is performed to obtain the segmented image with the left and right boundaries of the aligned images:

$$I' = I * \frac{\partial}{\partial x} \quad (3.6)$$

Where $\frac{\partial}{\partial x}$ is the first-order derivative operator given by $[1 \ -1]$ or $[-1 \ 1]$ for left or right boundary detection, respectively. Since the book reader design is set up for a minimally controlled environment, necessitating simple requirements such as having a background that is easily distinguishable from the book holder, which in turn is of uniform color to distinguish it from the book it will hold.

For this type of edge detection, and in order to avoid calculating any threshold to minimize the total time it would take for a read out, an empirical threshold of 80 in a gray scale range of $[0 \text{ to } 255]$ is found to detect optimally the boundaries of the red book holder from the background of the original image, just as shown earlier in Figure 3.6. A prior estimate of the threshold in this case is simply the average gray scale of the book holder given that the black background yields an average gray scale of zero.

After this simple operation of extracting the book holder boundaries, coherence matching points from the overhead and lateral images is to be determined. This is to satisfy the uniqueness constraint in that each pair of points in the two images represents a same point in the real world. Taking P_o as a point with a distance x_o to the left boundary

of the segmented overhead image as illustrated Figure 3-7, the matching point P_l in the lateral image with distance x_l to the left boundary will have to satisfy the relationship given by equation (3.7).

$$\frac{x_o}{M} = \frac{x_l}{N} = \frac{X}{L} \Rightarrow x_o = \frac{M}{N} x_l \quad (3.7)$$

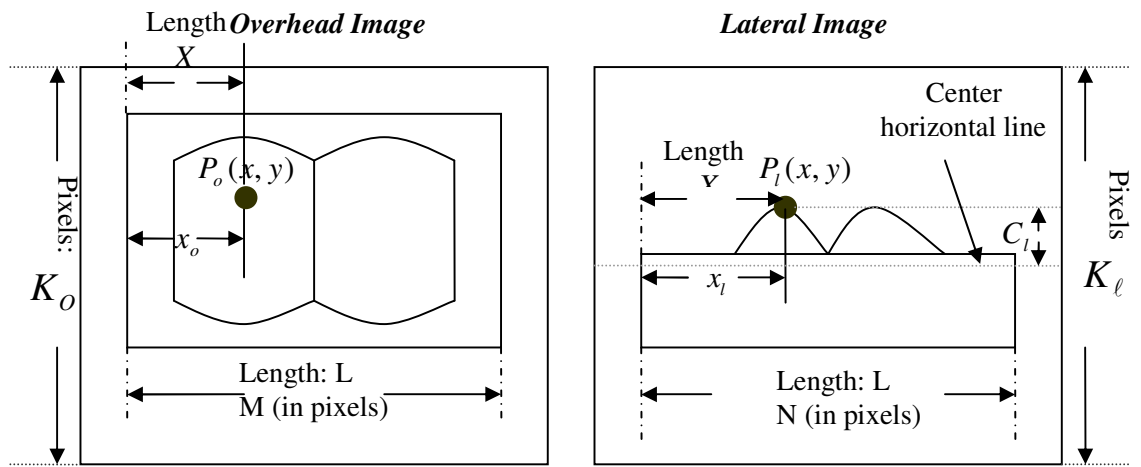


Figure 3-7 Matching points from the overhead and lateral images

Where M is the length in pixels of the overhead image, N is the length in pixels of the lateral image, with X being the real-world distance of that point to the left edge of the book holder, and L being the real-world length of the book holder.

3.3.2.2 Curvature extraction and height calculation

Extraction of the book curvature using the lateral camera is performed using the Laplacian of Gaussian filter (LoG) with an empirical value of $\sigma = 4$ chosen as the

standard deviation of the Gaussian function. This is due to the fact that the background color is predefined as stated in the preceding section [Woods R. E., 2002]. The curvature extraction operation is expressed mathematically in the spatial domain by equation 3-8.

$$g(x, y) = f(x, y) * LoG = f(x, y) * \left(-\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \right) \quad (3-8)$$

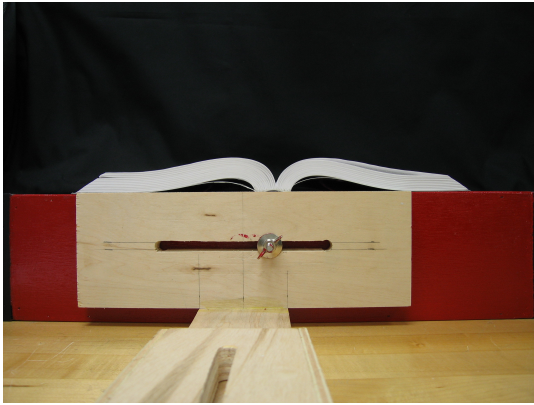
where x and y are the vertical and horizontal pixel position of any point with respect to a predefined center point of the image.

To avoid the convolution operation, Equation 3.8 is performed in the Fourier domain as a multiplication operation instead as shown in Equation 3.9, as any processing time saved is of the essence:

$$g(x, y) = F^{-1} \left\{ \sqrt{\frac{1}{2\pi\sigma^2}} \cdot e^{-2\pi^2\sigma^2(u^2+v^2)} \cdot (u^2 + v^2) \cdot F(u, v) \right\} \quad (3.9)$$

Threshold the zero-crossings of the processed image $g(x,y)$ and keep the stronger ones between the positive and negative value as white, which is location of the page curvature and set the rest as black, which is the background.

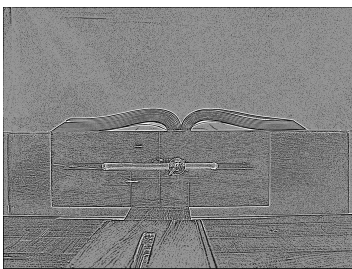
In Figure 3-8, the $g(x,y)$, the histogram of the $g(x,y)$ and zero-crossing thresholding results with different σ (2,4,6) are listed. When $\sigma = 2$, it is obvious in the histogram that we cannot thresholding the noise from object. While $\sigma = 6$, too much information of the object lost. So we choose $\sigma = 4$ to achieve the task of extracting the curvature very well. Typical results are as shown in Figure 3-8.



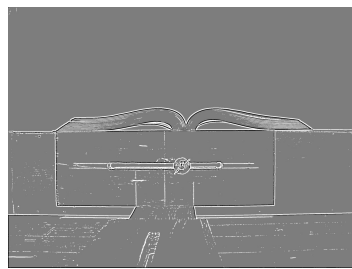
(a) Original Image



(b) Grayscale of (a)



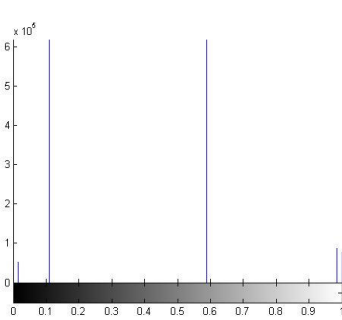
(c) LoG with $\sigma = 2$



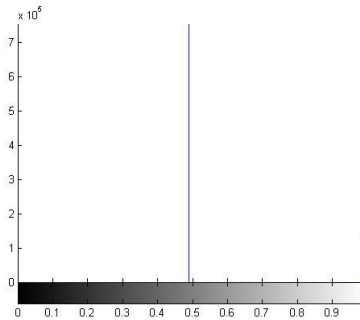
(d) LoG with $\sigma = 4$



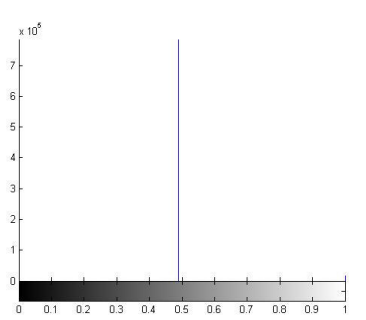
(e) LoG with $\sigma = 6$



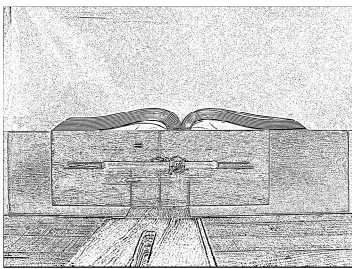
(f) Histogram of (c)



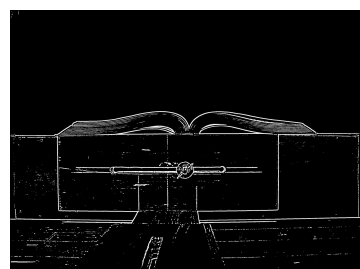
(g) Histogram of (d)



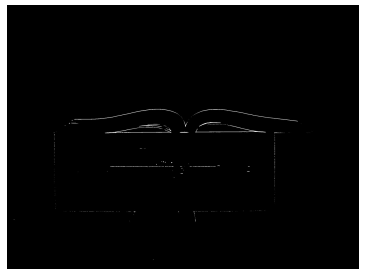
(h) Histogram of (e)



(i) Zero-crossing of (c)



(j) Zero-crossing of (d)

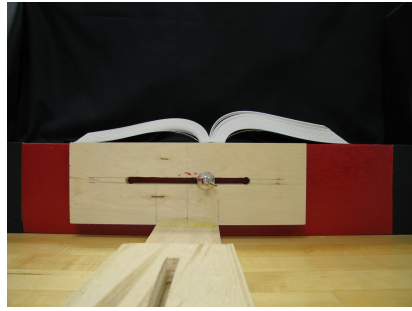


(k) Zero-crossing of (e)

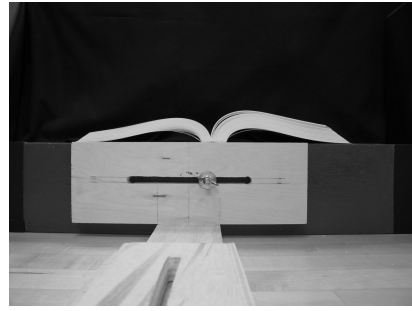
Figure 3-8: Results illustrating the curvature extraction process

Another approach that could be recommended as an alternative to the use of Laplacian of a Gaussian is one that is simpler in implementation. The idea is based on the knowledge that the book reader design is constrained in its set up and only the book to be read is what changes. Therefore, under normal conditions if you subtract the set up with a book placed on the book holder with a set up without any book, the difference will be akin to extracting the book from the image. This is reflected in Figure 3.9. In this figure, a picture of the set up is taken without the book and is used as a reference image. We subtract this reference image from the image that did contain a book, and the result is as shown in Figure 3-9 (e) and (f) with the curvature extracted accurately as shown in Figure 3-9 (e). It should be noted that such an approach although not necessitating a thresholding mechanism to extract the curvature, should be used only if the set up is guaranteed not to experience any change.

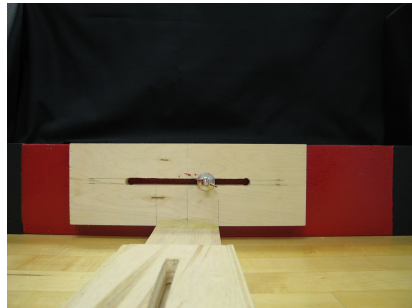
Once the curvature is extracted in the processed image $g(x, y)$ of equation (3.9), with N denoting the length in pixels of the segmented lateral image, the page curvature of the lateral image ℓ can be expressed as 1-D vector \mathbf{C}_ℓ whose elements are the pixel number of $[(g_k(x, y))]$ on the curvature to the center horizontal line of the image, with $k = [0, 1, \dots, N - 1]$. Having established \mathbf{C}_ℓ , it is critical to accurately project the curvature information into the overhead image so that the page readout is precise and free of the page curvature distortion. Fortunately, the mapping of \mathbf{C}_ℓ to its counterpart vector \mathbf{C}_o , with length of M , in the overhead image is already resolved through the use of equation 3.7 provided earlier.



(a) Original lateral image



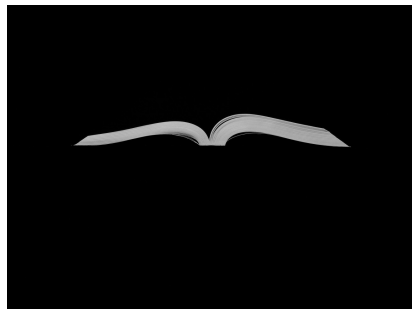
(b) Grayscale of (a)



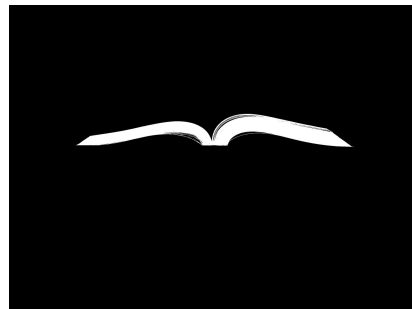
(c) Reference image



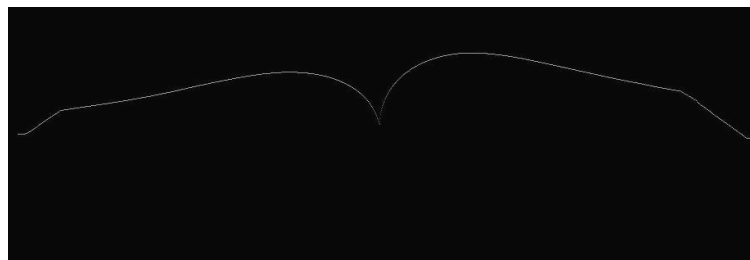
(d) Grayscale of (c)



(e) Subtract (d) from (b)



(f) Binary BW image of (e)



(g) Top edge of (f)

Figure 3-9 Another approach to obtain the edge of the book curvature

Since the objective is to flatten the page, only coordinate x is of interest, and the relationship between the two vectors will rely on the aforementioned expression

$$x_o = \frac{M}{N} x_\ell \text{ relating the overhead image (} o \text{) to the lateral image (} \ell \text{).}$$

Using Figure 3.10 as a reference, mathematical details of this problem can now be addressed. In this figure \mathbf{H}_c represents the vertical distance between each point on the page curvature and the overhead camera as a function of the image point (x_i, y_i) , as expressed in Equation 3.10.

$$\mathbf{H}_c = \mathbf{H}_o - (\mathbf{H}_{c\ell} + \mathbf{H}_\ell) \quad (3.10)$$

Furthermore, \mathbf{H}_ℓ is also expressed as a constant vector where each element has for value the distance between the center of the lateral camera lens and the top of the book holder. Similarly, \mathbf{H}_o is expressed as a constant vector where each element has for value the distance between the center of the overhead camera lens and the bottom of the book holder.

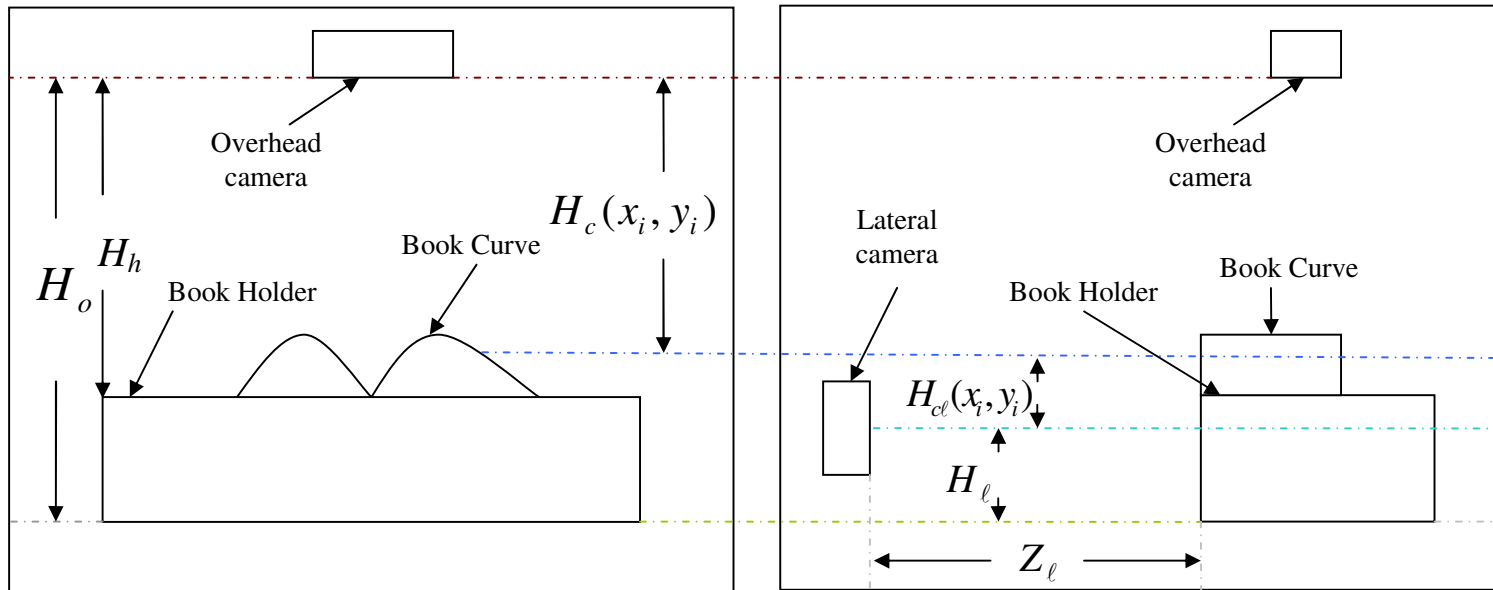
The distance $\mathbf{H}_{c\ell}$ between any given point on the curvature and the overhead camera can now be computed as:

$$\mathbf{H}_{c\ell} = \frac{\mathbf{C}_o \cdot H_s}{K_\ell \cdot f} Z_l \quad (3.11)$$

where H_s is the height of the imaging sensor within the lateral camera, in this case $H_s = 5.32mm$ for the Canon PowerShot S60, K_l is the height of the lateral image as shown in Figure 3-7 earlier, Z_l is the distance between the lateral camera and the lateral side of the book page, and f is the focal length of the lateral camera, in this case $f = 5.8mm$.

With these mathematical derivations, all the matching points from the overhead and lateral images are determined, and the distance of each point in the overhead image to the overhead camera can be computed. Two additional steps are now considered to complete the process of page curvature correction. These two steps are:

- Push operation in order to flatten the page so as to overcome what we term as the distortion in the vertical direction and
- Extend operation to uncompress pixels so as to reconcile the M and N horizontal dimensions of the lateral and overhead cameras, respectively. This step is performed in order to overcome what we term as the distortion in the horizontal direction, an artifact on the pixels due to the curvature after the push operation.



