A USER CENTERED DESIGN AND PROTOTYPE OF A MOBILE READING DEVICE FOR THE VISUALLY IMPAIRED

A dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

By

ROBERT B. KEEFER.
B.S., Lawrence Technological University, 1992
M.S., Wright State University, 2000
M.S.E., Wright State University, 2010

2011
Wright State University

Committee on Final Examination

Nikolaos G. Bourbakis, Ph.D.
Dissertation Director

Arthur Goshtasby, Ph.D.
Director, Computer Science and Engineering Ph.D. Program

Nikolaos G. Bourbakis, Ph.D.

Soon M. Chung, Ph.D.

Sundaram Narayanan, Ph. D.

Krishnaprasad Thirunarayan, Ph. D.
ABSTRACT

Keefer, Robert B. Ph.D., Department of Computer Science and Engineering, Wright State University, 2011. *A User Centered Design and Prototype of a Mobile Reading Device for the Visually Impaired.*

While mobile reading devices have been on the market and investigated by researchers in recent years, there is still work required to make these devices highly accessible to the visually impaired. A usability test with one such device revealed gaps in the current state of the art devices. These gaps focus mostly on the user interaction and his or her ability to quickly consume written reading material. In this dissertation a voice user interface (VUI) is presented that improves the ability of a blind user of a mobile reading device to interact with written material. The image processing techniques required to facilitate this interaction with a document image are also presented. Contributions of this research include a model of the VUI, which was validated by user tests of a prototype with visually impaired participants, a document image perspective correction technique, a document image dewarping technique, and a headline identification technique, among others. Three separate user tests with visually impaired participants were used to guide and validate the interaction research. The document image processing techniques, combined to form a document image processing pipeline, were tested on 25 document images. Comparative results from the user tests and processing of the document images are presented in this dissertation.
# TABLE OF CONTENTS

Chapter 1 Reading without Sight........................................................................................................... 1

1.1 Reading Task Model......................................................................................................................... 1

1.1.1 Perceptual Scanning.................................................................................................................... 2

1.1.2 Text Processing.......................................................................................................................... 3

1.1.3 Word Recognition....................................................................................................................... 4

1.1.4 Comprehension........................................................................................................................... 4

1.1.5 Implications for Design.............................................................................................................. 4

1.2 User Analysis..................................................................................................................................... 5

1.2.1 Seniors........................................................................................................................................ 5

1.2.3 Summary of Implications for Design........................................................................................ 10

1.3 Alternative Interfaces for Reading.................................................................................................. 10

1.3.1 Magnification Interfaces for Reading......................................................................................... 11

1.3.2 Tactile Interfaces for Reading.................................................................................................... 11

1.3.3 Aural Interfaces for Reading...................................................................................................... 13

1.4 Mobile Reading Devices............................................................................................................... 15

1.5 TYFLOS System Overview............................................................................................................ 17

1.5.1 Document Image Processing....................................................................................................... 19

1.5.2 Voice User Interface.................................................................................................................. 20

1.6 Research Contributions and Outline of the Dissertation.............................................................. 21

Chapter 2 A Survey of Document Image Processing.............................................................................. 24

2.1 Document Image Capture.............................................................................................................. 29

2.1.1 Framed Document Image Capture............................................................................................... 29

2.1.2 Illumination Issues...................................................................................................................... 30

2.1.3 Document Image Enhancement.................................................................................................. 31

2.2 Binarization Techniques................................................................................................................. 32

2.2.1 Global Thresholding Techniques............................................................................................... 33

2.2.2 Adaptive Thresholding Techniques............................................................................................ 34
<table>
<thead>
<tr>
<th>Chapter 3 Document Image Processing Pipeline</th>
<th>55</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Document Image Binarization</td>
<td>56</td>
</tr>
<tr>
<td>3.2 Document Image Enhancement</td>
<td>57</td>
</tr>
<tr>
<td>3.3 Perspective Correction</td>
<td>60</td>
</tr>
<tr>
<td>3.4 Page Curl Correction</td>
<td>63</td>
</tr>
<tr>
<td>3.5 Page Segmentation</td>
<td>64</td>
</tr>
<tr>
<td>3.6 Headline Identification</td>
<td>67</td>
</tr>
<tr>
<td>3.7 Segment Aggregation</td>
<td>69</td>
</tr>
<tr>
<td>3.8 Experimental Results</td>
<td>71</td>
</tr>
<tr>
<td>3.9 Conclusion</td>
<td>78</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 4 Creating an Interactive Document</th>
<th>79</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 OCR at a Distance</td>
<td>80</td>
</tr>
<tr>
<td>4.2 Conversion of an Aggregated Image to XML</td>
<td>83</td>
</tr>
<tr>
<td>4.3 Targeted Registration</td>
<td>86</td>
</tr>
<tr>
<td>4.4 Creation of a Composite XML Document</td>
<td>90</td>
</tr>
<tr>
<td>4.5 Conclusion</td>
<td>98</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 5 Design of a Voice User Interface</th>
<th>99</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 VUI Commands Evaluation</td>
<td>103</td>
</tr>
<tr>
<td>5.1.1 VUI Command Design</td>
<td>103</td>
</tr>
<tr>
<td>5.1.2 Evaluation of User Commands</td>
<td>105</td>
</tr>
<tr>
<td>5.1.3 Method of Evaluation</td>
<td>106</td>
</tr>
</tbody>
</table>
5.1.4 Evaluation of Results................................................................. 108

5.2 Articulation Evaluation................................................................. 110
5.2.1 System Command Design......................................................... 110
5.2.2 Evaluation of System Commands.............................................. 112
5.2.3 Method of Evaluation.............................................................. 114
5.2.4 Evaluation Results................................................................. 117

5.3 Conclusion.................................................................................. 119

Chapter 6 Modeling and Verification of a VUI................................. 120
6.1 Stochastic Petri-Nets................................................................. 120
6.2 User Commands SPN................................................................. 121
6.2.1 Evaluation of Model Design.................................................... 123
6.2.2 Method of Evaluation............................................................ 124
6.2.3 Evaluation Results................................................................. 125
6.3 Revised User Command.............................................................. 128
6.3.1 Method of Evaluation............................................................ 128
6.3.2 Evaluation Results................................................................. 129
6.4 Unified Grammar SPN............................................................... 132
6.5 Evaluation of Unified Grammar SPN............................................ 135
6.5.1 Participants............................................................................ 136
6.5.2 Measures.............................................................................. 136
6.5.3 Procedure............................................................................ 136
6.5.4 Evaluation Results................................................................. 138
6.6 Conclusion.................................................................................. 139

Chapter 7 Conclusion and Future Work......................................... 140
7.1 Conclusions............................................................................... 140
7.2 Future Work.............................................................................. 140
7.2.1 Document Image Processing.................................................. 140
7.2.2 User Interface....................................................................... 143
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3 Contributions</td>
<td>143</td>
</tr>
<tr>
<td>Appendix A - Personas</td>
<td>145</td>
</tr>
<tr>
<td>A.1 Student - Michelle Green</td>
<td>145</td>
</tr>
<tr>
<td>A.2 Senior - Richard Browning</td>
<td>146</td>
</tr>
<tr>
<td>Appendix B - Reading Alternatives Comparisons</td>
<td>147</td>
</tr>
<tr>
<td>Appendix C - Analysis of OCR Accuracy Results</td>
<td>150</td>
</tr>
<tr>
<td>Appendix D - T-Test Results of User Studies</td>
<td>151</td>
</tr>
<tr>
<td>References</td>
<td>153</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1.1 Model of Reading Task ................................................................. 2
Figure 1.2 Basic Braille Alphabet ................................................................. 12
Figure 1.3 TYFLOS Overview and Webcams Mounted in a Pair of Glasses .... 18
Figure 1.4 Human-Device System Flow Diagram ......................................... 18
Figure 1.5 Document Image Processing Pipeline ........................................... 19
Figure 2.1 Document Image Processing Pipeline ........................................... 25
Figure 2.2 Color Image Converted to Grey-Scale and Binarized ..................... 32
Figure 2.3 Document Image Before and After Perspective Correction ............... 37
Figure 2.4 Document Image Before and After Page Curl Correction ............... 43
Figure 3.1 Document Image Processing Pipeline ........................................... 55
Figure 3.2 Binarized Document Image using Average, Niblack’s Adaptive, and Otsu Binarization ................................. 57
Figure 3.3 Enhanced Image using Average, Niblack’s Adaptive, and Otsu Binarization Methods ......................... 59
Figure 3.4 Bounding Rectangles of Connected Components Before Correction ................................................ 60
Figure 3.5 Baseline Fitting using RANSAC ................................................... 61
Figure 3.6 Estimated Vertical Vanishing Points ............................................. 62
Figure 3.7 Bounding Rectangle of Connected Components after Correction .......... 62
Figure 3.8 Perspective Corrected Image using Average, Niblack’s Adaptive, and Otsu Binarization ......................... 63
Figure 3.9 Page Curl Corrected Image using Average, Niblack’s Adaptive, and Otsu Binarization ................. 63
Figure 3.10 Pyramidal Reduction by Linear Interpolation, Nearest Neighbor, and Revised Pyramidal ........................................ 65
Figure 3.11 Boundary Acceleration Example .............................................. 66
Figure 3.12 Method to Follow a Block Boundary .......................................... 66
Figure 3.13 Region Boundaries using Average, Niblack’s Adaptive, and Otsu Binarization Methods .......................... 67
Figure 3.14 Separated Region Boundaries .................................................. 67
Figure 3.15 Histogram of Font Sizes in a Document Image .............................. 68
Figure 3.16 Headline Identification using Average, Niblack’s Adaptive, and Otsu Binarization ......................... 69
Figure 3.17 Segment Aggregation using Average, Niblack’s Adaptive, and Otsu Binarization ......................... 70
Figure 3.18 Method to Frame Headline and Textual Regions ....................... 71
Figure 3.19 Document Image Correction Evaluation Steps ............................. 72
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.7</td>
<td>User Command Sequence Diagram</td>
<td>105</td>
</tr>
<tr>
<td>5.8</td>
<td>Location of Articles on the Printed Page</td>
<td>106</td>
</tr>
<tr>
<td>5.9</td>
<td>Updated VUI Command Grammar</td>
<td>110</td>
</tr>
<tr>
<td>5.10</td>
<td>System Command Grammar Network</td>
<td>110</td>
</tr>
<tr>
<td>5.11</td>
<td>Formal System Command Grammar</td>
<td>111</td>
</tr>
<tr>
<td>5.12</td>
<td>System Command Sequence Diagram</td>
<td>112</td>
</tr>
<tr>
<td>5.13</td>
<td>Participant using knfbReader to Read a Newspaper and a Textbook</td>
<td>116</td>
</tr>
<tr>
<td>5.14</td>
<td>Interaction Effect between Auditory Feedback and Degree of Page Curvature on Accuracy</td>
<td>118</td>
</tr>
<tr>
<td>5.15</td>
<td>Task Success Least Square Mean</td>
<td>119</td>
</tr>
<tr>
<td>6.1</td>
<td>Original Stochastic Petri-Net Model</td>
<td>122</td>
</tr>
<tr>
<td>6.2</td>
<td>Adjusted Stochastic Petri-Net Model</td>
<td>126</td>
</tr>
<tr>
<td>6.3</td>
<td>User Commands Stochastic Petri-Net Model</td>
<td>130</td>
</tr>
<tr>
<td>6.4</td>
<td>Unified Grammar Stochastic Petri-Net Model</td>
<td>133</td>
</tr>
<tr>
<td>6.5</td>
<td>Coverability Graph of Figure 6.4</td>
<td>134</td>
</tr>
<tr>
<td>6.6</td>
<td>Average Number of Tokens per Place in 10 Simulations</td>
<td>135</td>
</tr>
<tr>
<td>6.7</td>
<td>Participant Using TYFLOS to Read a Newspaper</td>
<td>137</td>
</tr>
<tr>
<td>D.1</td>
<td>Eight Sighted Person Study Results</td>
<td>151</td>
</tr>
<tr>
<td>D.2</td>
<td>Six Visually Impaired Person Study Results</td>
<td>152</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table 1.1 Blind Student use of Alternative Devices................................................................. 8
Table 1.2 Cost/Benefits of Tactile vs. Aural Alternatives for Reading........................................ 15
Table 1.3 Comparison of Systems Based on User Centered Design Control Qualities.................. 16
Table 1.4 Image Processing Support of User Centered Design Control Qualities......................... 20
Table 1.5 VUI Support of User Centered Design Control Qualities........................................... 21
Table 2.1 Comparative Aspects................................................................................................. 26
Table 2.2 Product Manager Scores for Aspects........................................................................... 27
Table 2.3 Software Developer Scores for Aspects....................................................................... 28
Table 2.4 User Scores for Aspects.............................................................................................. 28
Table 2.5 Aspect Weights.......................................................................................................... 29
Table 2.6 Document Image Capture Scores.................................................................................. 32
Table 2.7 Global Thresholding Scores......................................................................................... 34
Table 2.8 Adaptive Thresholding Scores...................................................................................... 36
Table 2.9 Two Dimensional Perspective Correction Scores.......................................................... 40
Table 2.10 Three Dimensional Perspective Correction Scores..................................................... 43
Table 2.11 Page Curl Correction Scores....................................................................................... 48
Table 2.12 Document Image Segmentation Scores..................................................................... 53
Table 3.1 Confusion Matrix Results for the Headline Classifier.................................................. 77
Table 3.2 Confusion Matrix Results for the Segmentation Aggregation Classifier......................... 78
Table 4.1 Average Metrics of Sighted Newspaper Readers........................................................... 81
Table 4.2 OCR Accuracy when Document is 20 cm From the Camera........................................ 82
Table 4.3 OCR Accuracy when Document is 30 cm From the Camera........................................ 82
Table 4.4 OCR Accuracy when Document is 40 cm From the Camera........................................ 82
Table 4.5 OCR Accuracy when Document is 50 cm From the Camera........................................ 82
Table 4.6 Example Similarity Measure Matrix for Figure 4.5......................................................... 88
Table 4.7 Example Similarity Measure Matrix for Figure 4.6......................................................... 88
Table 4.8 Confusion Matrix Results for the Similarity Measure..................................................... 90
Table 4.9 Targeted Registration Accuracy..................................................................................... 96
ACKNOWLEDGEMENT