Front view (left) and the lateral view (right) of the whole system

Figure 3-10 Front view and lateral view of the whole system with size measure

3.3.2.3 Push operation

It is evident that correcting for the page curvature distortion becomes an easier process when the height of each point on the overhead image from the overhead camera is determined. The push operation is akin to a rectilinear transformation of the points on the curve so that the text perceived by the overhead camera appears in a straight line. In reference to Figure 3-11, this process is viewed as an orthogonal projection of the points belonging to the curve onto a flat surface which is parallel to the image plane of the overhead camera.

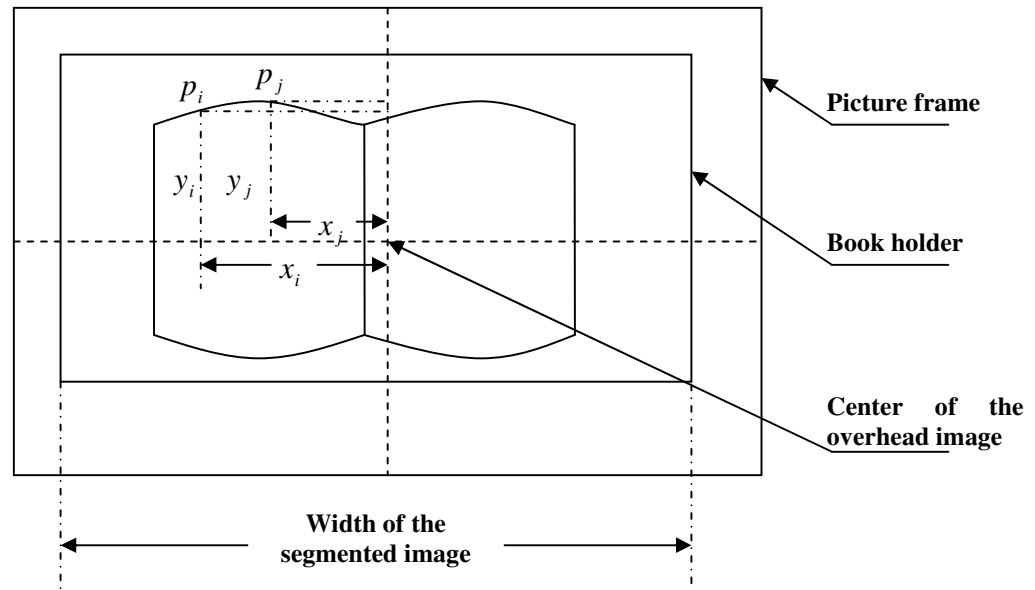


Figure 3-11 Push operation for the overhead image

$H_c(x_j, y_j)$ is a single-valued measure while \mathbf{H}_c is a vector whose elements are the different $H_c(x_j, y_j)$ values are evaluated with x_j varying between 1,..., M-1 and with the assumption that \mathbf{H}_c is independent of the coordinate y .

Figure 3-11 considers any two pixels $p_i(x, y)$ situated at (x_i, y_i) , and $p_j(x, y)$ situated at (x_j, y_j) both with respect to the center of the overhead image, which are assumed to belong to a same straight line in the real world, but with the distance of Δy between them in the overhead image. Let $\mathbf{H}_c[p_i] = \mathbf{H}_c(x_i, y_i)$ and $\mathbf{H}_c[p_j] = \mathbf{H}_c(x_j, y_j)$ denote the distances between the pixels p_i and p_j and the overhead camera, respectively in the real world as a function of the image point (x_i, y_i) . This measure \mathbf{H}_c is as given earlier in equation 3.10.

Using the standard perspective transformation, it is established that:

$$y_i = \frac{f.Y}{H_c(x_i, y_i) - f} \quad \text{and} \quad y_j = \frac{f.Y}{H_c(x_j, y_j) - f} \quad (3-12)$$

A modification weight vector \mathbf{W} with length M as given in equation 3.13 is established in order to satisfy the rectilinear transformation.

$$\mathbf{W} = \frac{\mathbf{H}_c - \mathbf{f}}{H_h - f} \leq 1 \quad (3-13)$$

where H_h is a constant reference, taken as the distance between the overhead camera and the book holder and $\mathbf{f} = [f_0, f_1, \dots, f_{M-1}]$ being a constant vector with length M and whose elements $f_k, k = 1, 2, \dots, M-1$ are all equal to the camera's focal length f .

Looking back at the two pixels situated at (x_i, y_i) and (x_j, y_j) in Figure 3.11, it is understood that the page curvature distortion correction is based on the perspective

transformation given in equation 3.12. In the rectilinear transformation, the initial goal is thus to ensure an equal y coordinate value for all the points belonging to the curve. Therefore, given that $\mathbf{W}=[W_0, W_1, \dots, W_{M-1}]$ and $\mathbf{H}_c=[H_0, H_1, \dots, H_{M-1}]$, the effect of the modification weight vector \mathbf{W} on equation 3.12 yields the following results:

$$\begin{aligned} y'_i &= y_i \cdot W_i = \frac{Y}{H_i - f} \cdot f \cdot \frac{H_i - f}{H_h - f} = \frac{Y}{H_h - f} \cdot f \\ y'_j &= y_j \cdot W_j = \frac{Y}{H_j - f} \cdot f \cdot \frac{H_j - f}{H_h - f} = \frac{Y}{H_h - f} \cdot f \end{aligned} \quad (3.14)$$

Given that $y'_i = y'_j$, this clearly establishes that the y coordinate for the two points is now the same in the corrected image, and this is true for all other points belonging to the curve.

Note if a given point on the curve intersects the book holder (i.e., $H_i=H_h$), then that point will not be modified by this weight vector, as expressed in equation 3.15:

$$y'_i = y_i \cdot W_i = \frac{Y}{H_h - f} \cdot f \cdot \frac{H_h - f}{H_h - f} = \frac{Y}{H_h - f} \cdot f \quad (3.15)$$

Thus the distortion on the vertical direction has been corrected and the two points, which are on the same line in the real world, will be on the same line in the image as well.

This transformation is viewed at this time as an image index transformation, in which all the calculations are based on the positional index of every pixel. Pixel

assignment in gray level between the new position and the original position of the pixel will be done ultimately just by assigning the appropriate gray scale to the new position.

For this image index transformation, Let K_o be the height of the overhead image as indicated earlier in Figure 3.7, the vertical index of the distorted image \mathbf{Q}_{dist} is set up as given in equation 3.15. In this equation, it should be noted that for an even K_o , the incremental measure is always 1 but the starting point is 0.5 (an in-between pixels compromise) since the zero line would otherwise cross the starting pixel.

$$\mathbf{Q}_{\text{dist}} = \underbrace{\begin{bmatrix} 0 & 0 & & 0 & 0 \\ 1 & 1 & & 1 & 1 \\ 2 & 2 & & 2 & 2 \\ \dots & \dots & \dots & \dots & \dots \\ K_o - 2 & K_o - 2 & & K_o - 2 & K_o - 2 \\ K_o - 1 & K_o - 1 & & K_o - 1 & K_o - 1 \end{bmatrix}}_M \quad (3.15)$$

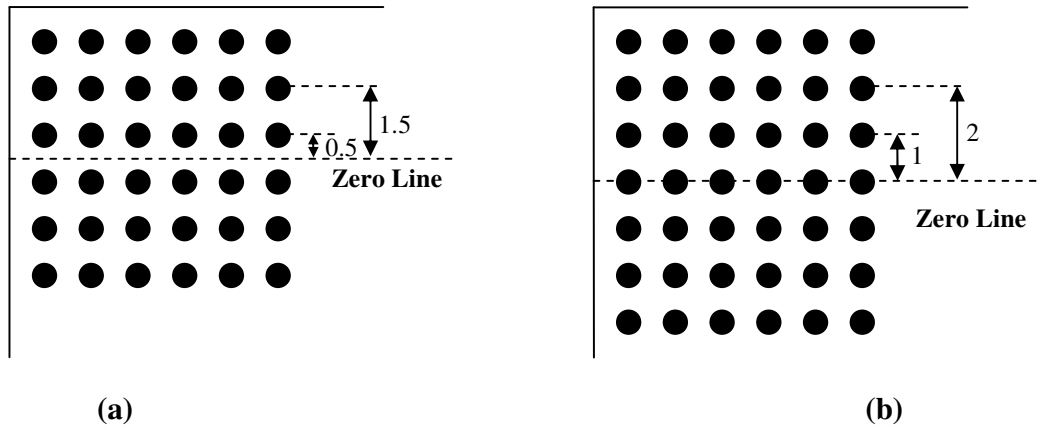


Figure 3-12 Illustration of pixel distance for images with (a) even and (b) odd height

Another vector named pixel distance vector \mathbf{D}_{dist} is established as equation 3.16:

$$\mathbf{D}_{\text{dist}} = \mathbf{V}_{\text{CIm}} - \mathbf{Q}_{\text{CIm_dist}} \quad (3.16)$$

where Matrix \mathbf{V} is of dimension $[K_o \times M]$ shown as below:

$$\mathbf{V} = \underbrace{\begin{bmatrix} \frac{K_o-1}{2} & \frac{K_o-1}{2} & \dots & \dots & \frac{K_o-1}{2} \\ K_o-1 & \ddots & & & K_o-1 \\ \frac{2}{2} & & \ddots & & \frac{2}{2} \\ \vdots & \vdots & & \ddots & \vdots \\ \vdots & \vdots & & & \vdots \\ \frac{K_o-1}{2} & \frac{K_o-1}{2} & \dots & \dots & \frac{K_o-1}{2} \end{bmatrix}}_M \quad \left. \vphantom{\begin{bmatrix} \frac{K_o-1}{2} & \frac{K_o-1}{2} & \dots & \dots & \frac{K_o-1}{2} \\ K_o-1 & \ddots & & & K_o-1 \\ \frac{2}{2} & & \ddots & & \frac{2}{2} \\ \vdots & \vdots & & \ddots & \vdots \\ \vdots & \vdots & & & \vdots \\ \frac{K_o-1}{2} & \frac{K_o-1}{2} & \dots & \dots & \frac{K_o-1}{2} \end{bmatrix}} \right\} K_o$$

where $\mathbf{V}_{\text{CIm}} = [\frac{K_o-1}{2}, \frac{K_o-1}{2}, \dots, \frac{K_o-1}{2}]^T$ is any column of \mathbf{V} and $\mathbf{Q}_{\text{CIm_dist}}$ is any column of the vertical matrix \mathbf{Q}_{dist} .

The vertical index matrix of the corrected image \mathbf{Q}_{cor} of dimension $[K_o \times M]$ is calculated via Equation 3.17.

$$\mathbf{Q}_{\text{cor}} = \mathbf{V} - \mathbf{D}_{\text{dist}} \cdot \mathbf{W} = \mathbf{V} - \frac{1}{H_h - f} \begin{bmatrix} \frac{K_o-1}{2} \\ \frac{K_o-3}{2} \\ \dots \\ -\frac{K_o-3}{2} \\ -\frac{K_o-1}{2} \end{bmatrix} [H_0 - f \ H_1 - f \ \dots \ H_{M-1} - f] \quad (3.17)$$

Since $\mathbf{W} \leq 1$, the vertical index of the corrected image $\mathbf{Q}_{\text{cor}} \geq 0$. Practically we still need to get the nearest integer for every element of \mathbf{Q}_{cor} because it is the vertical index of an image.

3.3.2.4 Extend operation

With the push operation accomplished, the distortion on the vertical direction is considered as having been eliminated. However, on the horizontal direction, the pixels seem to be compressed through the projection of the push operation as an effect of the rectilinear transformation of the curvature. Subsequently, as a next step, an extension operation on the horizontal direction is required to eliminate the page curvature distortion totally.

Let us consider pixel points p_i, p_j as any two points in the same line in the real world, as in Figure9. After push operation, we have:

$$x'_j - x'_i = \frac{X_j - X_i}{H_h - f} \cdot f \quad (3-15)$$

As in Figure 3-13, which shows the view of the section plane of the page curvature, assuming the two points p_i, p_j are very close to each other, and the horizontal distance between each other δ is very small, saying $\Delta x = 5 \sim 10$, then we can take the arch between the two points approximately as a straight line.

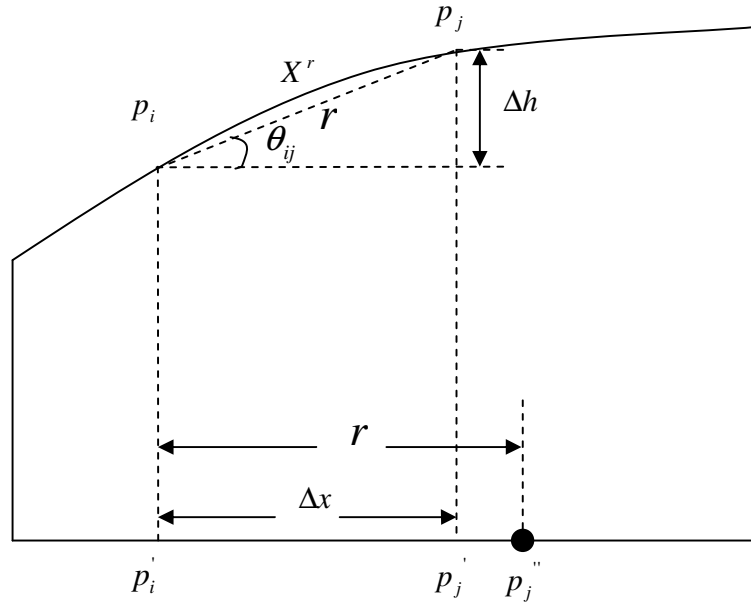


Figure 3-13 Section view of the page curvature

Let
$$\theta_{ij} = \text{angle}(p_i, p_j) = \arctan \frac{\Delta h}{\Delta x}$$

We take $\cos(\theta_{ij})$ as a factor to adjust the distortion in the X direction:

$$x_j'' - x_i'' = (x_j' - x_i') / \cos(\theta_{ij}) = \frac{(X_j - X_i) / \cos(\theta_{ij})}{H_h - f} \cdot f \approx \frac{X^r}{H_h - f} \cdot f$$

where X^r is the arch length between the two points p_i, p_j in the real world, which is approximately equal to r when Δx is very small. After this extension operation, the page curvature distortion on the horizontal direction is in turn eliminated.

CHAPTER 4

INTEGRATION OF OCR AND TTS PLATFORMS

4.1 Introduction

This chapter provides a description on how existing modules for optical character recognition (OCR) and for text to speech (TTS) modules have been integrated into the automated book reader design.

The OCR and TTS modules chosen are those confirmed to be the most reliable and yet are available under the Microsoft operating system free of charge.

4.2 Image classification and recognition

4.2.1 Image classification

Research of automated document image segmentation and classification has started in the early 1960's. The purpose of it is to distinguish the text from image part within a document image to facilitate the text recognition in OCR software and convert the image into editable document text file with the same layout.

The methods used for document image segmentation and classification, which are proposed in the last 20 years, may be classified as top-down, bottom up or as hybrid methods. Run length smoothing algorithm (RLSA), using horizontal and vertical parameters, is a typical top-down method in image segmentation [Wong K. Y. et al.,

1982]. For bottom-up methods, connected components are grouped and merged into successively larger components to discriminate other parts in the image [Strouthopoulos C. and Papamarkos N., 1998]. Hybrid approaches involve local and global strategies in combination with texture analysis techniques [Umer M. F. and Khiyal M. S. H., 2007].

RLSA has been pervasively used to detect long vertical and horizontal white lines in the image and it is also extended to identify different areas containing the various types of data. This algorithm is described in detail in the "Document Analysis System" by IBM as early as 1982.

The architecture of the toolkit is shown in Figure 4-1. A prototype of document image classification toolkit based on MATLAB is detailed in the next chapter.

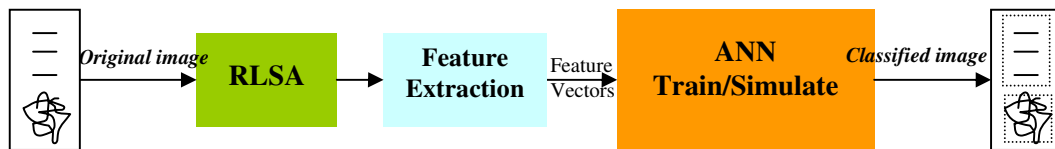


Figure 4-1 Architecture of document image classification

There are 4 features in each block, and they are (1) the ratio of block width to the image width, (2) the ratio of block height to the image height, (3) the ratio of the number of black pixels to the total number of pixels in the block, and (4) the ratio of run length to the number of pixels in the block. These features are extracted to be taken as inputs to

train the neural network. With the large amount of document image examples, the neural network is robust enough to classify any document image.

4.2.2 Image recognition

After image preprocessing, the text and other graphical parts of the document image have been separated. The image part can be directly copied to any text processing software, such as Microsoft Word, with its layout information, and obtain the digital text by inputting the text part image into the image recognition engine, such as Microsoft Office Document Imaging 2003 (MODI) object model for recognition and Microsoft Word Object Library for spell checking and correction, if necessary.

To use MODI object model automation, MODI 11.0 Type Library is referenced in our project. The MODI object model consists of the following objects, which are to be instantiated in .NET object oriented programming language:

- **Document** object represents an ordered collection of pages (images).
- **Image** object represents a single page of a document.
- **Layout** object exposes the results of optical character recognition on a page.
- **MiDocSearch** object exposes document search functionality.

Every picture obtained from the overhead camera is regarded as an **Image** object and can be organized into one Document. The OCR method can be performed on a single page or on the entire document according to the user's needs.

4.3 Text-to-Speech synthesis

A Text-To-Speech (TTS) synthesizer is a computer-based system that should be able to read any text aloud, whether it was directly introduced in the computer by an operator or scanned and submitted through an optical character recognition system.

There is a fundamental difference between the system we are about to discuss here and any other talking machine (as a cassette-player for example) in the sense that we are interested in the automatic production of new sentences. This definition still needs some refinements. Systems that simply concatenate isolated words or parts of sentences are identified as Voice Response Systems, and are only applicable when a limited vocabulary is required (typically a few hundreds of words), and when the sentences to be pronounced respect a very restricted structure, as is the case for the announcement of arrivals in train stations for instance.

In the context of TTS synthesis, it is impossible (and luckily useless) to record and store all the words of the language. It is thus more suitable to define Text-To-Speech as the automatic production of speech, through the so-called “grapheme-to-phoneme transcription” of the sentences to be vocalized.

4.3.1 TTS Architecture

Currently, the functional diagram of a very general TTS synthesizer is as illustrated in Figure 4-2.

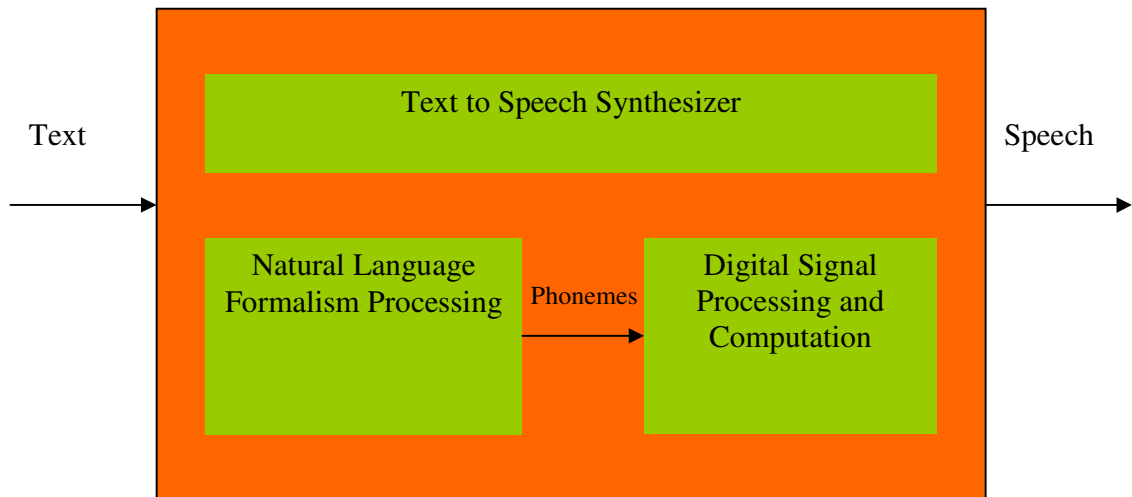


Figure 4-2 Text-To-Speech Architecture

This architecture comprises a Natural Language Processing module (NLP), capable of producing a phonetic transcription of the text that is read, together with the desired intonation and rhythm, which is often termed as prosody. A digital signal processing module transforms the symbolic information it receives into speech.

The formalisms and algorithms applied often manage to short-circuit certain processing steps, thanks to a judicious use of mathematical and linguistic knowledge of the developers. This is occasionally achieved at the expense of some restrictions on the text to pronounce, or results in some reduction of the "emotional dynamics" of the

synthetic voice, compared with human performances. However, it generally allows solving the problem in real time with limited memory requirements [Dutoit T., 2001].

4.3.2 TTS development skill

4.3.2.1 Apple

The first speech system integrated into an operating system was Apple Computer's MacInTalk in 1984. Since the 1980s Macintosh Computers offered text to speech capabilities through The MacInTalk software. In the early 1990s apple expanded its capabilities offering system wide text-to-speech support. With the introduction of faster PowerPC based computers which included higher quality voice sampling. Apple also introduced speech recognition into its systems which provided a fluid command set.

More recently, Apple has added sample-based voices. Starting as a curiosity, the speech system of Apple Macintosh has evolved into a cutting edge fully-supported program, called PlainTalk, which was destined for people with vision problems. VoiceOver, which is now included in Mac OS 10.4 Tiger is included with all installations of Mac OS Tiger and Mac OS Leopard.

4.3.2.2 Microsoft Windows

Modern Windows systems use SAPI4- and SAPI5-based speech systems that include a speech recognition engine (SRE). SAPI 4.0 was available on Microsoft-based operating systems as a third-party add-on for systems like Windows 95 and Windows 98. Windows 2000 and Windows XP finally added a speech synthesis program called

Narrator, directly available to users. All Windows-compatible programs could make use of speech synthesis features, available through menus once installed on the system. Microsoft Speech Server is a complete package for voice synthesis and recognition, for commercial applications such as call centers. Figure 4.3 shows the structure of the Microsoft TTS software development kit 5.3.

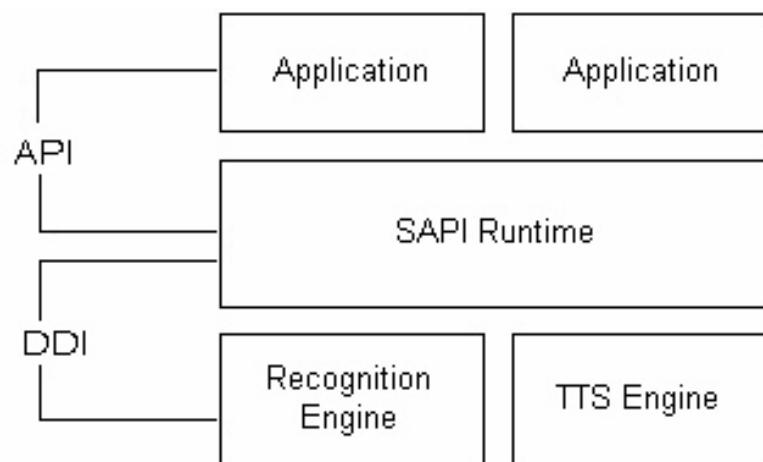


Figure 4-3 Microsoft TTS Software Development Kit 5.3

This Windows TTS architecture which embeds the Application Programming Interface (API) and Device Driver Interface (DDI), is the most current Microsoft TTS software development kit, which provides automation features to SAPI 4.0 for developers to implement speech applications using C# or any other automation languages.

Since SAPI is a Component Object Model (COM), an interop DLL is required to be created to use the SAPI from our project. Using the ISpVoice object, the configuration methods such as "SetRate", "SetVoice" and action method "Speak" can be called to implement a synthetic voice speech.

CHAPTER 5

INTRODUCING A MATLAB INTERFACE FOR AUTOMATED BOOK READER SYSTEM

5.1 Introduction

This chapter presents a user-friendly interface aimed at providing remote image acquisition through common digital cameras, image restoration, optical character recognition (OCR) and text-to-speech all integrated in a single graphical user interface (GUI). The user can also initiate all these steps at once in one single click of a button.

This is a MATLAB based interface provides adjustment panels for all the necessary parameters, controls the overhead and lateral image displays. The text to speech function is invoked after the OCR process has been finished successfully. This interface also provides the possibility for the user to process images which have been taken before. This interface is also established such as all image processing algorithms including the lens distortion correction and edge detection operation are monitored in real-time as these processes are carried on.

5.2 GUI overview

The Graphic User Interface (GUI) developed for the automated book reader is as shown in Figure 5-1. This interface provides the user with the ability to select overhead and lateral images from "Browse" button in the "Input Files" panel. The file path will be

displayed in the text box after the user chooses any image. "Image browsing" panel will display the image the user chooses just by clicking the "Preview" button in the "Input Files" panel.

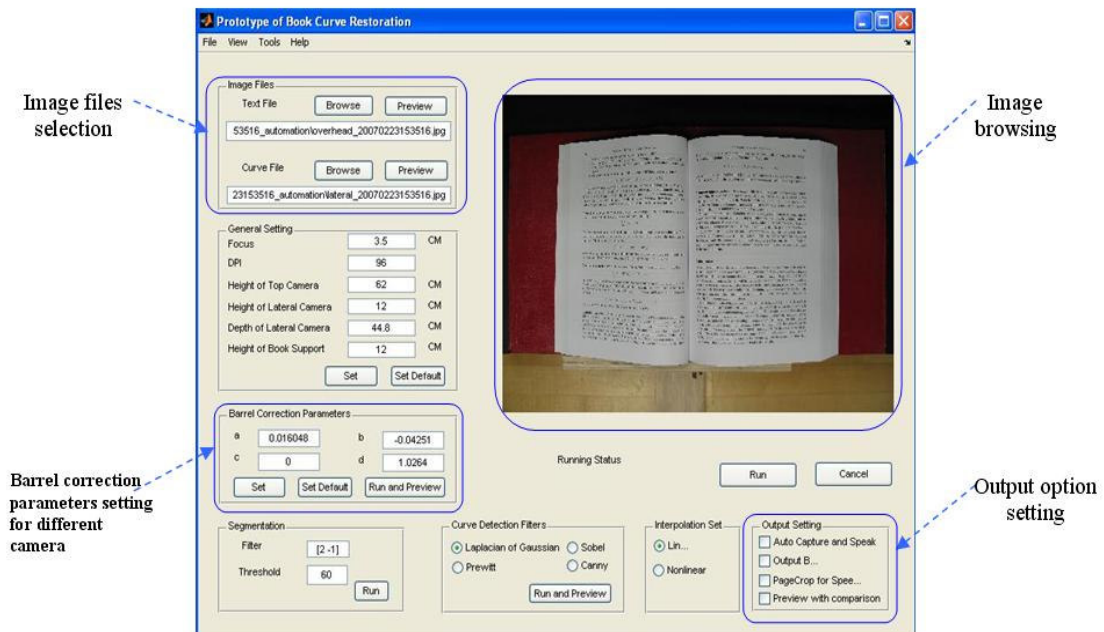


Figure 5-1 GUI for the book reader system

In the "General Setting" section, one can set the focus length of the overhead and the lateral cameras, and the dot per inch (DPI) of the current image. It also provides the configuration of the whole book reader system setup parameters, such as heights of the top camera and lateral camera to the book holder, horizontal distance between the book holder and the lateral camera and height of the book holder. You can set the current parameters by clicking "Set" button, or retrieve the default value by clicking "Set Default" button.

In the "Barrel correction parameters" section, according to the barrel correction algorithm, there are four parameters to be determined. The user can set them in this panel, and hit "Set" button to configure.

For "Segmentation" section, the image segmentation filters and threshold value for image segmentation operation can be configured. The system will generate appropriate filter to detect both left and right boundaries of the segmented images.

In the "Curvature Detection Filters" section, there are four filters, such as *"Sobel"*, *"Prewitt"*, *"Canny"* and the *"Laplacian of Gaussian"* which is the default filter in this case. These are provided for use in the curvature extraction process. The user can choose one filter at a time and hit the button "Run and preview" to see the curvature extraction image and the plot of the real distance between each point in a horizontal line and the overhead camera.

In the "Output option setting" section, there are four options provided, which are: (1)*"Auto capture and speech"*, (2)*"Output b/w image"*, (3)*"Pagecrop for speech"* and (4)*"Preview with comparison"*. The first option allows the system to automatically process all the tasks sequentially. The second option, as it states, will make the system output black and white binary image, which will be fed into the OCR and the document analysis software. The third option provides the function to split the whole page into left and right pages, then inputting to the OCR separately. The last option provides opportunity for users to compare the original image and the final image after all corrections have been accomplished.

5.3 GUI for automatic image acquisition

The GUI for automatic image acquisition can be invoked by the submenu "Acquisition" in the menu "Tools". Through this developed interface, the user can choose the overhead or the lateral camera to control separately, as shown in Figure 5-2.

Once the GUI is invoked, the overhead camera will be in the active status ready to preview and capture images.



Figure 5-2 GUI for automatic image acquisition

This GUI contains a preview area, a selection area and a control area. The preview area is on the left side of the window, which gives a preview to users in real time. Under the image preview plot box, there is a slider for user to adjust the focal length of the current camera in order to obtain a perfect view. The selection area is on the top right side of the window. In this section, you can choose overhead camera or lateral

camera as the current camera through the drop-down menu. Message of operation result will appear under the menu after any operation is invoked. The control area is located on the bottom right side of the window, consisting of four buttons, which are "Get View", "Stop View", "Capture" and "Quit Camera". The preview image will be displayed in real time in the preview area once the button "Get View" is hit and the user can stop the view by clicking "Stop view". Once the button "Capture" is pressed, the camera will do the capture operation, transfer the image to the hard disk and display the captured image in the image browsing area of the main GUI. After this operation, the current active camera will be back into release status and the GUI will be closed automatically, or this can be achieved manually by clicking the button "Quit Camera".

5.4 GUI for image restoration and OCR-TTS

All the steps of image processing can be experimented on if the option "*Auto capture and speech*" is not checked, as shown in Figure 5-3.

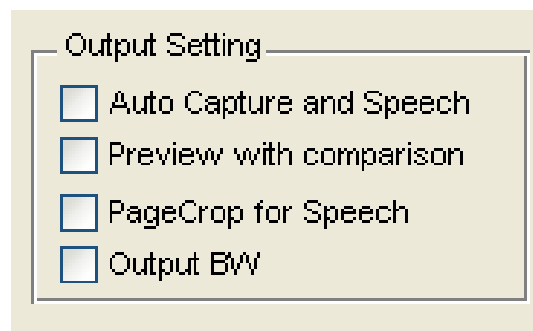


Figure 5-3 Output options of the main GUI

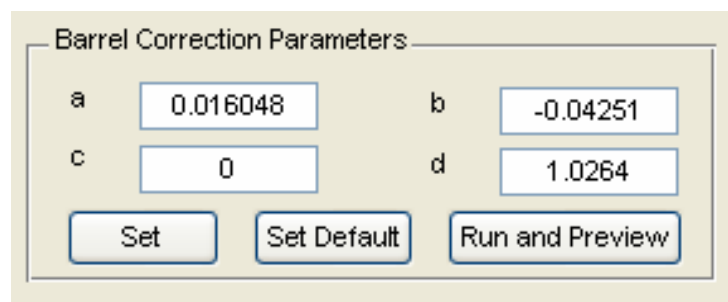
The operational steps include:

- 1) Barrel distortion correction,
- 2) Image segmentation,
- 3) Curvature extraction and real height calculation
- 4) Page curvature distortion correction
- 5) Optical Character Recognition and Text To Speech processes

In the following sections, each step will be introduced in details. The fully automated operation “Auto Capture and Speech” is presented after that.

5.4.1 Barrel distortion correction

The Barrel distortion correction parameters can be set in the "Barrel Correction Parameters" section, shown in Figure 5-4. Recall that these parameters are distinct properties of each lens.



Barrel Correction Parameters

a	0.016048	b	-0.04251
c	0	d	1.0264

Set Set Default Run and Preview

Figure 5-4 Barrel distortion correction parameter configuration

The user can press "Set" to configure the parameters as the current values or press "Set Default" to recover the default parameters setting.

A window with an animation, as shown in Figure 5-5, will be put on display to compare the change of the images before and after the barrel distortion correction, if the button "Run and Preview" is pressed.

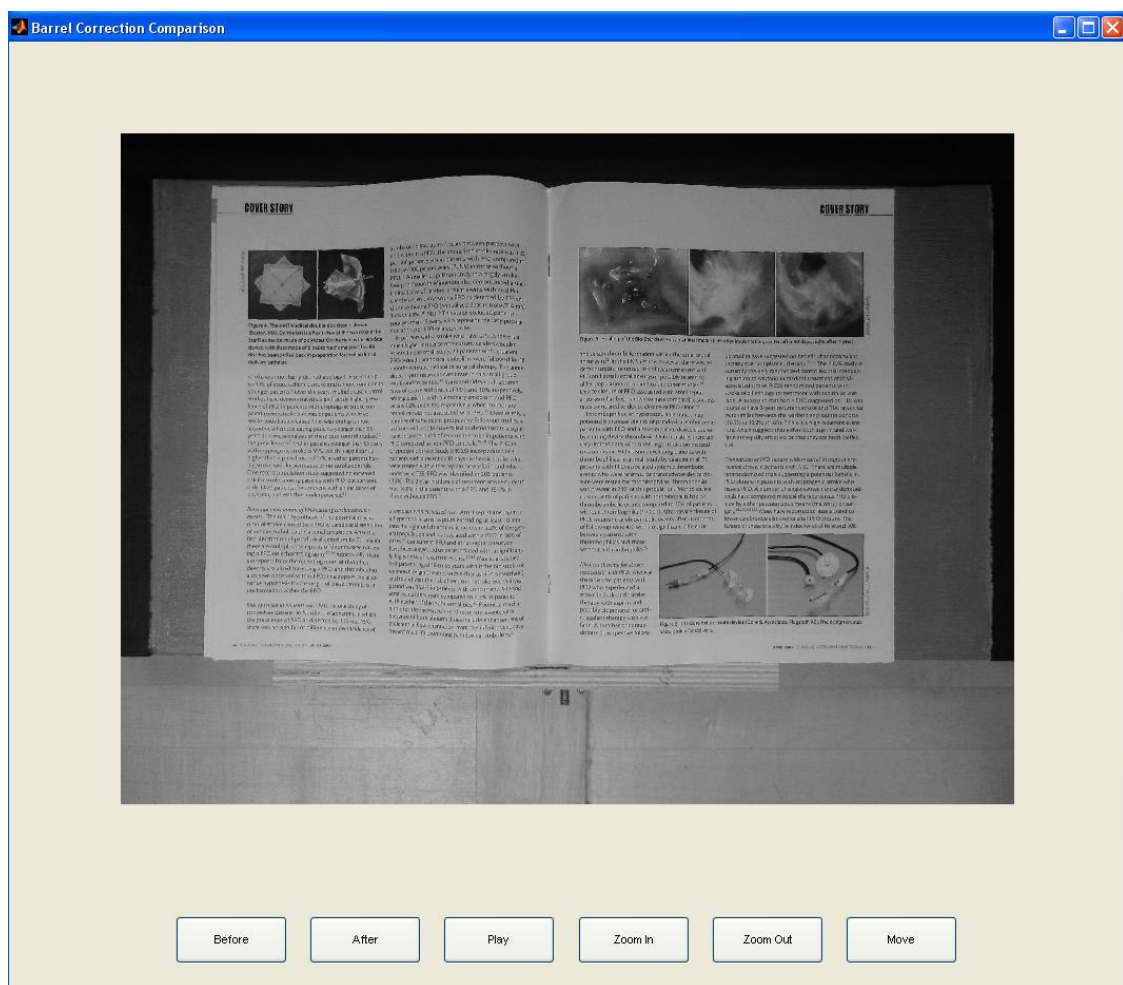


Figure 5-5 GUI for Barrel correction result comparison

As can be seen in Figure 5.5, there are six buttons in this developed GUI that can be used, and they are:

- 1) "Before", to show the original image;
- 2) "After", to show the barrel distortion corrected image;
- 3) "Play", to show the animation of the original and corrected images together;
- 4) "Zoom in" and "Zoom out", to zoom in or out the current image and
- 5) "Move", to move the image to a position of interest.

5.4.2 Image segmentation

In the "Segmentation" section in the main GUI, as shown in Figure 5-6, the parameters for setting first order linear filter and the threshold value can be adjusted to perfect the image segmentation result.