Nothing in life is performed in a vacuum, and this dissertation is no exception. It is the result of the support and encouragement of a few people. First of all, this dissertation would not have been written without the support of my incredible wife, Julie. She has been a constant encouragement to me, and somehow single-handedly maintained a form of normality for myself and our kids as I have pursued this endeavor.

Secondly, I would like to thank Dr. Nikolaos Bourbakis, my advisor, for his continuous support and partnership in this research. Special thanks also to Dr. John Flach for his input and guidance on the user interface portion of my work. I would also like to thank the members of my dissertation committee for their review and comments of my work: Dr. Soon Chung, Dr. Krishnaprasad Thirunarayan, and Dr. Sundaram Narayanan.

Finally, I would also like to thank my colleagues at the Assistive Technology Research Center for their collaboration and insight at various stages of my work: Dr. Raghu Kannavara, Dr. Dimitrios Dakopoulos, Dr. Alex Karargyris, Dr. Alex Pantelopoulos, and Thanos Tsitsoulis.
To JC and JK.
Chapter 1 Reading without Sight

With the invention of the printing press in 1439, Johannes Gutenberg introduced a method for communicating ideas across time and space for those able read. It has been said that the printing press was the greatest invention of that millennium. Unfortunately, the product of this extraordinary invention is not readily available to the 10 million people in the United States who are blind or visually impaired. Of these 10 million people, approximately 1.3 million report legal blindness, 55,000 of those are children. Most of these people have the mental ability to acquire and process information; the information merely needs to be made accessible to them.

When the visual modality is so impaired that no useful information can be communicated through this pathway, alternatives must be found as a substitute. Much work has been done to understand how people read and comprehend written text, and various methods have been developed to assist the visually impaired in the reading task. Thus, in this introductory chapter the reading skill is described, followed by an analysis of visually impaired users of alternative reading devices. This analysis led to the identification of six foundational user qualities an alternative reading device must have in order to support the reading task.

This analysis is followed by a survey of modern alternative reading devices available to the blind. The survey highlights the fact that no commercially available product or research prototypes report the ability to support the six qualities identified in the user analysis. This chapter concludes with a description of TYFLOS, a prototype of a mobile alternative reading device for the visually impaired. TYFLOS supports five of the six identified user qualities. The groundwork performed to enable these five user qualities is the topic of this dissertation; the primary contributions of the work described here stem from this support.

1.1 Reading Task Model

In order to understand the important qualities of a reading device, it is important to understand a reading task model, and to note that the goal of reading is comprehension. Although readers go through many of the same processes as listeners, the primary difference between reading and listening comprehension is that a sighted reader controls the rate of input, and a reading task model must account for this difference. Ideas communicated in written form enable the person to absorb the ideas at his or her own pace. Thus, the exercise of this aspect of reading enables a person to contemplate new ideas, review and enjoy stories, and understand the world around them. By contrast, a listener is dependent on the pace of the
speaker. A person listening to a recording may fast-forward or rewind the recording to review material, but is still dependent upon the recording to set the pace of input.

Another contrast between reading and listening pertains to the way humans’ store and process information. According to Nye and Bliss [Nye and Bliss, 1970], visual information is spatially organized, and is represented in the human mind by the relationship of objects to each other in space (i.e. left, right, up, down, near, far). In contrast, Kirman showed that humans organize auditory information temporally [Kirman, 1973]. That is, auditory signals are related in time rather than space. Kirman went on to show that tactile information is both temporally and spatially organized. However, the tactile sensitivity to space cannot process the detail that is available through the visual system. For example, to recognize a coin by only touching the face, one must move the finger, thus gathering tactile and temporal information simultaneously.

Much work has been done to develop unique models of the reading task and comprehension process. While the details of the models vary, a common structural pattern is comprised of four basic components: perceptual scanning, text processing, word recognition, and comprehension. For a person with no visual impairments or learning disabilities, these steps are measured in the microsecond range, and therefore are virtually imperceptible. A person with a disability may take much longer.

\[\text{Figure 1.1 Model of Reading Task}\]

1.1.1 Perceptual Scanning

The eyes are the most active of all human sensory organs. Whereas the skin, ears, nose, and tongue passively wait for and accept input signals, the eyes are constantly moving as they scan the visual field. These movements of the eyes play an important role in reading.

In 1878 it was discovered that when reading the eye moves in a series of small jumps, called saccades, rather than sweeping across the page as was the understanding of the day. With each saccade the eye pauses for an average of 250 milliseconds and jumps again. This pause enables the eye to capture approximately nine letter spaces. This focused vision enables detailed visual information to be obtained through the fovea, a small central area of the retina that has the highest concentration of photoreceptors. Foveal vision
encompasses a visual angle of 1 to 2 degrees, where peripheral vision is generally considered to be 180 degrees for a healthy person.

According to Norton and Stark [Norton and Stark, 1971], during reading there are typically 2 or 3 saccades per second, and these occur so fast that they occupy only about 10% of the viewing time. A movement of 10 degrees lasts only 45 milliseconds and, during the movement vision seems to be impaired. It appears then, that the recognition of letters and words in the nonfoveal field, which frequently occurs in the reading process, must be partly attributed to something other than the physical stimulation of the retina. Researchers believe that the reader relies on his or her understanding of the theme of the text for word recognition in the nonfoveal field.

1.1.2 Text Processing

Once the characters and words have been scanned, the text must be processed in order for recognition and understanding to occur. An influential series of experiments (called the Down-the-Garden-Path Experiments) performed by Carpenter and Dahneman [Carpenter and Dahneman, 1981] demonstrate one theory of processing. In these experiments a reader is presented with a short story in which an operative word is changed in a manner that surprises the reader. This is called the priming effect.

The following story serves as an example of the experiments:

*The young man turned his back on the rock concert stage and looked across the resort lake. Tomorrow was the annual one-day fishing contest and fishermen would invade the place. Some of the best bass guitarists in the country would come to this spot.*

The reader of this paragraph is led down the garden path in the first two sentences. In the third sentence, the word *bass* is read with a short *a* sound meaning the fish, not the long *a* sound meaning the instrument. The word *guitarists* causes the reader to backtrack and reread a portion of the text because guitarists does not fit what they expected to read. This backtracking is called regression, and occurs 10 to 15 percent of the time for the average reader.

According to Carpenter and Dahneman’s model, short-term memory plays an active role in the processing of words as they are scanned. Short-term memory stores the context of the topic being read. Context includes structures that are constructed in short-term memory by combining the preceding information represented from the text with information retrieved form semantic memory. Context effects
have been found in many language tasks that involve isolated words, rather than text. This priming effect, demonstrated above, has attributed to automatic activation among associated concepts in semantic memory.

1.1.3 Word Recognition

Word recognition has been demonstrated to occur in the ventral occipital lobe of the cerebrum. Posner [Posner, 1989] performed a series of experiments using a visual priming of a word (i.e. doctor-doctor) followed by a task involving a similar word pair (i.e. doctor-nurse). The results of the study showed that the visual priming and recognition follow similar pathways through the brain, namely through the ventral occipital lobe.

1.1.4 Comprehension

The reading process culminates in the reader’s comprehension of the presented ideas. Studies of comprehension have yielded nearly as many models as researchers. One model developed by Just and Carpenter [Just and Carpenter, 1980] demonstrates that when a reader encounters a section of written text that demands greater processing of information, he may require longer pauses to process unfamiliar words. This suggests that greater processing loads occur when readers are confronted with uncommon words, integrating information from important clauses and making inferences at the ends of sentences. Comprehension is also dependent on the coherence of key concepts in a sentence. When a text is written with strong links among the underlying ideas, it has been demonstrated that comprehension increases.

1.1.5 Implications for Design

Based on a reading task model comprised of perceptual scanning, text processing, word recognition, and comprehension, three implications for the design of a reading system to support the visually impaired are deduced:

- **Regression** - While this term refers to a reader’s ability to easily reread a portion of text (which occurs 10 to 15% of the time for sighted readers), a device should provide the ability to easily navigate throughout the text - forward or backward in large and small segments.