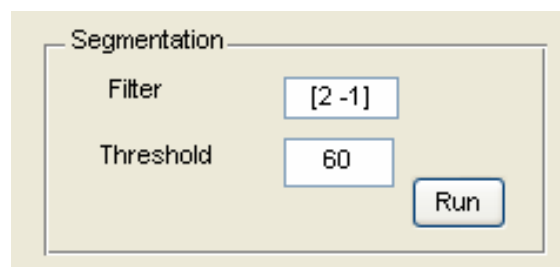


Figure 5-6 Segmentation parameters configuration

The segmentation operation is invoked if the button "Run" is pressed. Another window with the segmentation result will be displayed as in Figure 5-7.

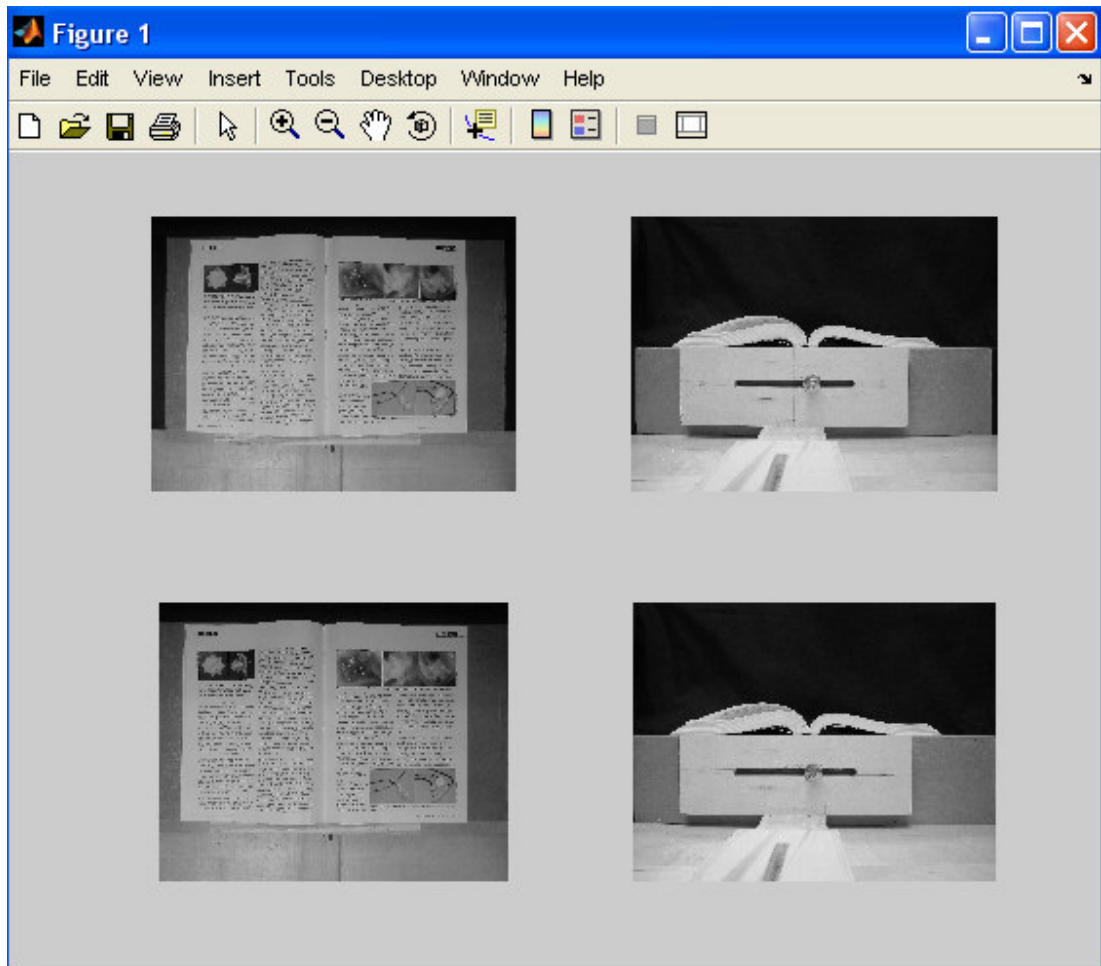


Figure 5-7 Segmentation window to display the segmented overhead and lateral image

5.4.3 Curvature extraction and real height calculation

In the "Curvature Detection Filters" section, as shown in Figure 5-8, there are four high-pass filters provided for the user to choose from to get the perfect curvature image.

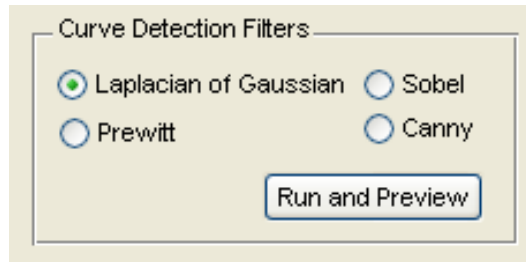


Figure 5-8 Curvature Detection Filter section in the main GUI

A window with a typical curvature extraction result can be displayed as in Figure 5-9 if the user clicks the button "Run and Preview". This result is obtained through using the Laplacian of Gaussian filter as a default in this case.

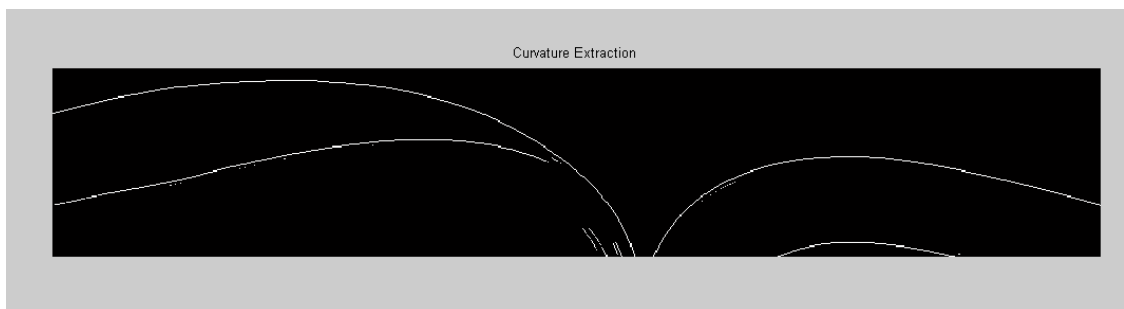


Figure 5-9 Curvature extraction result

Another window, as shown in Figure 5-10, following with the previous window will pop out to display the plot of the real height of each point on the curvature extracted to the overhead camera. From this figure, one can determine that the real height is between 50cm and 45.5cm. The distance between the overhead camera and the book holder is 50cm, and the distance decreases to the minimum value (45.5cm) at the peak of the page curvature.

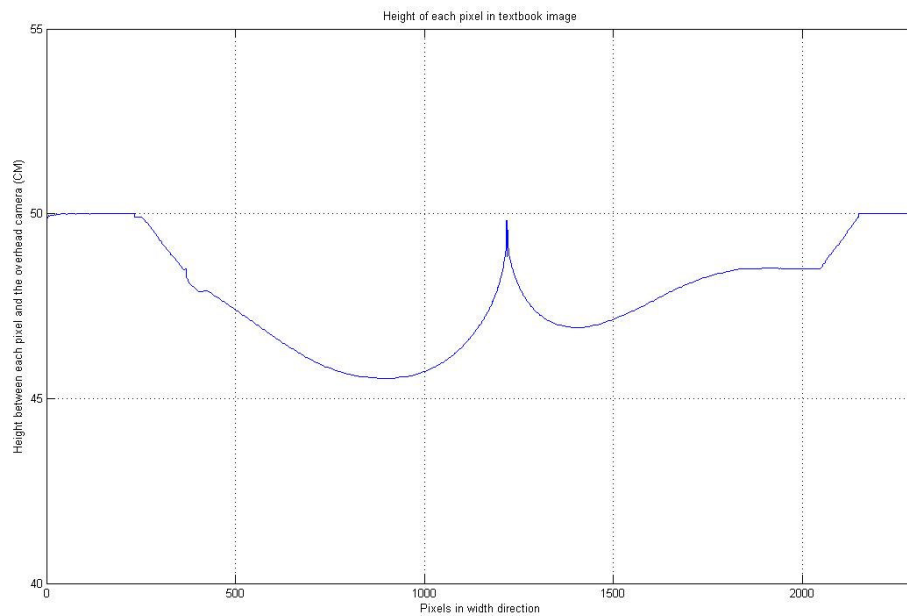


Figure 5-10 Plot of the height of each point on the curve to the overhead camera

After barrel distortion correction and curvature height calculation, the next step is to proceed with the page curvature distortion correction.

5.4.4 Page curvature distortion correction

The "Run" button to invoke the page curvature distortion correction is located under the image browsing area, as depicted in Figure 5-11. The system operation message, which indicates the current status and processing time for each operation, will be displayed to the left of the button.

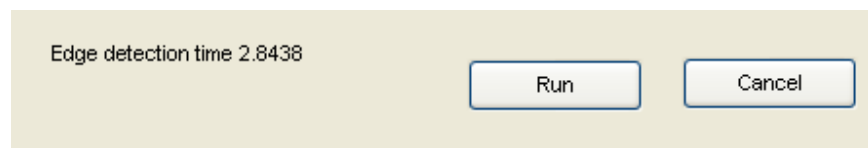


Figure 5-11 Operation warning messages and page curvature distortion correction invoking button

After the "Run" button is clicked, the distortion correction will proceed with updated operation messages. After the correction is done, the original image and the corrected image will be presented together for the user to compare if the option "Preview with comparison" is checked. The comparison between the original and corrected images is presented as shown in Figure 5-12.

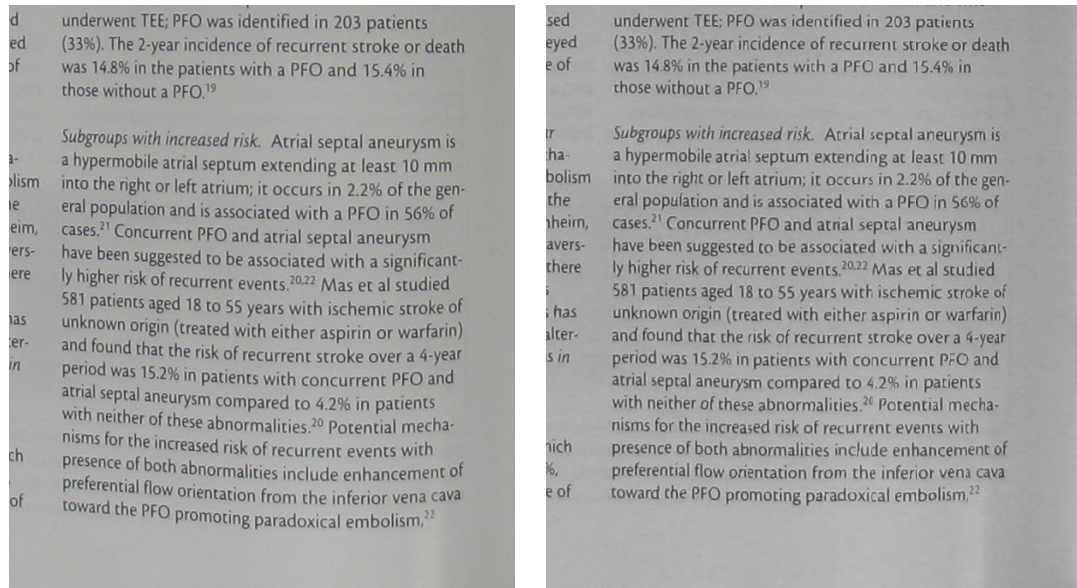


Figure 5-12 Comparison between the original and corrected images

5.4.5 Optical Character Recognition and Text-To-Speech

The Optical Character Recognition (OCR) process includes document image layout analysis and text character recognition. In this implementation, the document image layout analysis GUI is based on MATLAB, and the text character recognition software is based on Microsoft Office OCR library. As shown in Figure 5-13, the GUI consists of two parts: training part and simulation part. Sample document images are used for training the neural network.

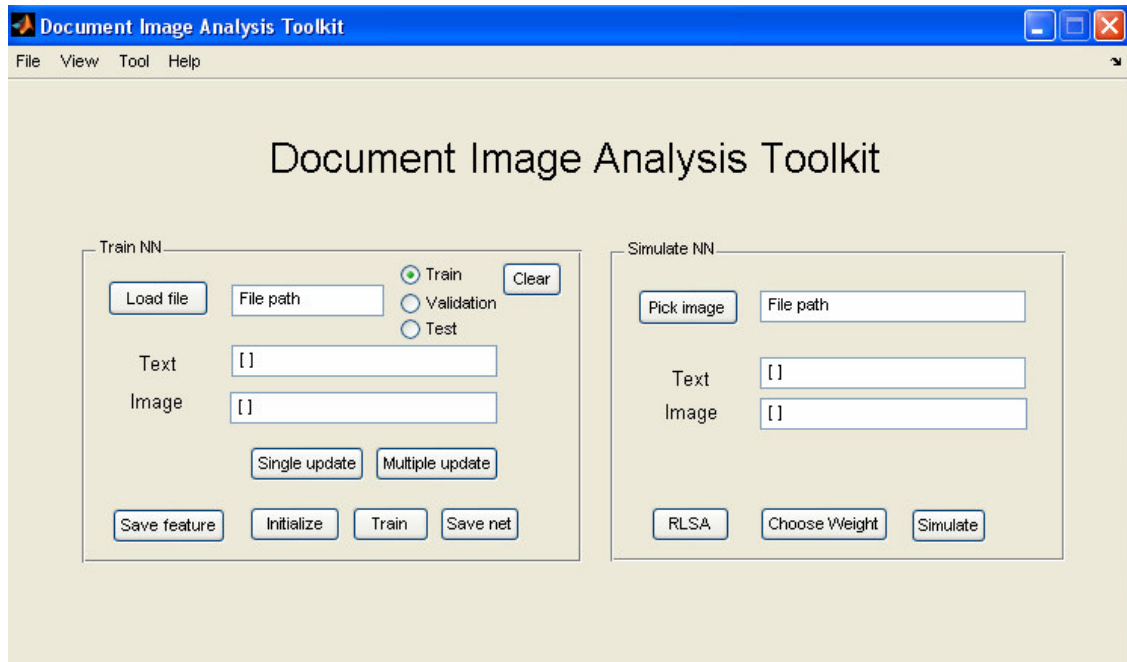


Figure 5-13 GUI for document image analysis toolkit

In the neural network training section, the sample images can be loaded into this software through "Load file" button in Figure 5-14. Another two windows will pop out to show the plot of the feature extraction result and the RLSA result, which was earlier described in Chapter 3. These are shown in Figures 5.15 and 5.16

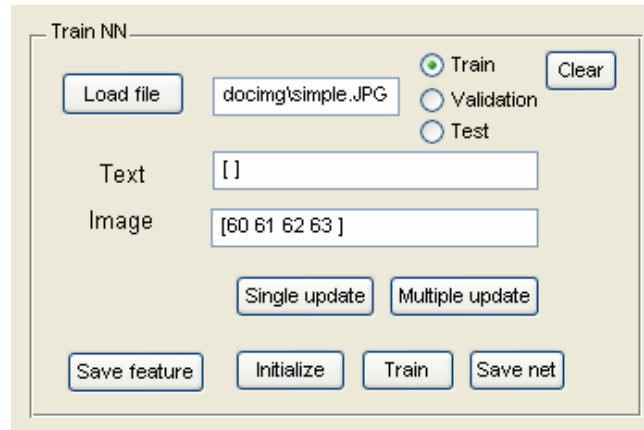


Figure 5-14 Training section of GUI for document image analysis

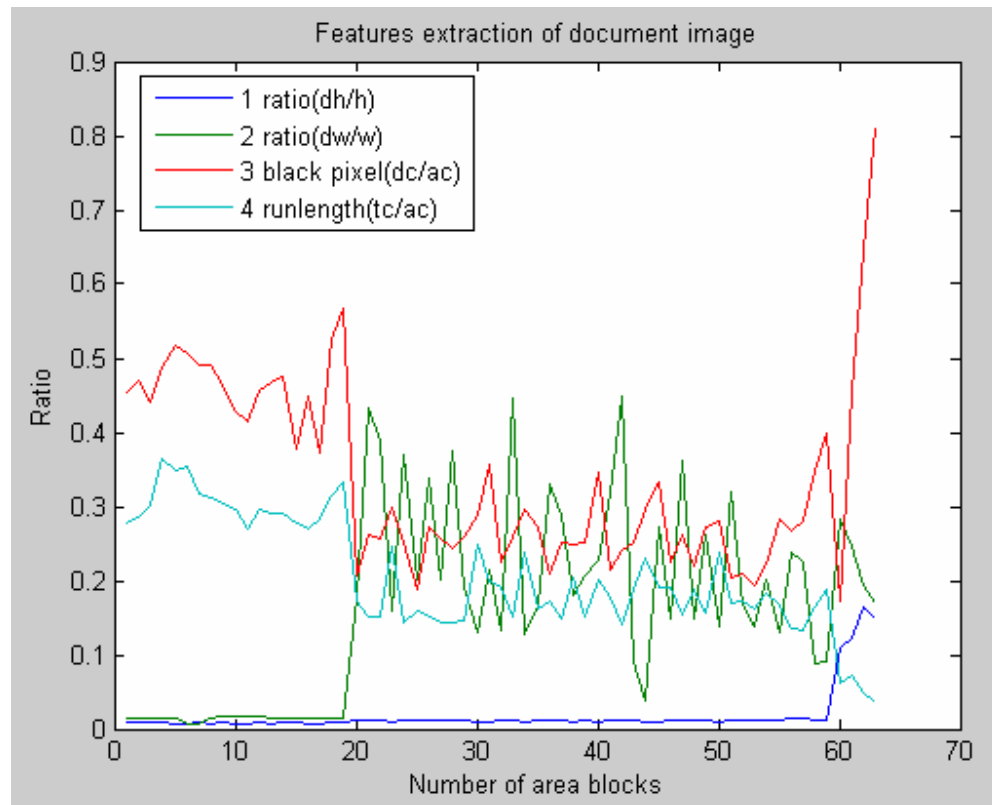


Figure 5-15 Feature extraction result of the image analysis

10	45	The New U.S. STCW Endorsement
4	46	Mr. R. Jim Luskous
11	47	Why Change Merchant Mariner's Documents
4	48	Mr. R. Jim Luskous
12	49	Obtaining a Mariner's Credential
5	50	Mr. Stewart Waller
	56	Regional Exam Centers
1	24	An Overview of Regional Examination Center
2	25	Lt. Cmdr. Anthony C. Curry
2	26	Regional Examination Center DIRECTORY
	58	Subunits
13	27	The Marine Safety Center
5	28	Capt. Ron Nash and Lt. Cmdr. Doug Simpson
3	29	The U.S. Coast Guard Marine Safety Laboratory
5	30	Dr. Wayne R. Lottinor
14	29	The Long Journey Home
3	30	Mr. Thomas Wain
	57	Industry Partnerships
15	31	Introduction to Maritime Security
2	32	Mr. Christopher E. Krus
16	21	Standards of Training, Certification, and Watchkeeping
3	22	Mr. Mark Hansen
17	23	Merchant Marine Personnel Advisory Committee
2	24	Mr. Anderson-McGraw
4	33	National Maritime Center and Military Sealift Command
3	34	Mr. Frank Bondar
5	35	Mariner's Perspective
3	36	Dr. Michael Landrey and Lt. Cmdr. Todd Duffel
	58	On Deck
6	37	Assistant Commandant's Perspective
3	38	Rear Adm. T. C. Simon
7	39	Chaplain's Point of View
5	40	Capt. Ernest J. Ford
18	41	U.S. A Feedback Mechanism
4	42	Mr. Bill Abernathy and Ms. Kriste Stromberg
19	43	Mariner's Seabay: Ranks of the National Maritime Center
5	44	Captain Robert Stanley Bate
	55	Nautical Queries
100	45	Engineering
20	46	Law



60
Back Cover Credit
 Robert Cumming Photography
Icon Credits:
 All are USCG Illustrations.
 Copyright © 2005 USCG and its
 licensors.

Figure 5-16 RLSA result for the sample document image

The user sets the box number into the text box in the training section, as shown in Figure 5-14, and trains the neural network by clicking the button "train". The training performance result will be shown as in Figure 5-17, and the coefficients of the neural network will be updated and saved in the same directory with this program.

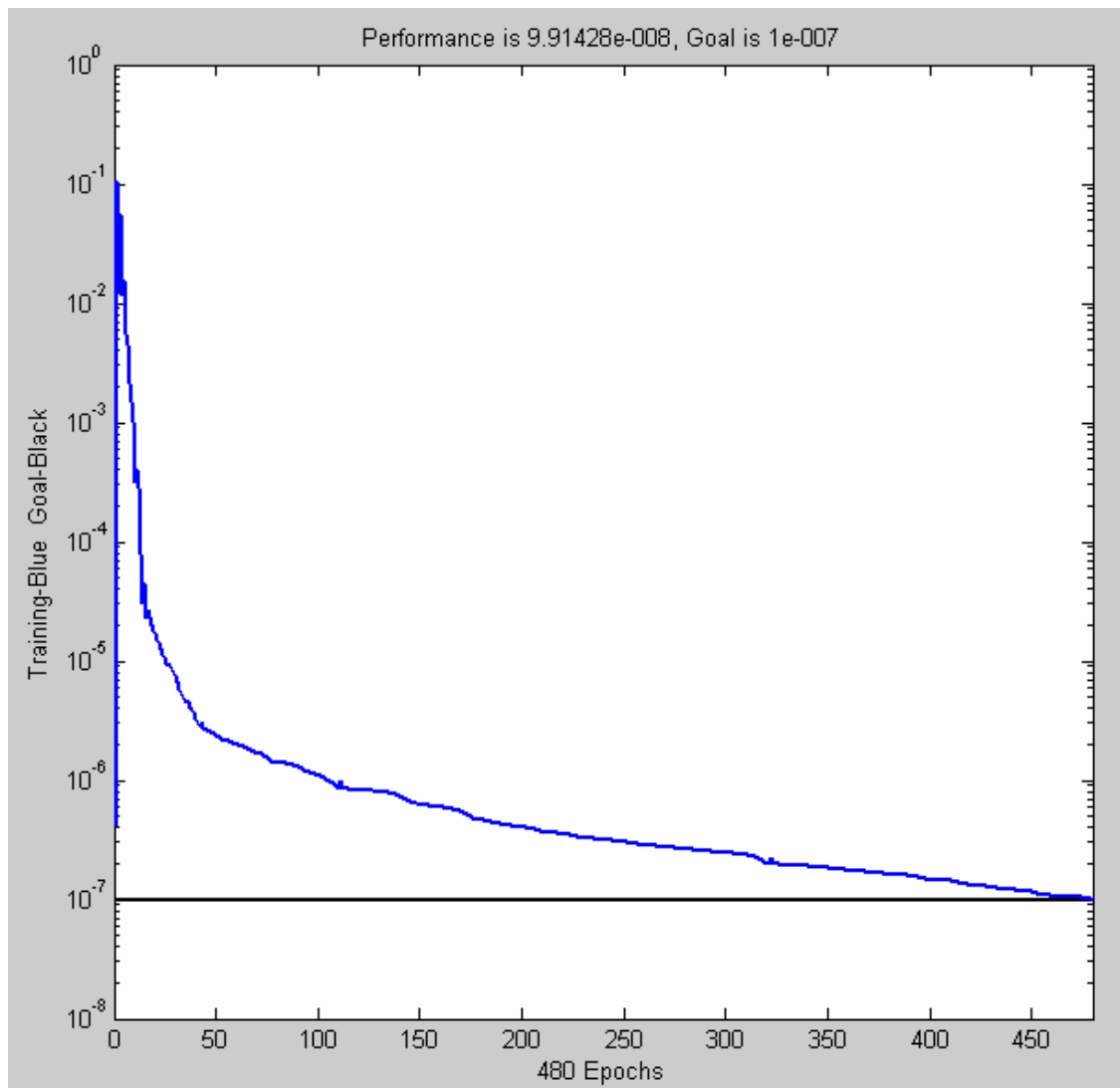


Figure 5-17 Training result

After the neural network has been trained with enough number of sample images, a test image can be fed into this network to simulate the layout. Press the "Simulate" button in the simulate section, and the layout result will be provided as shown in Figure 5-18, which shows the segmentation of the image into text segments and other graphical segments. Now the text part can be input into the OCR and TTS modules for text recognition and speech generation. A C# program of OCR-TTS based on Microsoft Office Document Imaging Library (MODI), Microsoft Word Object Library and Microsoft Speech SDK is embedded into this book reader system. With operation messages updates for each processing step, as shown in figure 5-19, this specific page can then be vocalized by the synthesis voice.

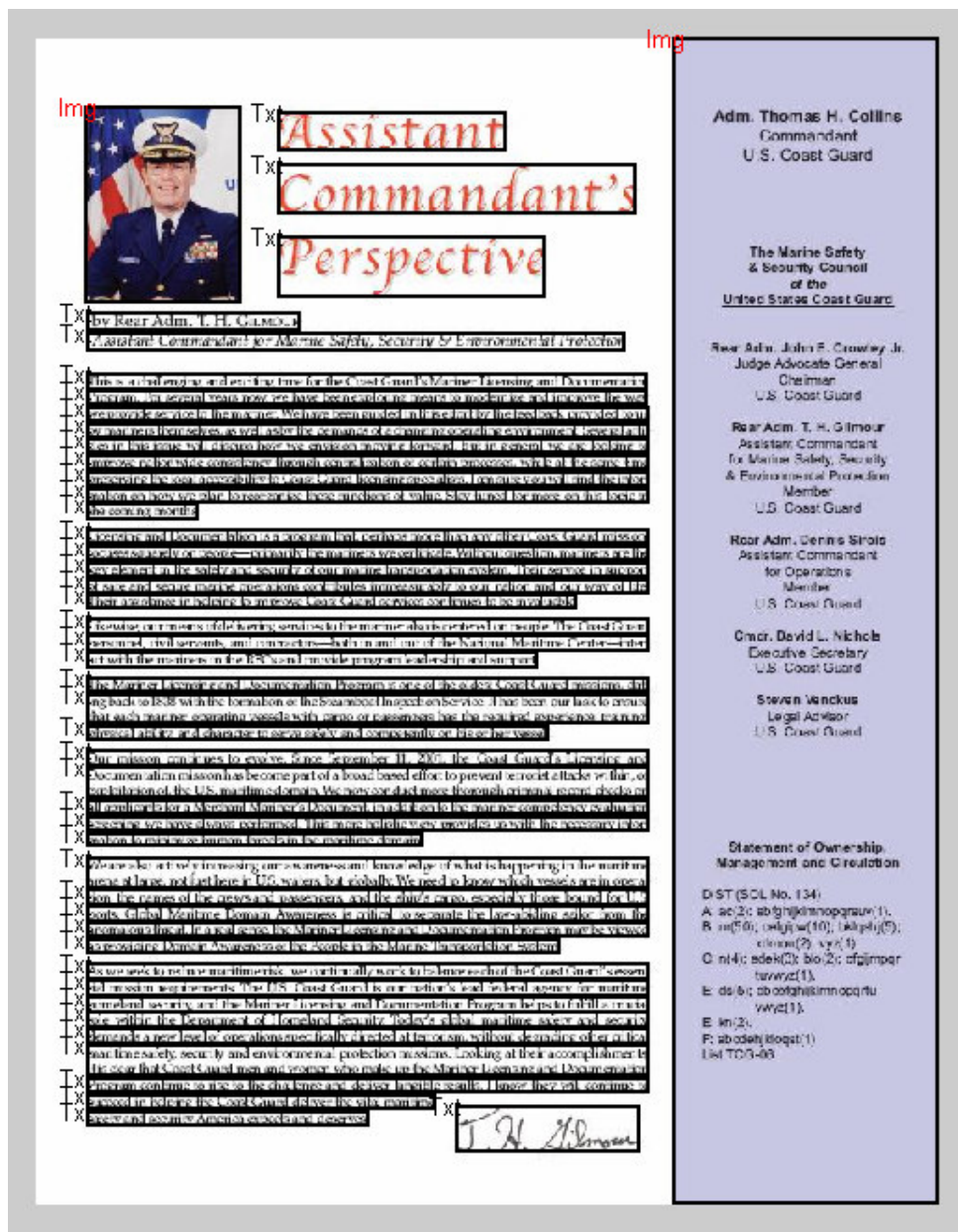
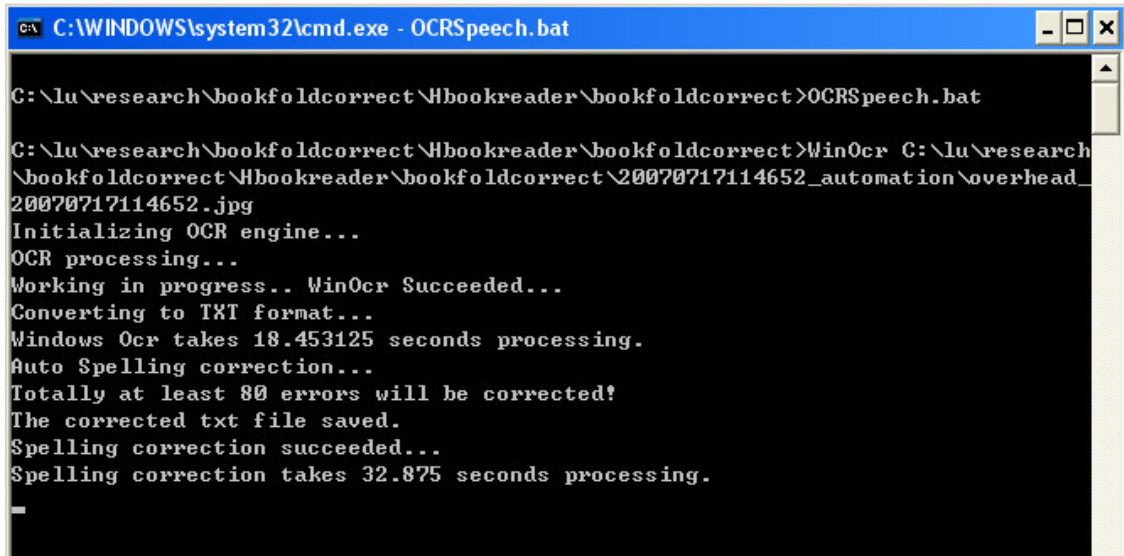


Figure 5-18 Simulation result for document image analysis



```
C:\WINDOWS\system32\cmd.exe - OCRSpeech.bat

C:\lu\research\bookfoldcorrect\Hbookreader\bookfoldcorrect>OCRSpeech.bat

C:\lu\research\bookfoldcorrect\Hbookreader\bookfoldcorrect>WinOcr C:\lu\research\
\bookfoldcorrect\Hbookreader\bookfoldcorrect\20070717114652_automation\overhead_
20070717114652.jpg
Initializing OCR engine...
OCR processing...
Working in progress.. WinOcr Succeeded...
Converting to TXT format...
Windows Ocr takes 18.453125 seconds processing.
Auto Spelling correction...
Totally at least 80 errors will be corrected!
The corrected txt file saved.
Spelling correction succeeded...
Spelling correction takes 32.875 seconds processing.
```

Figure 5-19 OCR and TTS embedded in the interface with processing messages

In Figure 5-19, it is indicated how long it takes to process the Windows OCR and text spell checking and how many words it corrected with its built-in dictionary.

5.4.6 Fully automatic operations

If the option "Auto Capture and Speech", as shown in Figure 5-3, is checked, the system will invoke all the tasks described above automatically, without any other interference by the user. At the end of this process, the synthetic voice read out will be provided to the user. We expect that this approach is the one that will be used by persons with blindness to minimize any extraneous effort in using such an interface.

CHAPTER 6

RESULTS ANALYSIS OF THE INTEGRATED BOOK READER INTERFACE

6.1 Introduction

This chapter presents the results of the automated book reader, and analyzes and compares them with the results from other products. The computational requirements for each step as well as for the entire process are assessed. The recognition rates of text before and after image correction are provided in context with the highly effective and commercially available OCR and TTS software modules.

6.2 Results and Outcome Assessments

As mentioned earlier, since the BookEye system is the only product in the current market that provides page curvature distortion correction, a comparison of the corrected image by the BookEye and the proposed book reader is provided for initial visual appreciations and for a worthwhile overall assessment of the proposed technology.

In the results presented in Figures 6-1 through 6-4, it is obvious that the results of the proposed book reader interface is much better than the ones produced through the BookEye with regards to the contentious problem of page curvature. For some uncertain reasons, it appears that the BookEye technology seems to use a model to correct the

original image, and it appears that the model cannot fit all the page curvatures very well every time.

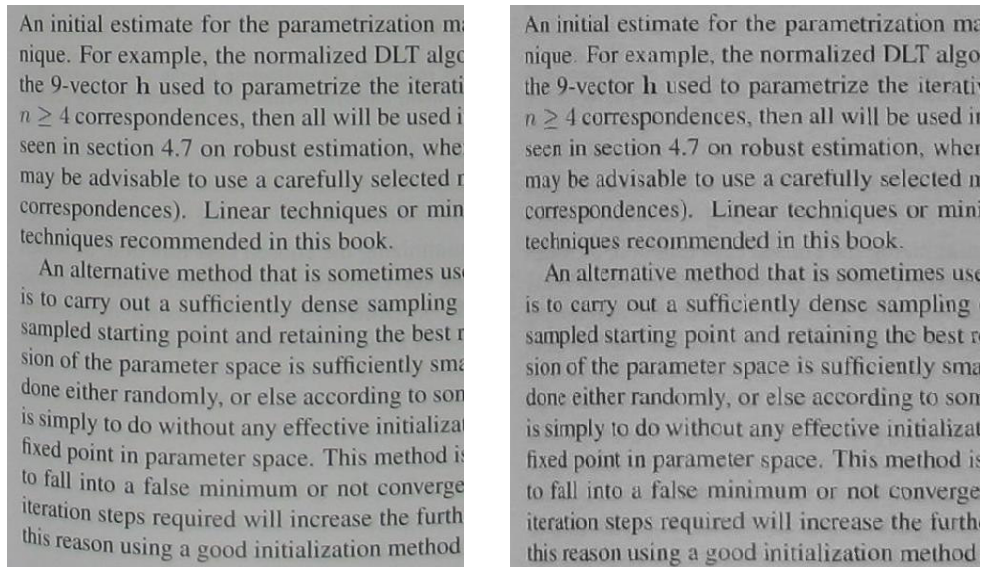


Figure 6-1 (Left) Distorted image directly from camera and (Right) Corrected image of the proposed algorithm