- **Spatial Cues** - Sighted readers utilize many spatial and temporal cues to find specific sections within a document. The spatial dimension is removed from a user of an auditory device. Thus, a device must supplement this temporal information with a form of spatial information.

- **Find-ability** - Without the spatial cues, a device can provide other forms of navigation and audible cues for a user to find information within a document.
The next section presents the analysis of a blind user of a mobile alternative reading device. The three implications concluded from this section will be combined with implications from the user analysis to form the foundation of a user-centered design of a mobile reading device for the visually impaired.

1.2 User Analysis

According to a recent report by the Jernigan Institute (2009), fewer than 10 percent of the 1.3 million people who are legally blind in the United States are Braille readers. This is due to a number of factors including a lack of teachers and lack of access to Braille books. There are a number of high-tech devices that can be used by the blind community to read. Unfortunately, none of these provide a voice user interface and none provide a high level of interaction for the user.

The reading task model described in Section 1.1 considers the development of a mobile reading device from the task perspective. In this section the users themselves are analyzed as a group profile. By considering the demographics, estimates of frequency of use, analysis of operational environment, and the experiences of potential users, an understanding of the user and how he would interact with a mobile reading device, further implications for the design of a device can be concluded.

In order to develop an inclusive understanding of the targeted user, two user groups were identified and are described below: seniors and students. While this analysis can predict with some certainty the requirements of a device, these requirements may not be complete. This incompleteness was demonstrated by two users of the knfbReader Classic who discarded the device because the battery in the camera cannot be easily changed. Situations such as this are harder to identify through user group profile analysis. A persona representing each user group is presented in Appendix A.

1.2.1 Seniors

As people age, their sight deteriorates often leading to a disruption of lifestyle, including losing the ability to read printed documents. The issues related to sight deterioration due to aging are well understood and should be accounted for in the design of a mobile reading device for the visually impaired.

1.2.1.1 Seniors Demographics

Age related visual impairments affect many people in the United States. Thus, one population group that serves as potential users of a mobile reading device is the elderly. Several age-related considerations in the development of such a system include:
• Age-related macular degeneration (AMD) is the most common cause of blindness and vision impairment in Americans aged 60 and older. More than 1.6 million Americans over age 60 have advanced AMD. [Lighthouse, 2009]

• Cataracts are the leading cause of blindness in the world. Cataracts affect nearly 20.5 million Americans age 65 and older. [Lighthouse, 2009]

• Vision changes in the elderly often result in increased susceptibility to glare [Lighthouse, 2009] and decreased contrast acuity.

• Common neurological symptoms that affect the elderly include decreased cognitive ability such as impairment of memory, deterioration of mobility, decreased sensory input (visual, auditory), and autonomic nerve system imbalance. [Sung, 2009]

• Decreased muscle mass, bone density and lubrication of the joints cause stiffness of the joints, osteoporosis, fractures of the hip are common and bone/joint functional impairment. [Sung, 2009]

• Age related hearing impairment affects over 25% of people over the age of 65.

1.2.1.2 Seniors Environment

According to the 2003 U.S. census: 38% of households with people over 65 years old had computers in their homes. [US Census, 2003]

Seniors who are open to attempting to learn to use a mobile reading device are assumed to have a computer at home. If not, they must be comfortable using one at the local library, senior center, or at their children's home. Most of the seniors who use a computer at home are comfortable with the way they have the workstation, chair, and desk space set up. Seniors using a reading device at home may be distracted by phone calls, interruptions by a spouse or grandchildren, and background noise such as television or radio in the next room.

Seniors will also use the device in public places such as a local coffee shop, a restaurant, or bookstore. In these situations the user will desire to avoid drawing attention to himself as he uses the device. A different set of distractions would emerge including other people in close vicinity or the server in a restaurant. (Note that the proposed system is not currently designed to filter background noise, nor does it address the open “cocktail party” problem of speech systems.)
1.2.1.3 Seniors Estimated Frequency of Use

The use of a mobile reading device by seniors is expected to be for magazines or newspapers around the house, or menus at a restaurant. Thus, it would be used a few times per week, and could be used for a few minutes or up to an hour daily.

1.2.1.4 Implications for Design for Seniors

Upon evaluating the demographics, environment, and frequency of use that a senior would use a mobile reading device, eight design implications become apparent:

- **Clear audible commands** - Audible commands must be focused and indicate intent. Elderly users must be able to distinguish between the set of available commands based on both hearing the command and understanding it.

- **Selectable reader voices** – The voice used to read the text and issue commands must be selectable to accommodate users with reduction in hearing capability in selective decibel ranges.

- **Concise help and in-process instructions** – The audible instructions must be accessible at any point in the interaction of the device, and the instructions must be clear.

- **Adjustable volume** – Volume of the device must be adjustable to accommodate users in various environments and hearing capabilities.

- **Low dexterity usage** – The tactile interface to the device must not depend on fine motor movements as elderly users may have reduced motor movement.

- **Audio feedback** – The user will listen to the device read the page to them rather than depend on a Braille display or other form of tactile feedback.

- **Audio input** – The user will interact with the device with voice commands rather then depend on learning to use a tactile interface.

- **Battery life** – The battery for the device should support the expected duration of use.

1.2.2 Students

Students are another population group that has a high dependency on reading printed material. Course notes, books, and presentation materials are all primarily available through a printed or hand written document. Thus, the issues related to school attendance must be included in the design of a mobile reading device for the visually impaired.
1.2.2.1 Student Demographics

People who are born with significant visual impairments or develop them over time require alternative methods for reading and performing education related tasks. Several considerations of this population in the development of a mobile reading device include:

- More than 20 million Americans report experiencing significant vision loss. The term vision loss refers to individuals who reported that they have trouble seeing, even when wearing glasses or contact lenses, as well as to individuals who reported that they are blind or unable to see at all. [NHIS, 2006]
- In 1994, approximately 1.3 million Americans were legally blind. [NHIS, 1995]
- There are approximately 57,696 legally blind children (aged 0 – 21) in the U.S. [APH, 2007]
- Approximately 5,626 legally blind children use Braille as their primary reading medium. Of the 57,696 children who are legally blind, 10% (5,626) are registered with the American Printing House for the Blind as Braille readers, 27% (15,303) as visual readers, 7% (3,942) as auditory readers, 34% (19,793) as non-readers, and 23% (13,032) as pre-readers. [APH, 2007]

1.2.2.2 Student Environment

According to the 2003 U.S. census, over 70% of households with the homeowner under the age of 65 had at least one computer. [US Census, 2003]

This leads us to assume that most younger people have had access to a computer much of their life and rely on it heavily for their education. All of blind students in a recent survey reported the use of screen reading software on a computer for reading (see Table 1.1). Thus, it is safe to assume that most students would not be averse to interacting with a mobile device.

<table>
<thead>
<tr>
<th>Method</th>
<th>User Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer - Note Taking</td>
<td>5/7</td>
</tr>
<tr>
<td>Computer - Screen Reader</td>
<td>6/7</td>
</tr>
<tr>
<td>Computer - Scanner</td>
<td>6/7</td>
</tr>
<tr>
<td>Printed Braille</td>
<td>6/7</td>
</tr>
</tbody>
</table>

*Table 1.1 Blind Student Uses of Alternative Devices*

In a recent survey, seven college students, with ages from 19 to 52, were questioned concerning their reading habits. Four were males and three were females, with an average age of 30. Each participant’s visual impairment was significant enough to require him or her to use alternative devices to access reading.
material. Six of the seven participants used a long-cane for mobile navigation. The seventh participant does not use any navigational aid. The participants' self-reported reading habits are as below:

- **Note taking device:** 5 participants had used a laptop computer, 2 had used a BrailleNote from HumanWare [HumanWare, 2010], and 1 participant had not used any note taking devices.

- **Printed Braille:** 6 participants had used printed Braille when textbooks or handouts could be electronically translated, and 1 had not used Braille.

- **Screen reader:** 6 participants had used Jaws from Freedom Scientific [FreedomScientific, 2010], whereas 1 had used ZoomText from AI Squared [AI Squared, 2010]

- **Scanning/reading:** 4 participants had used OpenBook also available from Freedom Scientific, 2 had used Kurzweil 1000 from Kurzweil Educational Systems [Kurzweil, 2010] and 1 had not used a scanner.

1.2.2.3 **Student Estimated Frequency of Use**

The use of a mobile device by students is expected to be significantly greater than that of seniors. For many students, over 50% of their reading is accessible through a computer, and thus a screen reader provides the access. However, much of the material required for reading must be translated into Braille or scanned into the computer for reading. It is this large amount of reading in which a mobile device may prove to be helpful.

Students will read textbooks, pamphlets, course notes and handouts from the instructor, as well as other books and articles throughout their education. It is estimated that students will spend 2 hours per day or more reading.

1.2.2.4 **Implications for Design for Students**

Upon evaluating the demographics, environment, and frequency of use that a visually impaired student would use a mobile reading device, five design implications become apparent:

- **Aesthetically pleasing** - People with disabilities often abandon a device if it does not meet user’s aesthetic preferences. [Kintsch and DePaula, 2002] Thus, the device must not draw attention to itself or the user.

- **Easily charge the battery** - People with disabilities often abandon electronic devices due to the difficulty of keeping the batteries charged. Thus, the device must provide a simple way to charge the battery.
• **Audio feedback** – The user will listen to the device read the page to them rather than depend on a Braille display or other form of tactile feedback.

• **Audio input** – The user will interact with the device with voice commands rather than depend on learning to use a tactile interface.

• **Tactile input** – The user will interact with the device with tactile commands in order to accommodate situations in which audible commands are not practical or draw attention to the user. (classroom, concert, etc.)

• **Extended battery life** – The battery for the device should support the expected extended duration of use (up to 2 hours).

### 1.2.3 Summary of Implications for Design

Based on the user analysis several implications for the design of a mobile reading device were identified. A device that accommodates all of these implications is beyond the scope of this research. However, the qualities that are addressed in this research are the ability of the user to issue audible/tactile commands and receive audible feedback.

Alternatives for reading have been developed for many years. The next section contains a survey of magnification, tactile and aural reading devices.

### 1.3 Alternative Interfaces for Reading

Many devices have been developed in an attempt to support the visually impaired in the reading task. Devices as simple as a magnifying glass can help those with early forms of sight deterioration. Individuals who have been born without sight or develop blindness at an early age often learn to read using Braille. In recent years, computer based systems have been developed that will convert written text into audible words.

The two primary alternative modalities for reading without sight are tactile and aural. Tactile interfaces generally take on the form of Braille, while aural is produced by some form of text-to-speech processing either by a human or machine. This section presents many of the alternatives that have been developed to aid the visually impaired in the reading task.
1.3.1 Magnification Interfaces for Reading

While it is generally assumed that visual impairments lead to total blindness, many people with visual impairments do have a limited degree of vision. Thus, for early stages of degenerative eye diseases, alternatives for reading take on the form of magnification systems. Two such systems include magnification aids and closed circuit television.

Magnification aids are inexpensive solutions for those individuals who have some residual sight. They typically take the form of a convex lens mounted in a frame or handle, and produce a magnified image of the page to be read. Magnifiers come in many shapes and sizes. Two of note are the bar magnifier and magnifier glasses. A typical magnifying glass has a small viewing area and requires the user to move the lens over the area of interest. However, a bar magnifier is a long, narrow lens that is convenient for reading a page of text without side-to-side movement. Magnifier glasses are worn over the eyes as spectacles are, and magnify everything the person looks at. While these only require the movement of the head to read, they are less convenient when looking up because the user would have to take them off to see anything further away.

A more versatile, yet expensive and stationary solution for magnification is closed circuit television (CCTV). These systems have been available for many years, and are useful for many tasks including reading, crafts, puzzles, etc. With the camera mounted above the work surface, anything that can be placed on the work surface can be displayed on the CCTV with significant magnification. Thus, visually impaired people find many uses for a CCTV. Unfortunately, this device is also quite expensive ranging from $1800 to $4,000.

1.3.2 Tactile Interfaces for Reading

The most widely used tactile alternative for reading for persons with visual impairments is Braille. Braille was originally developed as a military code that could be used by soldiers to communicate after dark. In 1821, Charles Barbier introduced the twelve-dot code to eleven year-old Louis Braille (1809 - 1852). Louis had been blinded in one eye at age three when he accidentally stabbed himself in the eye with an awl in his father’s workshop. He lost sight in his other eye due to sympathetic ophthalmia, an inflammation due to unilateral severe eye injury.

Braille began to develop and refine the military code into a code for the visually impaired to be able to read. Over the next three years Braille devised a reading method based on a cell of six dots. The system of
embossed writing he invented with an awl became accepted throughout the world as the fundamental form of written communication for blind individuals, and it remains basically as he invented it (Figure 1.2). The few modifications to the original system attempt to add contractions representing groups of letters or whole words that appear frequently in a language. The use of contractions enables Braille to be read faster and reduces the size of Braille books, making them less cumbersome.

![Figure 1.2 Basic Braille Alphabet](image)

Each Braille character consists of a cell of either six or eight dots. Six dot cells work well for basic reading; however, to represent more complex characters, such as contractions or computer codes, two more dots are added. The additional two dots raise the available number of characters from 64 to 256.

Recall, that people read using saccadic eye movements, with each jump gathering a group of letters at a time. In contrast, people reading with Braille move their hand across the line of raised dots, obtaining both spatial and temporal information. This smooth movement enables fluent Braille readers to read 200 words per minute, which is a similar pace to that of a sighted eighth grader.

Sighted readers use spatial information to find specific passages within a text [Nye and Bliss, 1970]. Braille users have a similar advantage in that spatial information can be used when searching a document. So, though on average it takes about four months to learn basic Braille and up to two years to learn Braille with contractions, many visually impaired individuals find that they can access information more quickly and perform tasks that involve reading or writing more efficiently using Braille than by listening to a personal reader, dictating to a personal secretary, or using alternative technologies such as audio recordings or voice synthesis. Experienced users of Braille are often able to read or take notes in Braille much more quickly than they can with other methods.
The primary advantage of Braille is that the users can read at their own pace, search for headings, paragraphs, spellings, etc. However, Braille is also difficult to learn and the pages are large and cumbersome. For example, a standard Braille page is 11" x 11.5", and only has room for 25 lines of 43 Braille characters. Though, some versions of Braille (Grade 2 Braille, Grade 3 Braille) use contractions, the entire transcription of a book to Braille can become unwieldy and heavy to carry. Due to heavy storage requirements and the costs associated with their corresponding translations, a very limited number of books and articles are available in Braille. The primary disadvantage of using Braille is the bulk of the documents that are printed in Braille. This is due to the limitations on the thickness of the paper used to emboss the documents, and the size of a Braille cell that can be felt with a finger. Braille is generally three times larger than its print version.

Two noteworthy devices available to Braille users are the refreshable Braille display and the Braille embosser. A refreshable Braille display, or Braille terminal, is an electro-mechanical device for displaying Braille characters, usually by means of raising dots through holes in a flat surface. Typical models display 40 to 80 eight-dot cells. Braille displays work with screen reader programs, described below. Because of the complexity of producing a reliable display that will cope with daily wear and tear, these displays are expensive - $2,000 to $7,500.

A Braille embosser is an impact printer that renders text as Braille. Utilizing translation software, a print document can be embossed with relative ease, making Braille production much more efficient and cost-effective. Embossers come in all shapes and sizes, and are used by everyone from individual computer users to large corporations that produce books, magazines, and other widely distributed publications, requiring fast, high-volume embossing capabilities. Thus, an embosser can cost from $2,000 to $80,000, depending on the user.

**1.3.3 Aural Interfaces for Reading**

The oldest and most prevalent use of an electronic device to provide auditory substitution for persons with visual impairment is recorded material. With the use of modern technology, recorded material is available through many different media. Audio books are now available on CD-ROM, through iTunes, and other media. However, due to the need of human translators, the number of books and articles available in audio format is quite limited.
While high quality recordings of books and other material available aurally serve the user well, visually impaired users must wait for the material to be read and recorded before it is accessible. Ray Kurzweil first introduced an automatic reading machine in 1981. This device scanned a printed page and performed text-to-speech processing to produce an audible version of the text. Similar reading devices available on the market today capture the document image, adjust the textual portion of the image for skew correction, page curl, and other deformities, perform optical character recognition (OCR), and finally produce an audible version of the text. These devices take one of two forms: software systems and stand-alone hardware systems.

Software reader systems are installed and used on a personal computer. Thus, the user must have access to a computer and a flatbed scanner. These software systems provide a full range of features including the ability to save the processed image to a text file and integrate with email and other Microsoft Office programs. These systems cost $100 to $1,500. Table B.1 (Appendix B) provides a comparison of nine different packages available on the market.

Stand-alone hardware systems come out of the box ready to use. There is no need to install software or connect to a scanner. While these systems have a limited set of features, they have been developed for those looking for a simple way to access print material without requiring access to a computer. These systems cost $2,300 to $3,500. Table B.2 provides a comparison of eight different devices available on the market. Reminiscent of the old “luggable” computers, “portable” stand-alone reader systems are large and require the user to carry a heavy piece of equipment (12 lbs.) to the library, or wherever it is going to be used.

Unfortunately, both computer based systems and stand-alone systems require the user to carry reading material to the device (i.e. stationary readers). An improved user experience would include a truly mobile device such as a mobile phone or PDA. Such devices are available on the market and continue to be under development in research labs. Before investigating the details of these devices, consider Table 1.2, which compares the costs/benefits of Braille reading with aural reading. It is important to understand these trade-offs in the user-centered design of a mobile reader for the visually impaired. While there is no direct substitute for visual input, both tactile and aural modalities provide alternatives for reading. Tactile modalities present the spatial and temporal information, which align more closely with visual input than aural. Aural modalities are limited to temporal information only, but require little training. Thus, a reading device must attempt to provide the benefits of each modality.
<table>
<thead>
<tr>
<th>Tactile (Braille)</th>
<th>Aural</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs of learning Braille are the time required to learn the skill (4 to 24 months), and the financial costs of supporting devices.</td>
<td>Costs of using aural substitution for reading are in the lack of spatial information, the reliance on other people or software to read the information, and the loss of self-directed pace.</td>
</tr>
<tr>
<td>Benefits of learning Braille are the spatial information, the ability to receive input at a self-directed pace, and the ability to search for things without fast forward/rewind.</td>
<td>Benefits of aural substitution for reading are the lack of time to learn Braille, and the lack of financial costs of the equipment. (e.g. Audio books can be borrowed from a local library.)</td>
</tr>
</tbody>
</table>

Table 1.2 Costs/Benefits of Tactile vs. Aural Alternatives for Reading

1.4 Mobile Reading Devices

Limited mobile reading devices have recently become available on the market. However, for a mobile reading device to be commercially viable, it must be affordable, small, and easy to use. The currently available readers are small, however, they are not affordable ($1,000 to $3,500), do not perform the image correction available from desktop computer based systems, nor do they integrate with other communication systems as the desktop systems do (i.e. MS-Word, email, etc.). Thus, while a person is more apt to use a handheld device than a “portable” reader, the technology is still in early stages of development. Table B.3 (see Appendix B) provides a comparison of three different devices available on the market.

A number of researchers are working on the development of a mobile reading device for the blind. Two general approaches are described in the literature: 1.) server-side processing, and 2.) on-board processing. Dumitras, et al. [Dumitras, et al. 2008] and Neo, et al. [Neo, et al., 2007] describe systems in which the processing is performed on a server. The captured image is transferred via the Internet from the mobile device to a server for processing. The image is processed and text returned back to the device for the user to hear. Gaudissart, et al. [Gaudissart, et al. 2004], Keefer, et al. [Keefer, et al. 2009b] and Panchanathan, et al. [Panchanathan, et al., 2003] describe systems in which the processing is performed on the mobile device itself. These systems require image-processing algorithms that have been optimized for hand-held devices. Table 1.3 presents a comparison of these proposed systems and highlights their differences relative to the qualities that were identified in the discussion of the User Analysis section above.
While there are many mobile systems that have been prototyped to perform various tasks with document/text processing, these seven are representative of systems that focus on providing a mobile reading device for the visually impaired. Of the qualities identified above, all of these systems focus on providing an accessible and portable (mobile) experience for the user. Since five of these systems are prototypes it is difficult to evaluate them on cost. Thus, the systems are compared based on their ability to provide regression, find-ability, spatial cues, and the required input and output forms.

- **CMU** – Dumitras, et al. [Dumitras, et al., 2008] describe a mobile reader for the visually impaired that is built on a Nokia 6620. The phone is used to capture an image and send the image (via HTTP) to a server for further processing. The server performs OCR processing on the image and sends the extracted text back to the phone for processing by a speech-synthesis engine built into the phone. The user interacts with the device using the tactile interface built into the phone and simply reads the text found in the scene from top to bottom. Thus, there is no regression, find-ability, or spatial cues.

- **Singapore** – Neo, et al. [Neo, et al., 2007] also describe a mobile reader for the visually impaired that performs the image processing on the server. The authors of this system utilize advanced image processing based on recognition text correction (RTC) techniques, providing improved robustness of text images able to be processed as compared to the CMU technique. After the server processes the image, the extracted text is sent back to the device for processing by a text-to-speech module. This system does not address regression, find-ability, or spatial cues, and the user interaction is limited to the tactile interface of the device itself.

- **SYPOLE** – Gaudissart, et al. [Gaudissart, et al., 2004] describe a mobile reader for the visually impaired that performs the image processing on the device rather than sending it to a server. By considering the user in the design of this system, the authors present a novel tactile human-device
interface. While this addition improves the user experience, it does not address regression, find-ability, or spatial cues in the experience.

- **iCare** – Panchanathan, et al. [Panchanathan, et al., 2003] describe a mobile reader for the visually impaired that provides a head-mounted camera to improve the image capture experience of the user. In a subsequent paper [Krishna, et al., 2005] this team identifies the importance of regression and spatial cues, however this system does not address these qualities directly.

- **knfbReader** - knfb Reading Technology, Inc. [knfb Reading Technology, 2008] has three products on the market that perform image processing on the device itself. Two of these devices are implemented on Nokia phones, and one is implemented on a PDA. All of these devices provide limited feedback to the user on document image quality, but do not address regression, spatial cues, or find-ability directly.