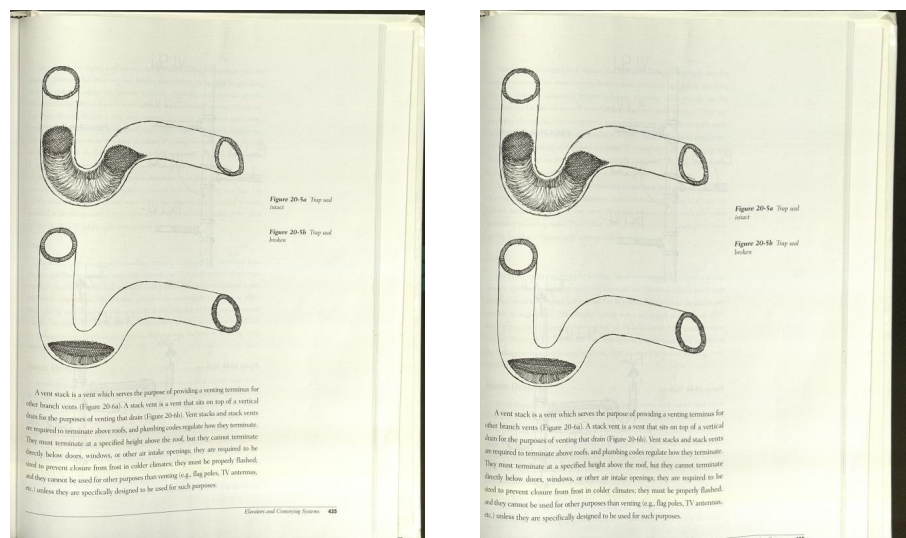
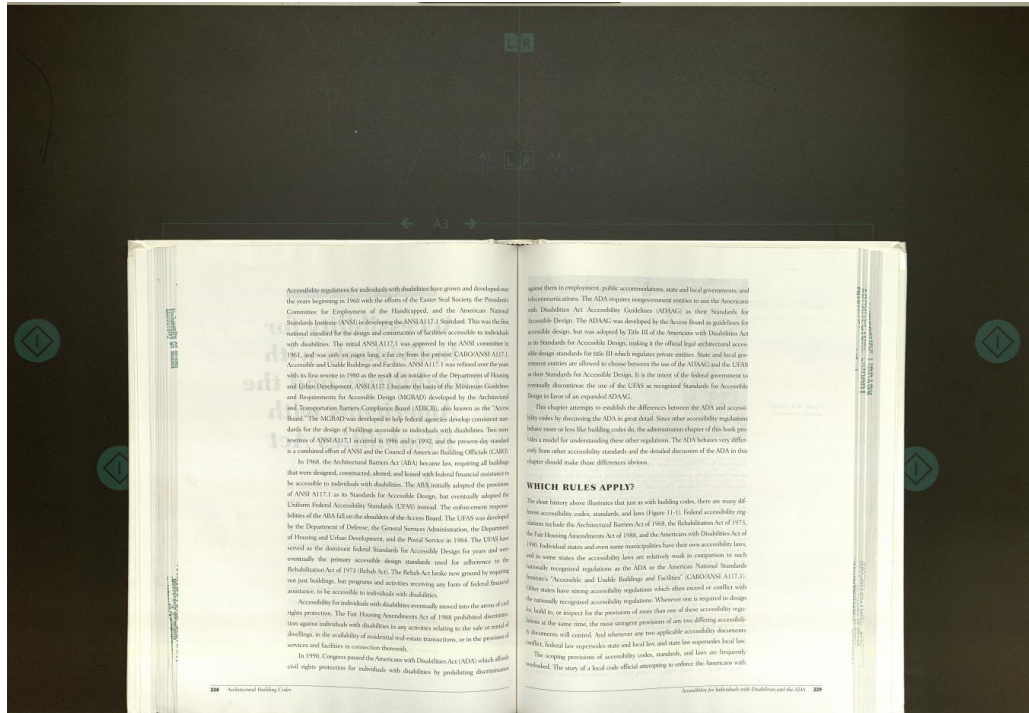
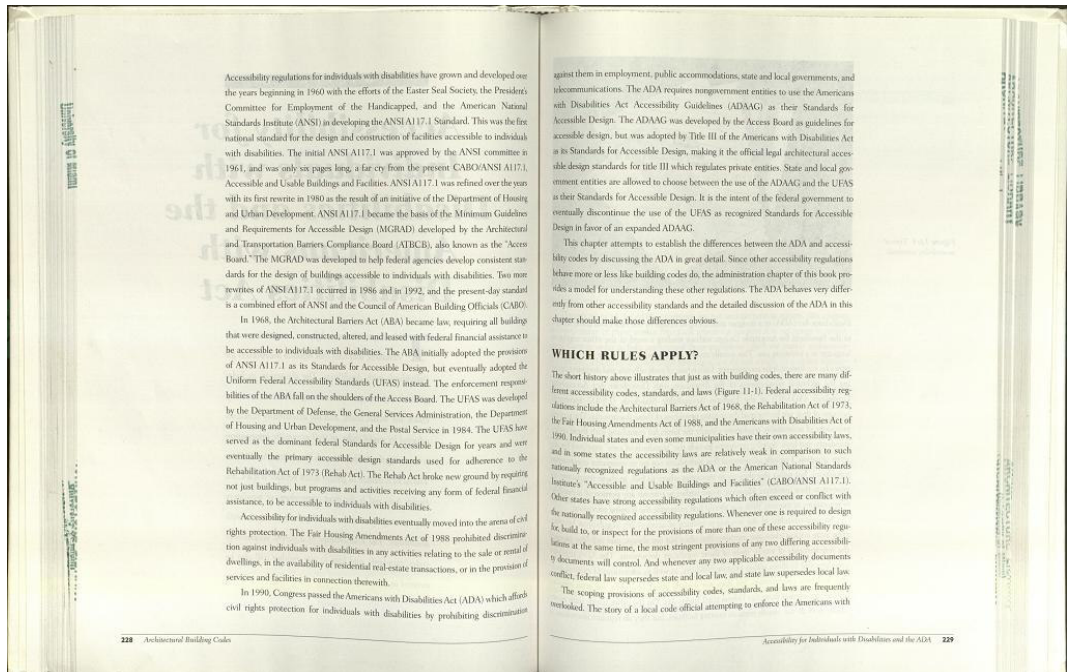


Figure 6-2 (Left) BookEye scan result with cropping original image and (Right) book fold corrected image



(a)



(b)

Figure 6-3 BookEye Scan of (a) original image and (b) book fold correct image

6.2.1 Computational Requirements

With this initial comparison with BookEye system for the page curvature correction, other processing functions are now assessed in a more comprehensive manner. A comparison of the time processing requirements in the image acquisition and distortion correction by the Interface and by the BookEye is given in Table 6.1. The program is running in the MATLAB environment on a PC equipped with Windows XP and Intel Pentium 4 CPU.

Table 6-1 Time consumed in seconds for image acquisition and distortion correction

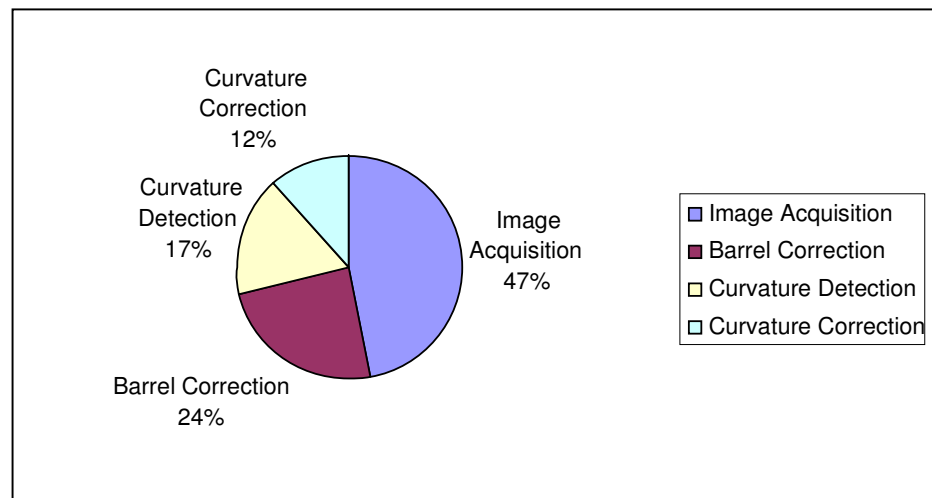
Time processing requirements for the book reader interface				BookEye
Image Size	576*768	1152*1536	2304*3072	2626*3007
Image Acquisition(s)	5.36			N/A
Barrel Correction(s)	0.39	0.81	2.79	N/A
Curve Detection(s)	0.18	0.61	1.95	N/A
Curvature Correction(s)	0.21	0.40	1.34	N/A
Total(s)	6.14	7.18	11.44	Around 10s to display

From these results, it is evident that the processing requirement on the image acquisition is much more significant than any other processing step. The reason is that there are two Canon cameras that are used to acquire the overhead and lateral images. To avoid any unreliability while releasing shutters for more than one camera simultaneously

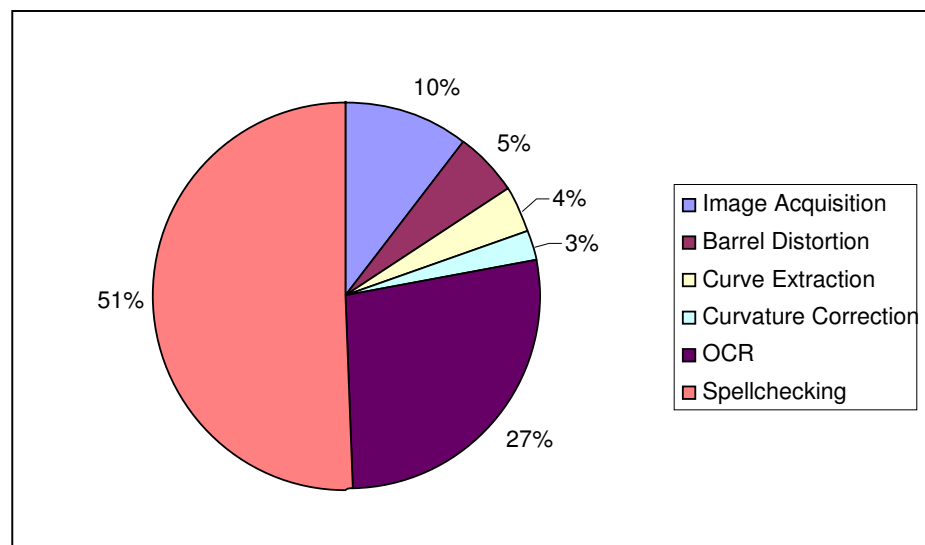
while using Canon SDK 7.0, the operations "open", "release" and "close" for every picture taken should be well synchronized. The time consumed in the image acquisition, however, will be dramatically decreased with the anticipated Canon release of its latest SDK version, which is expected to reliably support multiple cameras at the same time. For barrel distortion correction and curvature distortion correction, an indexing matrix to facilitate MATLAB matrix performance advantage was developed. In addition, memory pre-allocation and sub-blocking large matrix algorithm is designed help improve the processing performance.

In Figure 6.4 (a), processing time allocations are provided for the different components of the distortion correction process along with the image acquisition process. In Figure 6.4 (b), processing time allocations are provided for the different components of the distortion correction process along with the image acquisition process and together with OCR and spellchecking functions. These graphs allow an appreciation of the different time percentages within the whole context of the process from the image acquisition step to the final audio read-out.

Figure 6.5 illustrates how the different image resolutions affect the processing time in terms of the time required for image acquisition plus the time required for image distortion correction, as earlier indicated in Table 6.1.



(a) Processing time involving image acquisition and image distortion correction



(b) Processing time as in (a) but including OCR and spell checking applications

Figure 6-4 Processing time for any image with dimension of 2304*3072

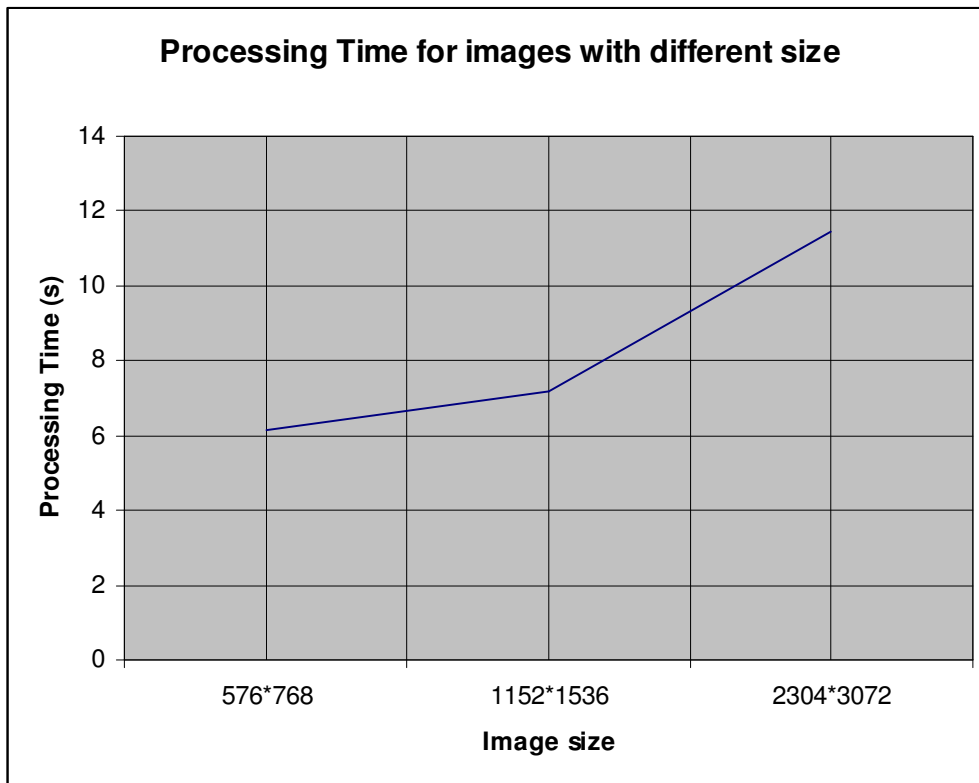
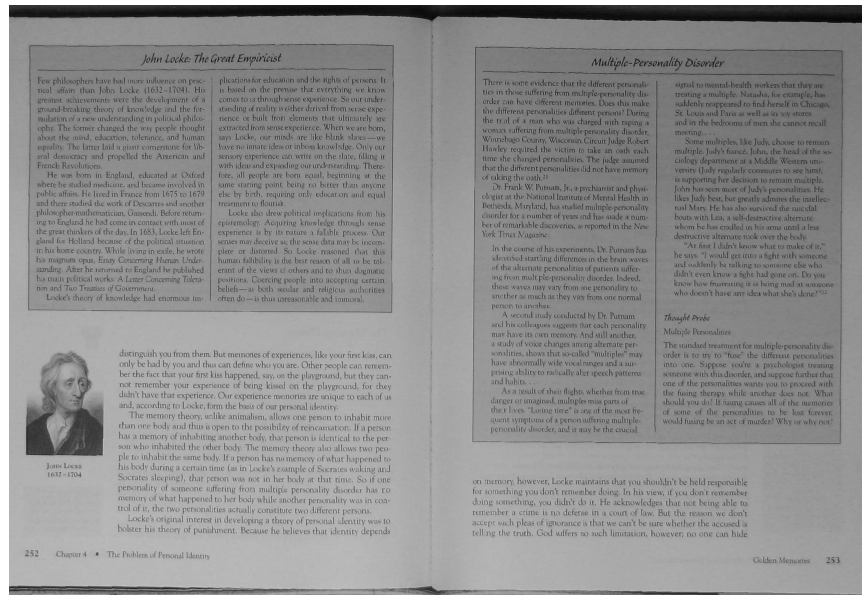
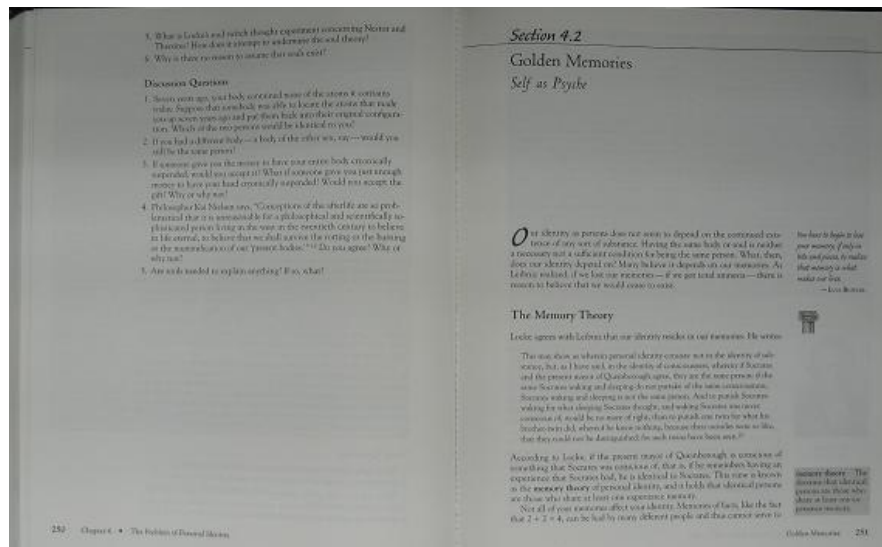


Figure 6-5 Processing time as a function of image size

It should be noted that for the OCR readout, the time required is a function of the extent of text provided in any given two pages. Figure 6-6 shows two examples with different number of words in each example and the required time for the readout of these pages. The user could change the speed of the read out to suit their needs. On the average, the readout speed is 2.5 words per second or 150 words per minute (wpm), which is the default speech rate of Microsoft TTS SDK 5.1. This is in the so-called "safe range" for normal speech. Microsoft TTS SDK 5.1 provides 3 times faster or lower rates of the default rate with 10-stop adjustment.



(a) Typical book page with 1217 words in 2 pages: (The readout took 8 minutes and 45 seconds, which is 525 seconds, for a 2-page with 1217 words)



(b) Page with small amount of words (574): The readout took 3 minutes and 25 seconds, which is 205 seconds, for a 2-page with 574 words

Figure 6-6 Readout time versus the number of words in the 2 pages

A comparison for the time processing requirements for the OCR and spell checking processes by the book reader interface, ABBYY and OmniPage is given in Table 6-2. For OCR, ABBYY and OmniPage are a little bit faster than the Interface. But they have not provided automated spell checking and correction functions so that it requires the manual operations to decide the correction from all the suggestions proposed by the software.

Table 6-2 Time processing requirements comparison in OCR and Spell check by the proposed Book Reader Interface in Contrast to ABBYY and OmniPage

	Book reader interface	ABBYY	OmniPage
Image Size	2304*3072		
OCR	13.96 s	About 10 s	About 11 s
Spell Checking	25.87 s	Manually	Manually

6.2.2 Recognition performance analysis

6.2.2.1 Book with common paper material

In this section, 10 different document images and their corrected counterparts are used as input to the ABBYY FineReader 8.0 and the Interface WinOCR for recognition rate (RR) comparisons. The 10 document images are captured with different book curvature angles. The recognition rates of WinOCR and ABBYY and on the original image and the corrected image are given in Tables 6-3 and 6.4, respectively, and their

plots are as shown in Figure 6-7. The average recognition rates for the original image are 54.67% for ABBYY and 43.84% for WinOCR. These recognition rates increase significantly when the corrected images are input to these same ABBY and WinOCR modules to attain 97.14% for ABBYY and 97.08% for WinOCR. This clearly validates the merits of the proposed book reader interface and its proposed image distortion correction.