- **Intel Reader** - Intel Corp., [Whitney, 2009] has released a commercial product that performs the image processing on the device itself. It does proved a limited amount of navigational aid by providing the ability to move forward or backward by pages. However, it does not address find-ability or spatial cues in any way.

- **TYFLOS** – Keefer, et al. [Keefer, et al., 2009b] describe a mobile reader for the visually impaired that performs the image processing on the device and enable a highly interactive interface for a user to understand the spatial layout of the page, find points of interest on a page, and navigate throughout the document using a robust voice user interface (VUI). Section 1.5 provides an overview of this system.

### 1.5 TYFLOS System Overview

The TYFLOS prototype has been designed to support a voice user interface, navigation, facial recognition, and reading for the visually impaired. The complete system consists of two small cameras mounted into a pair of glasses, a microphone and ear-speaker headset, a range sensor, a GPS device, an RFID Reader, and a 2-D vibration array connected to a portable computer.

The voice user-interface (VUI) of the TYFLOS prototype serves as the control mechanism for both the reader and the navigation. An overview of the hardware components of the TYFLOS prototype is illustrated in Figure 1.3. The microphone, speakers, camera and portable computer support both the reader and the navigator, and a vibration array support the navigation.
Figure 1.4 illustrates the human-device system flow of the reader functionality in TYFLOS. In this diagram the *situation* refers to the environment in which the device is operated. For example, the lighting will affect the quality of the captured document image, and background noise will affect the ability to hear the reading. The *information* in this diagram refers to the audible words that the user will process. Both the user and the system will track the position within the text in order to issue/respond to navigation commands. *Awareness* represents the concept that the user’s awareness of the content grows as the text is processed. This awareness will lead to understanding of both the content of the text and the state of the device. The content of the document or the performance of the device itself can produce surprise in the user.

The interaction between each of the states in this system flow diagram can be considered as consequence, intention/expectation, action, and error/surprise. Since the system will change behavior based on the command issued by the user, this is considered the consequence. When the user expects the articulated words to have meaning (correct language, slow enough to understand, etc.), or the user expects the system to respond to the issued commands, intention is communicated and expectations are set. Whether the user issues a command or continues listening as a response to the presented information, either
response is considered an action. Similarly, the system will respond to the voice command of the user, or will continue in the current state when no command is issued. When the system does not respond correctly, or the content of the text changes abruptly, the user will be surprised. Since this could be caused by an error condition, this is modeled as error/surprise.

When a user speaks a voice command into the microphone, TYFLOS will process the command and determine the appropriate response. For one command (i.e. “take picture”) TYFLOS will take a picture of the scene and perform image-processing tasks on the image. For the other commands TYFLOS will respond to the user’s command based on the current image in memory or the article of interest to the user. TYFLOS will articulate the requested text or issue a guiding command depending on the context. Based on what the user hears, he or she will continue to listen or will issue a new command. This response is based on whether the articulation meets the expectation.

1.5.1 Document Image Processing

While the emphasis of this chapter has been on the user interaction, no interaction can take place without the complex document image processing techniques that enable the interaction. Several image-processing techniques have been developed to support the overall structure of the TYFLOS prototype including several document image correction methods employed to rectify the distortions and produce the highest quality image possible for the OCR. The entire process is illustrated in Figure 1.5.

![Figure 1.5 Document Image Processing Pipeline](image)

The first step in this process is to clean and enhance the resolution of the image itself. Due to the low-resolution cameras employed by the TYFLOS prototype, a super resolution algorithm has been developed and employed to enhance the resolution and image quality of the document text.

Another defect in the captured document image may be due to the rotation of the document in 3-dimensions and curvature of the page. OCR techniques expect the text to be in linear rows, and thus this defect must be rectified prior to the OCR. A unique method has been developed to simultaneously rectify a document image for skew and page curl defects.

After the document image has been straightened, document layout analysis methods have been developed in order to identify and classify portions of the document image. This identification and
classification step plays an important role in the interaction with the user in order to identify section headings of a document, images on the page, article text, etc.

The headline classification method is followed by an aggregation method in which correlated portions of the page are grouped together. This aggregation method also supports the interaction of the user with the document. By grouping the article text with the section headings a user can request specific portions of the page to be read rather than listening to the whole document.

After an image has passed through this sequence of steps, a selected user command can be fulfilled and the requested text passed to the text-to-speech generator for translation to audible words. However, due to the low resolution of the images captured by the TYFLOS prototype, the user will need to capture an image more than once at varying distances away from the cameras in order to capture an image that can be read. Thus, when a user requests a specific article to be read, he is guided to move the document to the correct position so that the article is in view of the cameras. In order to accommodate this adjustment, an image registration technique has been developed that will track the location of various images relative to each other, and then combine several images to create a composite representation of the text in XML.

The complete details of this image-processing pipeline are described in Chapter 3 and Chapter 4. Different components of the document image-processing pipeline enable various user qualities identified above. Table 1.4 illustrates how the different components support these features. Note that this research does not include any improvements in support of the tactile input quality.

<table>
<thead>
<tr>
<th>Component</th>
<th>Regression</th>
<th>Spatial Cues</th>
<th>Find-ability</th>
<th>Audible Output</th>
<th>Audible Input</th>
<th>Tactile Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enhancement</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Skew/Curl Correction</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segmentation</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Headline Classification</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Framing</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Registration</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

*Table 1.4 Image Processing Support of User Centered Design Control Qualities*

### 1.5.2 Voice User Interface

While the system must provide a tactile interface for use in environments in which the user cannot issue voice commands, this interface is not ideal due to the fact that a mental model of a tactile interface is
not as familiar to the user as simply speaking commands. Thus, in this design we focus on the voice commands, and leave the tactile interface for future improvement.

As has been mentioned previously, the interaction with the TYFLOS prototype is through voice commands. For example, when a user issues the command “take picture” TYFLOS will capture a picture of the scene in front of the cameras. After having passed through the image-processing pipeline, the scene is examined for text. Once the headlines have been identified, the user can issue another command (“read headlines”) and only the portions of the image identified as headlines are processed by the speech synthesis components. As a result, the user will hear the headlines being read.

Note that the user must take an active role in ensuring the quality of the captured image. Because TYFLOS uses two inexpensive cameras, there is no focusing mechanism in the cameras. Thus, the system will work interactively with the user to capture an image. As mentioned above, if it is determined that the OCR could be improved through adjusting the distance from the camera to the document of interest, the TYFLOS prototype will prompt the user to do so.

A voice user interface (VUI) has been developed to accommodate the voice interaction with the TYFLOS prototype. The complete details of this interface and a grammar are described in Chapter 5 and Chapter 6. Different commands of the VUI enable various user qualities identified above. Table 1.5 illustrates how the different components support these features.

<table>
<thead>
<tr>
<th></th>
<th>Regression</th>
<th>Spatial Cues</th>
<th>Find-ability</th>
<th>Audible Output</th>
<th>Audible Input</th>
<th>Tactile Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Move Commands</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>User Read Commands</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>System Move Commands</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Reading</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

*Table 1.5 VUI Support of User Centered Design Control Qualities*

### 1.6 Research Contributions and Outline of the Dissertation

This chapter presents the primary motivation for research into the development of a mobile reading device for the blind. When the reading task is considered, three qualities of a supporting system emerge: regression, spatial cues, and find-ability. An analysis of two user profile groups that may use a reading device unveil several important qualities that should also be incorporated into a reading device including
audible commands and feedback. A survey of common alternatives for reading available to the blind, as well as investigation into mobile research prototypes, reveals that no commercial product or research prototype addresses five important qualities identified in the task and user analysis. The TYFLOS prototype has been developed to address these five qualities.

The significant contribution of this research is the design and prototyping of a mobile reading device based on user-centered design principles and an understanding of the reading task specifically. Document image processing techniques enable the interaction with the document. Thus, several document image-processing contributions were derived from this research including a document image enhancement method, a document image skew correction technique, a document image dewarping method, a document image segmentation technique, a headline classification method, and a targeted image registration technique. [Keefer, et al., 2009a, Keefer, et al., 2009b, Keefer, et al., 2009c, Keefer and Bourbakis, 2011a, Keefer, et al., 2011c] The voice user interface contributions derived from this research include the development and modeling of a grammar for interaction, and five studies in which users interacted with the voice interface in order to design and validate the model. [Bourbakis, et al., 2008, Keefer, et al., 2010, Keefer, et al., 2011b]

A key component in the reading of a document image is the image-processing pipeline that prepares a document image for optical character recognition (OCR), and ultimately for the text-to-speech synthesis. Thus, in Chapter 2 a survey of methods and techniques for processing a document image is presented. As the available methods were evaluated for the implementation of a mobile reading device, gaps in the document image processing literature were identified.

The identified gaps include techniques that are implementable on a mobile device: 1.) an image enhancement method to aid the OCR process, 2.) image perspective correction and dewarping techniques to aid in the segmentation and OCR processes, 3.) an image segmentation method to identify sections of a printed document, 4.) a headline identification method, and 5.) a framing technique to cluster related sections. Thus, Chapter 3 presents techniques to address each of these gaps. Test results from the prototypes of each of these methods are also presented.

Since a comprehensive image, with required detail, cannot be captured by a 640x480 pixel image, multiple images of a newspaper article must be captured and registered in order to perform proper text-to-speech synthesis of a requested article. Chapter 4 presents a targeted registration method that is used to
ensure that an image contains a user directed portion of the scene. This combined with standard regular expression matching, and a few conflict resolution rules, provides a method of creating a composite XML version of a requested article. Results of the registration process and the composite creation are also presented in this chapter.

As noted above, the design of the TYFLOS prototype followed a user-centered design process. This included user research during the design stage, as well as modeling and user verification during the implementation stage of the VUI. Chapter 5 presents the overall design of the VUI, including the formal grammar specification, and the user research used to influence the design. Chapter 6 presents a Stochastic Petri-Net model of the user interaction design and user research performed to validate and enhance the model. Chapter 6 concludes with results from a study of the system by visually impaired participants.

This dissertation is concluded in Chapter 7 with a presentation on the direction that this research could provide in the future. This direction includes scene analysis and translation of text for tourists or war fighters and speech-pattern based security models, among others.
Chapter 2 A Survey of Document Image Processing

Traditional document image analysis techniques play an important role in the processing of a document image in order translate the image into audible sounds. With the proliferation of high-powered computers, the availability of high quality and inexpensive digital cameras, and the increasing desire to process printed documents, the interest in document image analysis algorithms has grown. Five key steps comprise typical document image analysis: document image capture, binarization, page perspective correction in 3-dimensions, page curl correction, and page segmentation. The use of document image analysis is not only important to reading devices, but is also pertinent in the tasks of:

- optical character recognition, that is, the ability to translate an image into ASCII text
- document verification, that is, the ability to detect and validate watermarks and other properties of a document to verify its authenticity
- document restoration, that is, the ability to transform an image of an aging document into a useful facsimile of the original
- image indexing, that is, the ability to extract text from images for the purpose of automated query/retrieval of the document
- information extraction, that is, the ability to elicit data from images for the purpose of understanding

In its simplest form, document image analysis leads to systems that determine the structure or layout of a document image. In other words, a document can be broken into its component parts such as headlines, textual columns, images, etc. Further analysis may discover text within images. Document image analysis can be difficult due to:

- lighting distortion
- orientation of the page when the image was captured
- noise in the document image
- page curl caused by the bend in a book binding
- typographical constraints
- complex page layouts
- near real-time processing requirements

One can simplify the analysis problem by imposing constraints on the orientation of the page. For example, many analysis algorithms assume that the image was captured in a highly controlled environment
using bright lights, a mounted camera (or two), and a jig to ensure that the page has a horizontal orientation. Prior knowledge of the content or orientation of the page can also further simplify the problem.