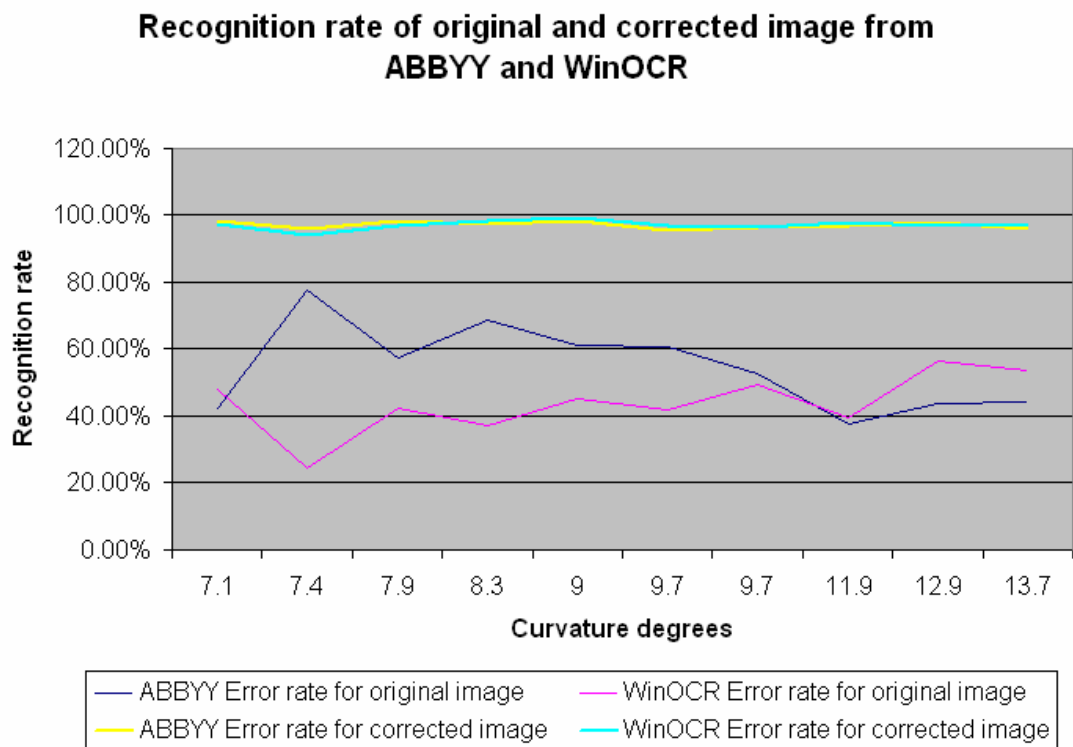


Figure 6-7 Plot of Recognition rate of ABBYY and WinOCR

Table 6-3 Comparison of Interface WinOCR recognition rate of original and corrected images

#	Curve angle (Degree)	Word count	error of original using WinOCR*	RR of original using WinOCR*	error of corrected image using WinOCR*	RR of corrected image using WinOCR*
1	7.1	464	223	48.06%	12	97.41%
2	7.4	383	94	24.54%	22	94.26%
3	7.9	944	400	42.37%	27	97.14%
4	8.3	963	357	37.07%	18	98.13%
5	9.0	496	224	45.16%	4	99.19%
6	9.7	456	192	42.11%	15	96.71%
7	9.7	513	254	49.51%	19	96.30%
8	11.9	898	357	39.76%	20	97.77%
9	12.9	567	320	56.44%	17	97.00%
10	13.7	898	480	53.45%	28	96.88%
<i>Avg.</i>	<i>9.76</i>	<i>658</i>	<i>290</i>	<i>43.84%</i>	<i>18</i>	<i>97.08%</i>

* WinOCR is the application embedded in the interface, written in C#, which call Microsoft OCR Type Library and Word Spellchecking Library.

Table 6-4 Comparison of ABBYY recognition rate of original and corrected image

#	Curve angle (Degree)	Word count	error of original using ABBYY	RR of original using ABBYY	error of corrected image using ABBYY	RR of corrected image using ABBYY
1	7.1	464	267	42.46%	8	98.28%
2	7.4	383	85	77.81%	15	96.08%
3	7.9	944	400	57.63%	15	98.41%
4	8.3	963	300	68.85%	24	97.51%
5	9.0	496	192	61.29%	9	98.19%
6	9.7	456	180	60.53%	21	95.39%
7	9.7	513	243	52.63%	17	96.69%
8	11.9	898	560	37.64%	29	96.77%
9	12.9	567	320	43.56%	11	98.06%
10	13.7	898	500	44.32%	35	96.10%
<i>Avg.</i>	<i>9.76</i>	<i>658</i>	<i>304</i>	<i>54.67%</i>	<i>18</i>	<i>97.14%</i>

6.2.2.2 Book with glossy paper material

Books made of glossy paper material are as anticipated more troublesome due to the specular reflection of incident light rays. But this problem is encountered only if the overhead camera is set with built-in flash on. As shown in Figure 6.8, the indicated

segment in dashes is highly specular because of the over-exposure with the flash on and consequently the text information in that segment is lost or highly unreadable. To avoid this from happening, we knowingly set the built-in flash of the camera OFF, as reflected in Figure 6.9. Four glossy pages with different curvature angles are thus tested and the results are listed in Tables 6-5 and 6-6.

Table 6-5 Recognition rate for glossy book with flash off

	Angle of Page Curvature	Character Count	Recognition Rate for Original Image	Recognition Rate for Corrected Image
1	7.0	1022	62.01%	98.45%
2	8.2	1254	55.54%	98.22%
3	11	397	42.29%	97.61%
4	11.5	849	34.12%	98.35%
Avg.	9.4	880	48.49%	98.16%

Table 6-6 Recognition rate for glossy book with flash on

	Angle of Page Curvature	Character Count	Recognition Rate for Original Image	Recognition Rate for Corrected Image
1	7.0	1022	59.21%	95.30%
2	8.2	1254	51.38%	94.57%
3	11	397	37.41%	91.62%
4	11.5	849	30.22%	91.23%
Avg.	9.4	880	44.55%	93.18%

It is evident from these results that the book reader interface handles the glossy paper material just as well as the common non-glossy white paper with the flash set to be OFF.

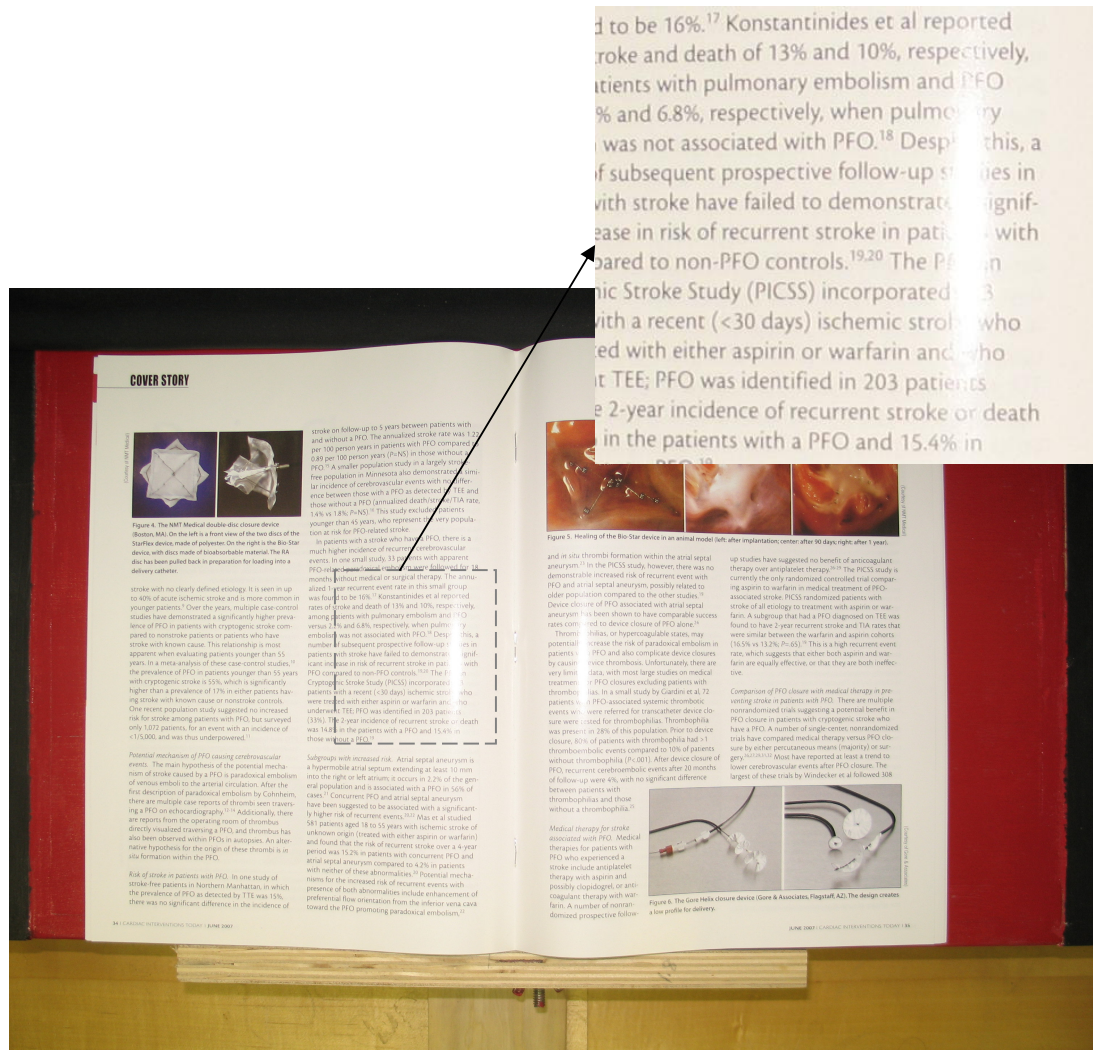


Figure 6-8 Overhead capture of the glossy book with flash on and the detail of the over-exposure part

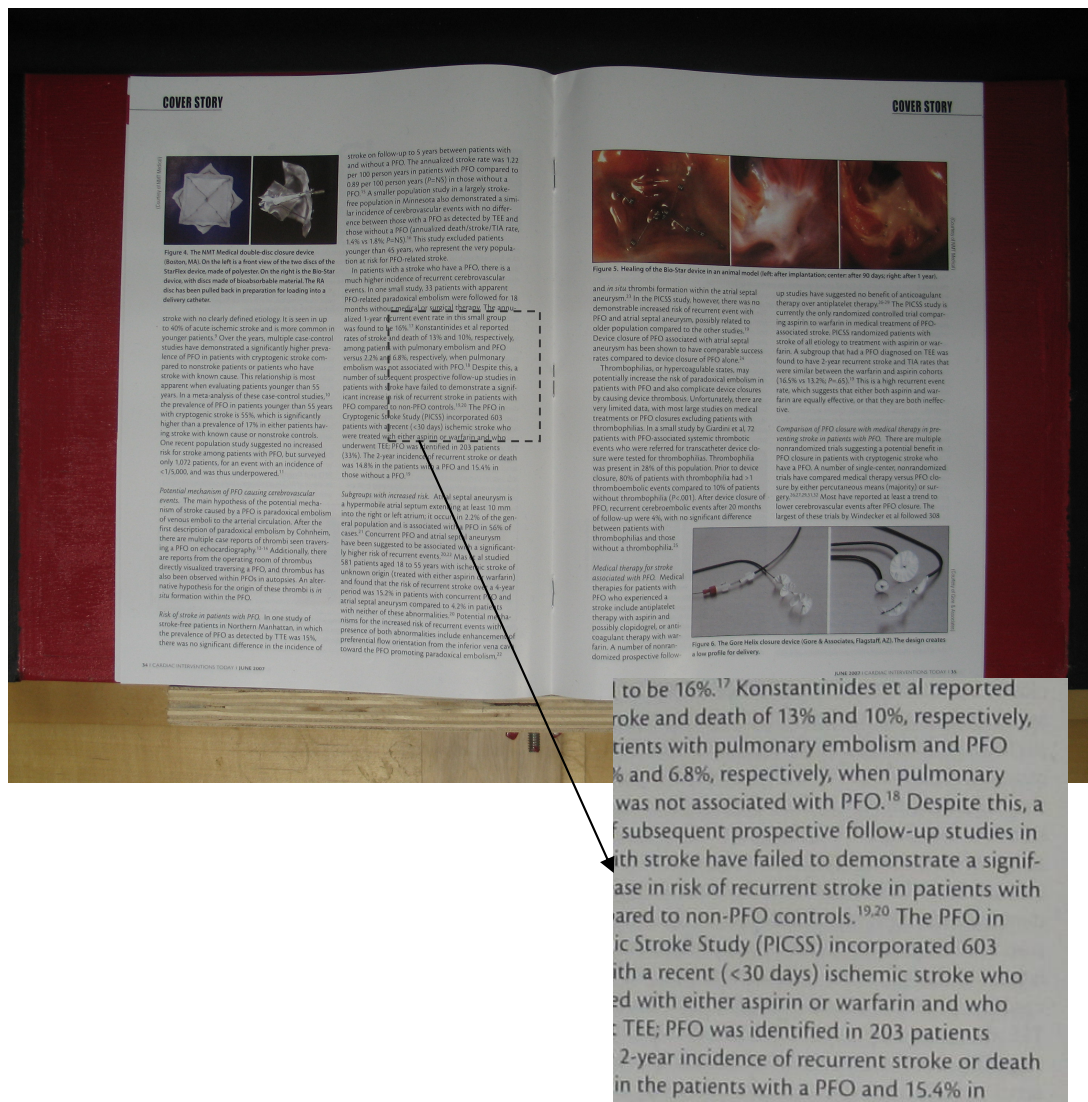


Figure 6-9 Overhead capture of the glossy book with flash on and the detail of a part of the image

6.2.2.3 Recognition rate comparison for books with different font and size

For the OCR part of this interface, Microsoft Office Document Imaging (MODI) 11.0 Type Library is embedded for the document image recognition. It supports all the font families in the Microsoft Word and different font sizes. Currently, the interface cannot handle document image with too many different font sizes, such as headings which will be regarded as a graph in the image. And the interface cannot retain the layout information.

Here three document images with different fonts are tested. Moreover, one document image with different font family and font size is also tested. The results are given in Table 6.7.

Table 6-7 Recognition for different font family

	Image with Garamond Font	Image with New Roman font	Mixed Font Family and Font Size
Words	898	765	1210
Recognition Rate for Original Image	49.51%	52.31%	42.00%
Recognition Rate for Corrected Image	97.61%	96.35%	78.51%

In this experiment, it is determined that for a document image with single font family and font size, the recognition rates are almost the same with the results given earlier in Table 6-3. For the document image with mixed font families and font sizes as reflected in Figure 6-10, the recognition rate reduces to 78% on the average, which is obviously much lower than the average rate of 97% listed in the Table 6-3. What is learned from here is that an initial segmentation into different text blocks with consistent family fonts and font sizes will overcome this problem.

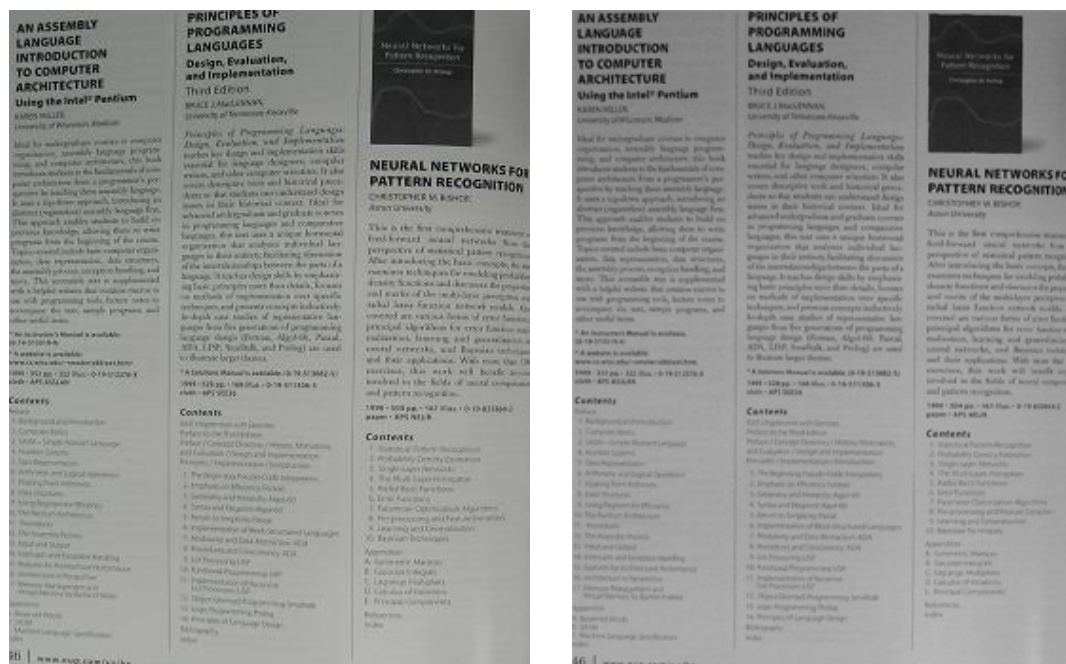


Figure 6-10 Original image (Left) and corrected image (Right) with multi-font and multi-size, including Helvetica Font, and New Roman

6.2.2.4 Book pages containing segments of shaded text

At times in different books, authors or book makers may choose to shade with different gray scales parts of text as ways to delineate the importance or the relevance of that segment in contrast to other segments. An example is shown in Figure 6.11. When the corrected image of this page was fed for a read out, it was found that the reading accuracy of the shaded area was just as good as those text segments which were not shaded.

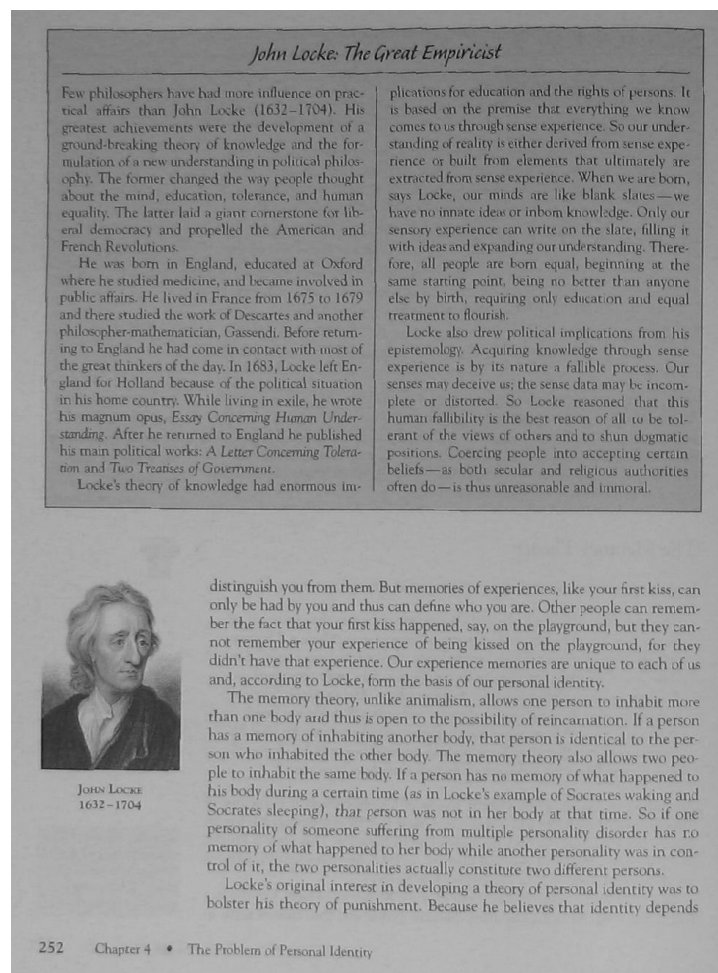


Figure 6-11 A document page with text segments of different gray scales

CHAPTER 7

CONCLUSION

The accomplishments of this dissertation synthesize to the design and implementation of an automated book reader that integrates the processes of (a) automated image acquisition through the use of two inexpensive cameras, (b) image distortion correction through the development of specialized algorithms; (c) integration of sophisticated Optical Character Reading (OCR) software available for download free of charge; (d) integration of the Microsoft error correction, (e) implementation of a segmentation algorithm in order to delineate text from other forms of graphics including mathematical formulas; and (e) automated generation of an audio read-out.