A number of approaches for document image analysis have been proposed. These approaches are broadly categorized as bottom-up analysis, top-down analysis, and a hybrid approach. A top-down approach is followed in describing the issues that need to be addressed when one sets out to analyze a document image. A simplified process, as illustrated in figure 1, is to capture the document image, prepare the image for segmentation, and then perform the image segmentation process itself. Section 1 discusses methods to capture a document image, which influence the degree to which a document image must be processed. In Section 2, binarization techniques commonly used in document image analysis are discussed. Sections 3 and 4 discuss further document rectification algorithms that facilitate perspective correction in 3-dimensions and page curl correction. In Section 5, existing page segmentation methods are described and strengths and weaknesses identified.

![Image: Document Image Processing Pipeline](image)

**Figure 2.1: Document Image Processing Pipeline**

In a recent survey, Liang, et al., [Liang, et. al., 2005b] present a summary of text detection literature. In this summary, task types are differentiated between identification of caption text and scene text. They present test data and results for a number of different methods. However, in this chapter a comparative survey based on several aspects of a document image analysis system design is presented (see Table 2.1). These aspects were identified as important to the decision making process of selecting any algorithm or third-party tool to be incorporated into a production environment.

The maturity formula in Table 2.1 was created based on clustering similar positive and negative characteristics. For example, availability aspects such as a method that is readily available, prototyped, and incorporated into a product are related and thus their scores are multiplied together. Similarly, performance measures such as robustness, reliability, and speed are related.
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td><strong>Availability</strong> - The ability to obtain/implement the system based on the description of the method expressed in mathematical formula, pseudo-code, or compiled code. A higher score indicates that a satisfactory amount of information is presented in the description to replicate the system. For example, a system with a score of 10 will contain a clear description of the method and C-code that could be implemented; whereas a system with a score of 5 may only have a mathematical formula and short process description.</td>
</tr>
<tr>
<td>Co</td>
<td><strong>Cost</strong> - The amount of money needed to use and/or implement the system based on the description provided. This score reflects the cost of equipment as well as implementation complexity.</td>
</tr>
<tr>
<td>FI</td>
<td><strong>Further Improvements</strong> - The methodology has the potential for further enhancement. A higher score indicates that a methodology can be improved upon, whereas a system with a lower score is considered more mature and less likely to be improved upon.</td>
</tr>
<tr>
<td>MC</td>
<td><strong>Model Complexity</strong> - Complexity of model used in the methodology. For example a system utilizing a neural network or wavelet is considered more complex than one that uses a run length smoothing algorithm.</td>
</tr>
<tr>
<td>O</td>
<td><strong>Originality</strong> - The methodology is based on original algorithms and/or mathematical operations; or the synergistic combination of simple methods composing a new method. A method that is referenced in the literature as original is given a higher score than one that builds on another method.</td>
</tr>
<tr>
<td>P</td>
<td><strong>Prototype</strong> - The methodology has been successfully implemented at the experimental stage and produced desirable results. Scores for this aspect were also affected by the results presented. A paper that presented comparative results scored higher than one that presents an illustrative example.</td>
</tr>
<tr>
<td>RP</td>
<td><strong>Released Product</strong> - The methodology has been implemented in a commercial setting. This aspect has a value of either 1 or 3, where the few methods/systems that have been utilized in a commercial setting are given a slight advantage over others.</td>
</tr>
<tr>
<td>Re</td>
<td><strong>Reliability</strong> - The methodology produces expected results under normal operating conditions.</td>
</tr>
<tr>
<td>Ro</td>
<td><strong>Robust</strong> - The methodology produces acceptable results under extenuating circumstances. This score is based on the features of the methodology as compared to methodologies in a similar category. For example, in the Perspective Correction category, methods that account for both 2D (skew) and 3D (perspective) are considered more robust than those that account for only one or the other.</td>
</tr>
<tr>
<td>Sp</td>
<td><strong>Speed</strong> - Reported processing time for sample tests. Note that some authors do not report performance metrics. For these we give a score of 3 out of 10.</td>
</tr>
<tr>
<td>U</td>
<td><strong>Usability</strong> - The methodology offers a user-friendly interface so that the user can work easily with it. Systems that require no user input are given a higher score than those that require input parameters or training data.</td>
</tr>
<tr>
<td>ST</td>
<td><strong>Scan Technique</strong> - Document image capture method used with method: flatbed scanner (FS), digital camera (DC), custom configuration that often requires calibration (CC).</td>
</tr>
<tr>
<td>M</td>
<td><strong>Maturity</strong> - A measure that combines the scores of the different aspects. Maturity = U + O + ((A<em>P</em>RP) + (Re<em>Ro</em>Sp))/(Co<em>FI</em>MC)</td>
</tr>
</tbody>
</table>

**Table 2.1: Comparative Aspects**

Availability and performance are considered as positive aspects whereas cost, the need for further improvements, and complexity are considered negative aspects. Thus, dividing by the negative scores reduced the impact of the positive scores. In this way, an expensive and complex method that is fast and robust will have a lower maturity score than one that is less expensive and simpler but equally fast and robust.

In order to determine weighted scores for each aspect of a system, a series of surveys were administered to gather industry perspectives on the aspects. Ten product managers, ten software developers, and ten users with industry experience were surveyed. The surveys covered every aspect listed in Table 1.
except for the Further Improvements (FI) and the Scan Technique (ST) aspects, as these are not pertinent to industry decision makers in general.

Ten product managers from various industries were surveyed to obtain a real-world perspective on the importance of these aspects of system development. Industries represented in this survey include print publishing, electronic commerce, national defense, financial institutions, and electronic medical records. The products represented by this survey include web applications, desktop applications, and back-end processing applications. The product managers were asked to rank the importance placed on the various aspects of a system as it relates to either selecting a product to incorporate into a system or decide to release a product to market. The results of this survey are reported in Table 2.2.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>Co</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager 1</td>
<td>10</td>
<td>7</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>6</td>
<td>10</td>
<td>6</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>Manager 2</td>
<td>10</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Manager 3</td>
<td>7</td>
<td>4</td>
<td>8</td>
<td>4</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>7</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>Manager 4</td>
<td>8</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Manager 5</td>
<td>1</td>
<td>9</td>
<td>7</td>
<td>10</td>
<td>3</td>
<td>7</td>
<td>10</td>
<td>8</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Manager 6</td>
<td>6</td>
<td>6</td>
<td>9</td>
<td>4</td>
<td>3</td>
<td>8</td>
<td>10</td>
<td>9</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>Manager 7</td>
<td>7</td>
<td>7</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>7</td>
<td>10</td>
<td>7</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Manager 8</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>Manager 9</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>10</td>
<td>10</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>Manager 10</td>
<td>8</td>
<td>9</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>10</td>
<td>9</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>Average</td>
<td>7.2</td>
<td>7.2</td>
<td>5.9</td>
<td>4.3</td>
<td>3.4</td>
<td>7.0</td>
<td>9.3</td>
<td>8.1</td>
<td>8.6</td>
<td>9.8</td>
</tr>
</tbody>
</table>

*Table 2.2: Product Manager Scores for Aspects*

A similar survey was presented to ten professional software developers in order to discover weighted scores for each aspect of a system from their perspective. Industries represented by the software developers include electronic commerce, national defense, financial institutions, electronic medical records, and electronic publishing. All of the developers had a minimum of 10 years of industry experience, influence technical decisions in product development. The products represented by this survey include web applications, desktop applications, back-end processing applications, and mobile applications. These developers were asked to rank the importance placed on the various aspects of a system as it relates to either selecting a product to incorporate into a system or decide a method to use when constructing a product. The results of this survey are reported in Table 2.3.
A third survey was presented to ten software users in order to discover weighted scores for each aspect of a system from their perspective. Industries represented by this user community include electronic commerce, financial, health care, and electronic publishing. The products represented in this survey are web application, desktop applications, and mobile applications. These users were asked to rank the importance placed on the various aspects of a system as it relates to selecting a product to incorporate into their day-to-day work activities. The results of this survey are reported in Table 2.4.

### Table 2.3: Software Developer Scores for Aspects

<table>
<thead>
<tr>
<th>Developer</th>
<th>A</th>
<th>Co</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer 1</td>
<td>8</td>
<td>10</td>
<td>7</td>
<td>2</td>
<td>8</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>Developer 2</td>
<td>10</td>
<td>7</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>8</td>
<td>7</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>Developer 3</td>
<td>1</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>8</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Developer 4</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>9</td>
<td>7</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>Developer 5</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Developer 6</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>10</td>
<td>10</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>Developer 7</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>8</td>
<td>3</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>Developer 8</td>
<td>7</td>
<td>8</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>7</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Developer 9</td>
<td>8</td>
<td>1</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Developer 10</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>5.0</td>
<td>6.9</td>
<td>3.6</td>
<td>3.5</td>
<td>3.9</td>
<td>7.2</td>
<td>9.1</td>
<td>8.7</td>
<td>7.1</td>
<td>8.8</td>
</tr>
</tbody>
</table>

### Table 2.4: User Scores for Aspects

<table>
<thead>
<tr>
<th>User</th>
<th>A</th>
<th>Co</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>User 2</td>
<td>9</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>User 3</td>
<td>10</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>User 4</td>
<td>10</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>User 5</td>
<td>9</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>User 6</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>User 7</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>User 8</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>User 9</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>User 10</td>
<td>10</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>Average</td>
<td>9.8</td>
<td>9.7</td>
<td>1.5</td>
<td>1.3</td>
<td>1.6</td>
<td>2.1</td>
<td>9.8</td>
<td>9.7</td>
<td>9.1</td>
<td>8.9</td>
</tr>
</tbody>
</table>

A compilation of the three surveys provides a weight for each aspect that can then be used for comparison throughout the rest of this survey. For each system component or methodology presented in the sections below a score for each aspect of the proposed approach is presented, along with a score weighted by the perspective brought through the industry surveys. Each method was given a score based on the
degree to which it accounted for a given aspect: does not account for the aspect (1), somewhat accounts for the aspect (4), mostly accounts for the aspect (7), and fully accounts for the aspect (10). The aspect score is then multiplied by the aspect weight presented in Table 5. Initial scores were determined based on the consensus of a small group of colleagues.

<table>
<thead>
<tr>
<th>Aspect Weight</th>
<th>A</th>
<th>Co</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager Perspective</td>
<td>7.2</td>
<td>7.2</td>
<td>5.9</td>
<td>4.3</td>
<td>3.4</td>
<td>7.0</td>
<td>9.3</td>
<td>8.1</td>
<td>8.6</td>
<td>9.8</td>
</tr>
<tr>
<td>Developer Perspective</td>
<td>5.0</td>
<td>6.9</td>
<td>3.6</td>
<td>3.5</td>
<td>3.9</td>
<td>7.2</td>
<td>9.1</td>
<td>8.7</td>
<td>7.1</td>
<td>8.8</td>
</tr>
<tr>
<td>User Perspective</td>
<td>9.8</td>
<td>9.7</td>
<td>1.3</td>
<td>1.6</td>
<td>2.1</td>
<td>9.8</td>
<td>9.7</td>
<td>9.1</td>
<td>8.9</td>
<td>9.7</td>
</tr>
<tr>
<td>Aspect Weight</td>
<td>0.73</td>
<td>0.79</td>
<td>0.36</td>
<td>0.31</td>
<td>0.31</td>
<td>0.80</td>
<td>0.94</td>
<td>0.86</td>
<td>0.82</td>
<td>0.94</td>
</tr>
</tbody>
</table>

Table 2.5: Aspect Weights (average of perspectives divided by 10)

As a simple example, consider the robustness score for the perspective correction methods. A score of seven (7) may be given to a method that fully corrects for two dimensional skew correction, whereas a score of ten (10) would be given to a method that fully corrects for two dimensional skew and perspective distortion in three dimensions. The weight for robustness obtained from the industry surveys (0.86) is then multiplied with each score. The tables below contain two values: one is the aspect score we assigned and the other is the score multiplied by the industry weight.

2.1 Document Image Capture

To perform document image analysis, an image of a document must be captured. There are a variety of methods for doing so, including flatbed scanners, handheld digital cameras, and highly precise custom configurations. There are a number of issues to overcome in order to perform accurate analysis of the captured image including framing the image, proper illumination, and image distortion avoidance/rectification. Many researchers have developed methods for dealing with these issues.

2.1.1 Framed Document Image Capture

The document image of interest requires less processing if it is framed such that perspective skew and rotation are avoided. Organizations have developed different approaches to solving this problem. Three approaches presented here are the Parc BookScanner, developed at XEROX PARC, a Structured Light Technique, developed by Hewlett Packard, and the Copibook scanner, developed by I2S.

- **Parc BookScanner.** Swartz, et al., [Swartz, et al., 2000] developed a scanner specifically designed to scan rare and fragile books. The platen system, comprised of one camera and a mirror unit, holds the page surfaces such that each page is framed without perspective skew, rotation or page curl. This
enables the document images to be preserved or further analyzed without requiring image rectification processes.

- **Structured Light Techniques.** Pollard and Pilu [Pollard and Pilu, 2005] propose the use of structured light patterns to assist with proper framing. These structured light patterns are generated using laser light and a diffractive optical element (DOE). The DOE generates a structured light pattern that, when applied to a document before the image is captured, is used for framing the document. The properly framed document image avoids the need for image rectification processes.

- **CopiBook.** The I2S CopiBook scanner [I2S, 2008] employs physical guides to ensure proper placement of the document in the scanner. These guides enable a document to be framed such that perspective skew or rotation is nearly eliminated. However, no adjustment for page curl is provided by this one camera system, thus further document image processing techniques may be required to rectify the page curl.

### 2.1.2 Illumination Issues

The illumination of the captured document will greatly affect further analysis of the image. A flatbed scanner provides a consistently bright light, whereas a digital camera may use only ambient light. However, under both conditions specular reflections can prove to be problematic. Several techniques have been developed to eliminate or correct the specular reflections found within an image including a dual flash technique, a dual polarized light technique, and a plural angles technique.

- **Dual Flash Technique.** Pollard and Pilu [Pollard and Pilu, 2005] propose a dual-flash solution as a method for dealing with the specular reflections. This method proposes the capture of two images of the document using a fixed camera, and alters the position of the flash, which in turn adjusts the position the glare caused by the flash to different locations on the document. The two images are then merged into one image that has no glare.

- **Dual Polarized Light Technique.** Mudge, et al. [Mudge, et al., 2005] propose a method for removing specular reflection using double polarized light. A structured light projector was equipped with a linear polarizer. A second linear polarizer was placed over the camera lens and its angle rotated until all specular reflections were removed. When the calibration is complete a clean image can be captured.

- **Plural Angles Technique.** Barkan and Shepard [Barkan and Shepard, 2007] propose a system in which a collection of lights are arranged at different angles and directed at the document. Thus, the camera...
receives light reflected from the document at a plurality of angles. Specular reflection associated with one set of angles is minimized by light from a second set.

### 2.1.3 Document Image Enhancement

Devices used to capture document images continue to improve over time, and the ability to capture these images at a high resolution significantly improve the analysis. However, some document images continue to require resolution enhancement to increase the performance of other downstream processes. Two methods that have been developed to enhance the resolution of a document image are the convolution method and an example-based method.

- **Convolution Method.** Cameras embedded in small devices such as a mobile phone typically have low resolution. Pilu and Pollard [Pilu and Pollard, 2002] propose using a bicubic convolution method to upscale the captured image. The resulting image is passed through an unsharp mask filter in an attempt to produce a higher quality image. This process is effective in an embedded environment due to the few add-multiply operations required.

- **Example-based Method.** Datsenko and Elad [Datsenk and Elad, 2007] propose an image resolution method based on a database of examples. Using a nearest-neighbor approach, a low-resolution image is separated into portions. Each portion is then assigned several candidate high-quality portions of images from the examples. Candidates are then used to enhance the resolution of the image through a global MAP penalty function. This function is used to reject irrelevant examples and for reconstructing/enhancing the captured image.

Table 2.6 presents the maturity scores for the image capture techniques discussed above. Note that the framing, illumination and enhancement techniques could be combined to create an improved document image. While poor image quality and illumination issues cause problems for advanced image analysis techniques, the images can be distorted in a variety of ways including perspective skew, rotation, and page curl. The next few sections present methods for overcoming these issues.
2.2 Binarization Techniques

While the methods used to capture a document image play an important role in the quality of the image analysis, many modern settings are not ideal. These conditions may lead to distortions of the document image such as shadows, low contrast, and non-uniform illumination. Proper rectification of these distortions leads to greater quality in other document image analysis techniques. Thus, the ability to account for distortions in a document image, and separate text from the background of an image, have a significant impact on the overall quality of a system. In this section, common techniques used to separate text and from the background in a document image are described.

Binarization generally refers to the conversion of a grey-scale image into a binary image. If the source is a color image, it is first converted to a grey-scale image. A common conversion for this process is:

\[ \text{GreyValue} = (0.3)(\text{RedValue}) + (0.59)(\text{GreenValue}) + (0.11)(\text{BlueValue}) \]  

(2.1)

where the color values range from 0 to 255. Most document image analysis algorithms assume a grey-scale image, and thus it is best to apply this filter early in the process when needed. Figure 2.2 illustrates the effect binarization has on an image.
2.2.1 Global Thresholding Techniques

There are two general categories of binarization techniques: global and adaptive. The global techniques presented here are simple global thresholding, grey-level histogram, connectivity, and photometric correction. A few adaptive techniques are described below.

- **Global thresholding.** The global thresholding algorithm [Jain, et al., 1995] is quite simple. In this algorithm a fixed intensity threshold value $T$ is chosen by the user. The intensity value of each pixel in the image is then compared with $T$ to determine if it is part of text or background. Background pixels are set to white and pixels associated with text are set to black.

- **Maximum Entropy.** Otsu’s method [Otsu, 1979] separates the pixels of a normalized grey-scale document image into two classes: foreground and background. An initial intensity threshold is calculated based on the mean and variance of each class. Subsequent threshold values are calculated for each intensity level until one is found that minimizes the variance within a class. This method reduces the thresholding problem to a search problem (i.e. search for an optimized threshold value), which has more potential for identifying an optimal threshold than the global thresholding method.

- **Connectivity.** The method proposed by O’Gorman [O’Gorman, 1994] uses a global approach calculated from a measure of local connectivity information. The method follows three steps. In step one, horizontal and vertical connectivity values are tracked for global thresholding for each intensity level, thus creating a histogram of intensity values. In this case, the connectivity value is the number of consecutive pixels along a row or column. The histogram is then analyzed and the maximum peak values are identified. Finally, in step three, a set of thresholds is determined based on the peak values, thus preserving the connectivity of regions.

- **Photometric correction.** Lu and Tan [Lu and Tan, 2007] propose a global thresholding technique to rectify a document image that has been distorted by varying degrees of illumination. This adjustment is followed by a global thresholding technique. A two-dimensional Savitzky-Golay filter is used to estimate the shading variation of a document image. The filter employs a two-step surface fitting process on the document image: first, a rough estimate of the background is calculated, followed by the estimation of the global shading variation based on the background estimate. Finally, a uniformly illuminated document image can be generated from the estimated shading variation. The resulting document image is then binarized using a simple global thresholding technique.
It is difficult to determine a global threshold for poorly illuminated document images. The method proposed by Lu and Tan compensated for poor lighting by calculating a global two-dimensional Savitzky-Golay filter, which was then applied to the image to compensate for shaded distortions. While this method is faster than adaptive methods it assumes that the distortion caused by shading is smooth. Thus, due to the local calculations, an adaptive approach will produce better results when the captured image contains an abrupt change in illumination or background color.

The connectivity method proposed by O’Gorman may be considered a hybrid approach. This method determines a global threshold based on local pixel connections, and thus is not as susceptible to limitations of other global thresholding techniques. O’Gorman demonstrates that a threshold derived from connectivity preservation is superior to Otsu’s maximum entropy method.

The score for availability presented below reflects the quality of the presented work. O’Gorman presents the mathematical foundations of his method, with pseudo-code, and a C-like outline all of which make this method readily available. The information in his paper provides a clear guide to an implementation of the method.

<table>
<thead>
<tr>
<th>Method</th>
<th>A</th>
<th>Co</th>
<th>FI</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global</td>
<td>9/6.6</td>
<td>2/1.6</td>
<td>8</td>
<td>2/0.7</td>
<td>8/2.5</td>
<td>9/2.8</td>
<td>3/2.4</td>
<td>2/1.9</td>
<td>2/1.7</td>
<td>9/7.4</td>
<td>3/2.8</td>
<td>CC</td>
<td>20/12.8</td>
</tr>
<tr>
<td>Max. Entropy</td>
<td>7/5.1</td>
<td>3/2.4</td>
<td>3</td>
<td>3/1.1</td>
<td>8/2.5</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>3/2.8</td>
<td>3/2.6</td>
<td>3/2.5</td>
<td>7/6.6</td>
<td>CC</td>
<td>18/12.5</td>
</tr>
<tr>
<td>Connectivity</td>
<td>9/6.6</td>
<td>4/3.2</td>
<td>5</td>
<td>4/1.4</td>
<td>8/2.5</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>7/6.0</td>
<td>7/5.7</td>
<td>7/6.6</td>
<td>CC</td>
<td>19/16.8</td>
</tr>
<tr>
<td>Photometric</td>
<td>6/4.4</td>
<td>6/4.7</td>
<td>5</td>
<td>6/2.2</td>
<td>8/2.5</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>7/6.0</td>
<td>7/5.7</td>
<td>5/4.7</td>
<td>-</td>
<td>15/10.5</td>
</tr>
</tbody>
</table>

Table 2.7: Global Thresholding Scores

2.2.2 Adaptive Thresholding Techniques

Global thresholding techniques do not work well with poorly illuminated document images. The photometric correction method, in particular, is susceptible to sharp contrasts in illumination. Adaptive thresholding techniques have been developed to compensate for these distortions. These techniques include the Niblack adaptive thresholding technique, the Sauvola adaptive thresholding technique, and the Gatos adaptive thresholding technique.