As a result of all these efforts, a working prototype was built encompassing a friendly graphical user interface that allows the user to monitor the working conditions of the different steps of the book reader design. Such an interface serves also as means to calibrate and set up the system for an optimal result in its initial first use.

What follows is a retrospective on the major outcomes and the prospects for advancing the merit of this research in order to extend its real-world practicality as we seek universal access to documentation.

7.1 Major Outcomes

The main contribution of this dissertation is in the development of an integrated approach to resolving the problem of page curvature and other image distortions in order to produce an automated read out of books. There is a direct implication to the digitization process as the book pages are now stored in memory for future retrieval. This outcome is further enhanced by the facts that:

- The system as proposed is highly economical costing one tenth the price of commercially available scanners even when the curvature problem is yet to be addressed by these systems;
- The computational requirements are within practical constraints with no wait time except for the first two pages which would require:
 - A mere 11.5 seconds for the corrections of all distortions (image acquisition, curvature and barrel distortions corrections)
 - An additional 14 seconds for the OCR to initiate a read-out of two pages. Through this stage of the process the reading accuracy rate averages 91.33%. (refer to Table 7.1)
 - An additional 26 seconds will allow for spellchecking and error correction resulting in an average reading accuracy rate of 97.08% (refer to Table 7.1)
- It is unfortunate to note that the most popular and effective OCRs still make several errors when there is a mixed mode of fonts (two or more different fonts used in a given page).

Beyond these very important points, it should be noted that a human listener could forgo the step of spellchecking and error correction, as a human is able to overcome such a small percentage of errors as they get used to the system. Furthermore, within these 51.5 seconds in total waiting time for reading these first two pages, a user could turn to the next two pages to be read as soon as the first 11.5 seconds, and the reading of the next two pages is pipelined for the reading process. However it remains a point of contention for future research to address the issue of mixed fonts.

7.2 Prospective Research Endeavors

New attempts at improving the efficiency of this automated book reader should focus first and foremost on the full implementation of the segmentation process to determine the layout of text with respect to other graphics involving mathematical formulas, tables, figures, images and the like. We view this first step as a major undertaking, one that is regarded serve as an ultimate contribution to the work presented in this dissertation. In the way the current design is implemented, the read out will ignore anything that is not text. The groundwork has already been established for the segmentation process itself, it is left for other researchers to take these segmented blocks containing such things as formulas and other graphics and to initiate appropriate recognition algorithms (just like OCR for text).

Table 7.1: Comparison of ABBYY recognition rate of original and corrected images

#	Curve angle Degree	Word count	errors of original using WinOCR	RR of original using WinOCR	errors of corrected using WinOCR without spellchecking	RR of corrected using WinOCR without spellchecking	errors of corrected image using WinOCR after spellchecking	RR of corrected using WinOCR after spell checking
1	7.1	464	223	48.06%	45	90.30%	12	97.41%
2	7.4	383	94	24.54%	61	84.07%	22	94.26%
3	7.9	944	400	42.37%	53	94.39%	27	97.14%
4	8.3	963	357	37.07%	82	91.48%	18	98.13%
5	9.0	496	224	45.16%	18	96.37%	4	99.19%
6	9.7	456	192	42.11%	47	89.69%	15	96.71%
7	9.7	513	254	49.51%	51	90.06%	19	96.30%
8	11.9	898	357	39.76%	77	91.43%	20	97.77%
9	12.9	567	320	56.44%	57	89.95%	17	97.00%
10	13.7	898	480	53.45%	40	95.55%	28	96.88%
<i>Avg</i>	<i>9.76</i>	<i>658</i>	<i>290</i>	<i>43.84%</i>	<i>53</i>	<i>91.33%</i>	<i>18</i>	<i>97.08%</i>

It is most realistic to assume that a first step along this research endeavor is to devise an algorithm that would recognize all the symbols and elements already embedded in the Equation Editor of Microsoft. In so doing, one is impelled to realize that the use of Braille at all levels of our educational system is not as prevalent as one would presume. Then we are confronted with the fact that even when the blind person is assumed to know Braille that does not guarantee that the sighted math teacher does also know Braille, making the communication link even more exasperating. With an automated system that can provide an audio read out of these math expressions, there is potential to overcome this dilemma.

Now the processing time for all the procedures is 11 seconds totally, which is competitive to the most products in the market. In the time processing requirements distribution graph shown in the chapter 5, we found that almost 50% of the processing consumed time is spent on the procedure of image acquisition. The most time is wasted on opening and closing the camera, and switching between the overhead and lateral camera because the Canon SDK 7.1 can just support one camera at a time during image acquisition. Multiple cameras will be supported for the advanced Canon SDK so that these cameras can be kept open all the time and it will take no time to the camera capture operation.

The other crucial issue must be concerned is the image quality. Like ATIZ's BookDrive DIY applying two DSLR cameras to capture two pages of the book respectively, we can replace the overhead camera with the high quality commercial

camera, such as Canon DSLR, to obtain images with better resolution. This also involves the more complicated control software embedded with Canon EDS SDK or RCS SDK.

The layout information is lost in OCR module which integrates Microsoft MODI DLL. Layout, however, is a big concern for document image digitization, although it is not for book reader which cares only about the text part to correct speech for blind people.

Portability and compatibility are also matters to be considered when this prototype is put to be commercialized. The lateral camera and the overhead camera can be designed to fold together in the book holder so that it can be easier to carry. Currently only Canon releases the SDK for developers to implement software to control Canon's camera. The further research on compatibility is the goal to support any Canon's camera so that user can easily install his or her own Canon's cameras for image acquisition.

7.3 Summary

Given the complexity of the problem, the results obtained are highly precise for the text readout with a 97% average accuracy across the different pages that were read. These experiments included glossy materials as well, where the same high accuracy was achieved since the camera flash, which is the main cause of specular reflections in glossy materials, was not used nor was it necessary in the proposed system design.

In terms of computational requirements, the entire process of image restoration including page curvature correction and barrel distortion correction took a mere 6.1

seconds. Image acquisition consumed 47% of the processing time or 5.4 seconds. Consequently a read out of a page can begin 11.5 seconds from the time a book is appropriately placed onto the book holder. It is important to note that once the book is open, and image acquisition of the first two pages is accomplished a user can flip the book to the next two pages as the steps of distortion correction are being executed for the previous two pages. In this way once the reading of the book begins it would be a non-stop process with zero wait time for subsequent pages as the time to read two pages is much greater than the time it takes to correct the distortion.

On the economic front, the proposed design requires the purchase of two additional inexpensive cameras anywhere between \$300 to \$450, and the framing wood structure that holds the cameras and the book holder together, which cost us less than \$10 to construct, with the assumption that a computer is already available for use. This price certainly is most desirable in contrast to the sophisticated scanning devices such as the ATIZ book reader that is priced at \$35,000.

It is noted that the focus at this time is placed on text only, which is a very challenging problem in itself. Figures, Images and Equations are not yet resolved at this stage under the current design. We have begun working on mathematical expressions which involve several symbols and other special characters. The approaches we are considering is based on the fact that equations have special formatting and as such new tables will be used for the recognition of these specific formatting issues and consequently recognize the symbols and special characters associated with them. Once this task of identifying equations is resolved; the second step is in recognizing these

equations for what they are and read them out to the user. Such a process is very important as a new platform for teaching mathematics to blind persons.

This new prospect of extracting math equations and reading them to the blind person may facilitate the notable intent pursued by such programs as MAVIS (Mathematics Accessible to Visually Impaired), TIGER, DOTS plus, MathPlus Toolbox, Graphical Calculus in universities, such as New Mexico State, Oregon State, and Baruch College [Gardner, 2006; Edwards et al. 2006; Karshmer et al., 1998, 2001, 2002, 2003, 2004; and Archambault, 2004 and 2006, Clark, 2003; Stanley and Karshmer, 2006]]. These programs can as well serve as test-beds for complementary and/or comparative work for this worthy research endeavor.

LIST OF REFERENCES

- 1 Adjouadi M. and Candocia F., "A Stereo Matching Paradigm Based on the Walsh Transformation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 12, pp. 1212-1218, Dec. 1994.
- 2 Adjouadi M., Candocia F., Riley J., "Exploiting Walsh-based attributes to stereo vision", IEEE Transactions on Signal Processing 44 (2), pp. 409-420, Feb. 1996.
- 3 Adjouadi M., Ruiz E., and Wang L., "Automated Document Reader for Visually Impaired Persons", Springer's Lecture Notes on Computer Science (LNCS) series, K. Miesenberger et al. (Eds.): LNCS 4061, pp. 1094 – 1101, 2006.. Also presented at the 10th International Conference on Computers Helping People with Special Needs, July 12-14, 2006, Linz, Austria
- 4 Agam G. and Wu C. H., "Structural rectification of non-planar document images: application to graphics recognition", Fourth International Workshop on Graphics Recognition Algorithm and Applications, Kingston, Ontario, Canada, pp. 289-298, 2001.
- 5 Archambault D. and Moço V.. "Canonical mathml to simplify conversion of mathml to braille mathematical notations", K. Miesenberger, J. Klaus, W. Zagler, and A. Karshmer, editors, Proc. ICCHP 2006 (10th International Conference on Computers Helping People with Special Needs), volume 4061 of LNCS, pages 1191--1198, Linz, Austria, July 2006. Springer.
- 6 Archambault D., Fitzpatrick D., Gupta G., Karshmer A., Miesenberger K., and Pontelli E., "Towards a universal maths conversion library", Proc. ICCHP 2004 (9th International Conference on Computers Helping People with Special Needs), volume 3118 of LNCS, pages 664-669, Berlin, July 2004. Springer.
- 7 ATIZ, http://www.atiz.com/bookdrive_home.php.
- 8 Batušić M., Miesenberger K., and Stöger B., "Labrador, a contribution to making mathematics accessible for the blind", Proc. ICCHP 98 (6th International Conference on Computers Helping People with Special Needs), Oldenbourg, Wien, München, 1998.
- 9 Brown M. S. and Seales W. B., "Document restoration using 3D shape: a general deskewing algorithm for arbitrarily warped documents", International Conference on Computer Vision, vol.2, Vancouver, Canada, pp. 367-374, July 2001.
- 10 Clark P. and Mirmehdi M., "Rectifying perspective views of text in 3D scenes using vanishing points", Pattern Recog 36(11):2673-2686, 2003.

- 11 Coyle K., "Managing Technology: Mass Digitization of Books", *The Journal of Academic Librarianship*, Vol. 32 (6), pp. 641-645, Nov. 2006.
- 12 Doerman D., Liang J. and Li H., "Progress in camera-based document image analysis", *Proceeding International Conference of Document Analysis and Recognition*, pp. 606-616, 2003.
- 13 Dutoit T., "An Introduction to Text-to-Speech Synthesis", Springer, ISBN 0-7923-4498-7, 2001
- 14 Edwards A. D. N., McCartney H. and Fogarolo F., "Lambda: A multimodal approach to making mathematics accessible to blind students", *Proceedings of Assets*, Portland, Oregon, ACM, pp.48-54, 2006.
- 15 Gardner J. A., Ungier L. and Boyer J. J., "Braille Math Made Easy with the Tiger Formatter", *Springer's Lecture Notes on Computer Science (LNCS) series*, K. Miesenberger et al. (Eds.): LNCS 4061, pp. 926 –934, 2006.
- 16 Hafner K., "History, Digitized (and Abridged)", *The New York Times*, March 11, 2007.
- 17 Hartley R. and Zisserman A., "Multiple View Geometry in computer vision", Cambridge University press, Second Version, 2003.
- 18 Karshmer A. and Gillan D., "How well can we read equations to blind mathematics students: some answers from psychology", Stephanidis C., editor, *Universal Access in HCI - Inclusive Design in the Information Society*, volume 4, pp. 1290-1294, Mahwah, New Jersey, USA, 2003.
- 19 Karshmer A., Gupta G., Geiger S., and Weaver C., "Reading and writing mathematics: The MAVIS project", *Behavior and Information Technology*, 18(1):2--10, 1998.
- 20 Karshmer A., Bledsoe C., and Stanley P., "The Architecture of a Comprehensive Equation Browser for the Print Impaired", *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, Volume 3118/2004, *Computers Helping People with Special Needs*, pp. 614-619, 2004.
- 21 Karshmer A. I., Gupta G., and Gillan D., "Architecting an Audio Browser for Navigating Mathematical Equations", *Proceedings of the ICCHP 2002 Conference*, Linz, Austria, July 17, 2002.
- 22 Karshmer A. I., Gupta G., Pontelli E., Miesenberger, K., and H-F Guo, "The Development of a Tool to Enhance Communications between Blind and Sighted Mathematicians, Students and Teachers: A Global Translation Appliance," *Proceedings of the HCI Conference*, New Orleans, 2001.

- 23 Landy M. S., Maloney L. T, Johnston E. B. and Young M., "Measurement and Modeling of Depth Cue Combination - in Defense of Weak Fusion", *Vision Research* 35 (3), pp. 389-412, Feb 1995.
- 24 Liang J., DeMenthon D. and Doermann D., "Flattening curved documents in images", *International Conference on Computer Vision and Pattern Recognition*, San Diego, USA, pp. 338-345, June, 2005.
- 25 Lu S. and Tan C. L., "The Restoration of Camera Documents Through Image Segmentation", *LNCS* 3872, pp. 484-495, 2006.
- 26 Markoff J., "Media, Digitized", *The New York Times*, May 12, 2003.
- 27 Marr D. and Poggio T., "Computational Theory of Human Stereo Vision", *Proceedings of the Royal Society of London Series B-Biological Sciences* 204 (1156), pp. 301-328, 1979.
- 28 Moco V. and Archambault D, "Automatic conversions of mathematical braille: A survey of main difficulties in different languages", *Lecture Notes in Computer Science* 3118: 638-643 2004.
- 29 Pavic D., Schonefeld V. and Kobbelt L., "Interactive image completion with perspective correction", *Visual Comput*, DOI 10.1007/s00371-006-0050-2, 2006.
- 30 Pilu M., "Extraction of illusory linear clues in perspectively skewed documents", *CVPR*, December 2001.
- 31 Pilu M., "Perspective deskewing of documents from linear clues", *HP Labs Technical Report HPL-2001-6*, January 2001.
- 32 Pilu M., "Document capture", *US Patent Application US20020149808 A1*.
- 33 Pilu M. and Pollard S, "A light-weight text image processing method for handheld embedded cameras", *British Machine Vision Conference*, September 2002.
- 34 Pilu M., "Undoing Paper Curl Distortion Using Applicable Surfaces", *International Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, USA, pp. 67-72, 2001.
- 35 Pollard S. and Pilu M., "Building cameras to capture documents", *International Journal of Document Analysis and Recognition*, Vol. 7 (2), pp. 123-137, July, 2005.
- 36 Pollard S., Pilu M., Hayes S., Lorusso A., "View synthesis by edge matching and transfer", *Image and Vision Computing*, No. 9, pp. 749-75, June 2000.

- 37 Stanley P. B. and Karshmer A., "Translating mathml into nemeth braille code", K. Miesenberger, J. Klaus, W. Zagler, and A. Karshmer, editors, Proc. ICCHP 2006 (10th International Conference on Computers Helping People with Special Needs), volume 4061 of LNCS, pages 1175--1182, Linz, Austria, July 2006.
- 38 Strouthopoulos C. and Papamarkos N., "Text identification for document image analysis using a neural network", Image and Vision Computing 16, pp. 879-896, 1998.
- 39 Umer M. F. and Khiyal M. S. H., "Classification of Textual Document Using Learning Vector Quantization", Information Technology Journal 6 (1), pp. 154-159, 2007.
- 40 Vass G. and Perlaki T., "Applying and Removing Lens Distortion in Post Production", Colorfront Ltd., Budapest, 2003.
- 41 Wikipedia, http://en.wikipedia.org/wiki/Speech_synthesis.
- 42 Wong K. Y., Casey R. G., and Wahl F. M., "Document Analysis System", IBM J.RES. DEVELOP VOL 26, No.6, November 1982.
- 43 Woods R. E., "Digital Image Processing", Prentice-Hall, ISBN 0-201-18075-8, 2002.

VITA

LU WANG

Born, Hefei, Anhui, China

- | | |
|-------------|-------------------------------------------------------------------------------------------------------------------------------|
| 1997 - 2001 | B.S., Electrical Engineering
Zhejiang University
Hangzhou, Zhejiang, China |
| 2001 - 2003 | Software Engineer
Alcatel Shanghai Bell Company
Shanghai, China |
| 2003 - 2005 | M.S., Computer Science
Florida International University
Miami, FL |
| 2005 - 2007 | Doctoral Candidate in Electrical Engineering
Florida International University
Miami, FL |
| | Research Assistant
Center for Advance Technology and Education (C.A.T.E.)
Florida International University
Miami, FL |

PUBLICATIONS AND PRESENTATIONS

1. M. Adjouadi, E. Ruiz, and L. Wang, "Automated Document Reader for Visually Impaired Persons", Springer's Lecture Notes on Computer Science (LNCS) series, K. Miesenberger et al. (Eds.): LNCS 4061, pp. 1094 – 1101, 2006.
2. Presented at the 10th International Conference on Computers Helping People with Special Needs, July 12-14, 2006, Linz, Austria
3. Patent Application with OSRA (Lu Wang and M. Adjouadi)
4. Journal publication is underway for Pattern Analysis and Machine Intelligence
5. Journal publication is underway for Journal of Visual Impairment and Blindness