- **Niblack adaptive thresholding.** Wayne Niblack developed the most common thresholding algorithm used in document image analysis [Niblack, 1986]. This algorithm can be understood by considering a rectangular window slid over a grey-scale image and calculating threshold values within the window. The local threshold values are calculated for each image pixel by using the intensity of the pixels
within a small neighborhood window. The threshold $T$ is computed using the mean $m$ and standard deviation $s$ of all the pixels in the window:

$$T = m + (k)(s)$$

(2.2)

where $k$ ($0 < k < 1$) is a user-defined constant that determines the thickness of the binarized stroke that is retained. This local threshold is then used to binarize the pixels within the window, and as the window is slid over the image, the complete document image is binarized. Clearly, the size of the sliding window and the value of $k$ affect the quality of the resulting image.

- Sauvola adaptive thresholding. The approach proposed by Sauvola and Pietikainen [Sauvola and Pietikainen, 2000] is a variation of the Niblack algorithm in that threshold values are calculated by sliding a rectangular window over a grey-scale image. However, the threshold value $T$ is computed in a slightly different manner using the mean $m$, standard deviation $s$, the user defined constant $k$ and the dynamic range of the standard deviation value $R$:

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

(2.3)

Similar to the Niblack method, the complete document image is binarized as the window is slid over the image. Also, as with the Niblack algorithm, the size of the window and the value of the user defined parameter $k$ affect the resulting image. An added dependency with this approach is the value of $R$, which is also user-defined, and is based on knowledge of the contrast of the document image.

- Gatos adaptive thresholding. The method proposed by Gatos, et al. [Gatos, et al. 2006] build on Sauvola’s work, and follow a multi-step process. First, a Wiener filter is used to reduce the noise in the document image. The location of textual regions in the document image is then estimated. Based on this estimate, the background surface of the image is approximated. The fourth step of this method determines the final textual regions by employing a distance-difference technique on the estimates of the textual region and the approximated background. When the distance exceeds a user-defined threshold, the pixel is marked as text, otherwise it is marked as background. Next, a bicubic interpolation up-sampling technique is applied to increase the quality of the final binary image. The final step incorporates a multistage process to eliminate noise and improve the overall quality of the image.

The difficulty in determining a global threshold for poorly illuminated document images led to the various adaptive thresholding techniques. Though the Niblack method works well with poorly illuminated images, it does have two major drawbacks. First, as noted above, it is highly dependent on the window size.
If the window size is large, the method will slow down due to the need to calculate the mean and standard deviation of a large set of pixels for each pixel in the image. However, if the window is too small it may produce background noise. Another drawback of the Niblack algorithm is its dependence on the value of $k$; different values of $k$ produce different results for different images.

Sauvola and Pietikainen proposed a slight variation of the Niblack algorithm in order to account for the background noise it produces in situations in which the background contains light grey values that exceed the threshold value. However, as noted above, this variation requires the user to know the contrast of each document being processed, and set a parameter ($R$) accordingly. For example, in a low contrast document image, a preset value for $R$ could cause the entire document to be set to background, or vice versa.

While the improvements to the Niblack approach are important, it is difficult for a user to understand contrast values. Methods could be developed to automatically determine the contrast and not require as much understanding by the user. The scores below reflect this required user understanding, the quality of the presentation of the method, and its improvement in robustness over the Niblack method.

The method proposed by Gatos, et al. builds on the work of Niblack and Saulvola. This approach uses Saulvola’s method for the initial foreground estimation and thus, is sensitive to the document contrast. Further processing in this method accounts for image degradations caused by shading or low contrast. The scores below reflect the improvements made on this sliding window approach.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>Co</th>
<th>Fl</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Niblack</td>
<td>9/6.6</td>
<td>3/2.4</td>
<td>6</td>
<td>3/1.1</td>
<td>8/2.5</td>
<td>9/2.8</td>
<td>3/2.4</td>
<td>3/2.8</td>
<td>3/2.6</td>
<td>5/4.1</td>
<td>3/2.8</td>
<td>CC</td>
<td>16/10.1</td>
</tr>
<tr>
<td>Sauvola</td>
<td>7/5.1</td>
<td>4/3.2</td>
<td>5</td>
<td>4/1.4</td>
<td>6/1.9</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>5/4.3</td>
<td>3/2.5</td>
<td>2/1.9</td>
<td>DC</td>
<td>10/6.4</td>
</tr>
<tr>
<td>Gatos</td>
<td>6/4.4</td>
<td>6/4.7</td>
<td>4</td>
<td>6/2.2</td>
<td>6/1.9</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>7/6.0</td>
<td>3/2.5</td>
<td>2/1.9</td>
<td>DC</td>
<td>9/7.2</td>
</tr>
</tbody>
</table>

Table 2.8: Adaptive Thresholding Scores

### 2.3 Perspective Correction in 3-Dimensions

A document image may have the text rotated to such a position that it is difficult for OCR algorithms and other document image processing components to operate properly. Thus, methods have been developed to detect and correct these distortions. The documented rectification methods can be categorized under two broad headings: perspective correction and skew correction. Perspective correction deals with a document image rotated away from the camera (i.e. in the z-direction). Skew correction deals with a document image rotated with respect to the bottom edge (i.e. in the x-y direction). Methods for simultaneously correcting documents rotated in all three dimensions have also been developed.
Many of the methods described below use the Hough transform [Hough, 1962]. Recall that the Hough transform was first introduced as a method of detecting complex patterns of points in a binary image. In essence, it converts a global pattern detection problem into a local peak detection problem, by transforming the values in the image space into values in a parameter space. The patterns detected in the methods outlined below are straight lines of text. Figure 2.3 illustrates the perspective correction has on an image.

![Figure 2.3 Document Image Before and After Perspective Correction](image)

### 2.3.1 Two Dimensional Skew Correction

Several methods have been developed that only correct skew distortions. These include an iterative method, automated page orientation detection, an improved Hough transform estimation method, a centroid-based estimation approach, a vertical-line based approach, and a nearest-neighbor chain based approach.

- **Iteration based on Projection Profiles.** Baird [Baird, 1987] proposed an iterative method in which the bottom-centers of each connected component are used to determine the skew angle of the document. This approach utilizes a multi-step approach: First the projection profiles are calculated at a number of different angles. The angle that maximizes the sum of squares is determined to be the document skew. These calculations are performed iteratively until the desired accuracy is reached.

- **Automated Page Orientation Detection.** The methodology proposed by Le, et al. [Le et al., 1994] consists of two general processes, one for detecting the page orientation and another for skew angle detection. The page orientation identification process consists of three steps. First, the document image is divided into blocks and each block is classified as to whether it contains text or not. Each textual block is then used to estimate the page orientation. Finally, each block is clustered with the nine surrounding blocks. Using the estimates from step two, a weight is calculated for each cluster. The cluster values are then used to determine the page orientation.
The blocks identified in the page orientation process are used in the skew angle detection process. The first step in this process is to select a block, segment it into components (i.e. characters), and then label the components. Next a simplified block is created from the chosen block by selecting only the bottom pixels from each component. A Hough transform is then applied to the lines generated from the bottom pixels of the components, resulting in the calculation of the skew angle. The derived skew angle is then used to adjust the document image.

- **Improved Hough Transform Estimation.** The method proposed by Pal and Chaudhuri [Pal and Chaudhuri, 1996] improves on the Hough transform method for skew estimation. The first step of this method uses connected component analysis to identify and label characters. The bounding box of each character is determined and the average height of all characters is used in later calculations. Capital letters and characters with long vertical strokes such as b, d, g, and j are filtered out of the calculations to optimize performance. The upper-left pixel and the lower-right pixel of each character are also determined from the bounding box. Note that the top pixels will fall along a mean line and the bottom pixels will fall on a base line for the line of text. These two lines are used by the Hough transform to estimate the skew angle for a line of text. This estimate is then used to adjust the document image.

- **Centroid-based Skew Estimation.** The method proposed by Yu and Jain [Yu and Jain, 1996] extracts centroids from connected components (textual characters) to be used in a Hough transform to estimate the skew angle of a document image. The centroid of each character is determined by calculating a block adjacency graph, searching the graph, and computing the weighted average of the centroids of the blocks connected in the graph. The centroid of each character is used in the Hough transform to estimate the skew angle, and adjust the document image.

- **Vertical line-based Skew Estimation.** The method proposed by Gatos, et al. [Gatos, et al., 1997] extracts vertical line parameters from run-length textual lines and uses these parameters in a Hough transform to estimate the skew angle in a document image. In this method, two or more vertical lines are identified within the document image. Two of these vertical lines are chosen to calculate the skew angle. By searching for text pixels in one vertical line that correlate with pixels in the other, a joining line can be identified. This joining line forms the hypotenuse of a triangle representing the skew of the image. Thus, using this line and the vertical distance between the points on the two vertical lines, the skew angle can be estimated using simple trigonometry. The angles for each pixel are gathered into a
correlation matrix. The image skew angle is then obtained by determining the global maximum of the vertical projection of the correlation matrix. The document image is then rectified using the discovered skew angle.

- Nearest-neighbor Chain Based Skew Estimation. Lu and Tan [Lu and Tan, 2003] propose a method that identifies nearest neighbor chains and estimates the document image skew angle based on the slope of these chains. The connected components in the document image are identified in the first step of this process. This is followed by the detection of nearest neighbor chains using the centroid distance between two components. Given these chains, the skew angle for each component is calculated. The overall skew angle of the document image is then computed using the skew angle of each component. The document image is then rectified using the computed skew angle.

The iterative method proposed by Baird is novel and straightens the document to 0.3 degrees. A simple direct implementation of this approach would be computationally expensive, however the iterative method described in this paper accounts for this by using successive approximations at a higher fidelity.

The approach proposed by Le, et al. primarily uses the non-textual data to divide a document image into blocks. These blocks in turn are used to determine the page orientation and skew estimation. The accuracy of the skew estimate is improved in this method over other methods due in part to the use of non-textual data.

Pal and Chaudhuri give a greater weight to pixels that contribute to accurate estimation. This is accomplished through the use of pixels in both the mean and baseline of characters. This method is has proven faster than others due to its reliance on typical lower case letters, rather than attempting to account for all cases. However, this method is bounded by the computational complexity of the Hough transform, and thus is not optimal.

The method proposed by Yu and Jain uses the centroids of the connected components, rather than all the pixels in its calculations. This significantly reduces the computational complexity of the method. However, this method also uses the Hough transform and thus is still computationally expensive.

The method proposed by Gatos continues to optimize the Hough transform by using only the pixels used in the vertical line definitions. This increases the performance of this method over traditional Hough transform methods that use all of the image pixels. The dependency on the vertical line definitions also enables this method to perform equally well with mixed text and graphics.
The nearest-neighbor chain based approach proposed by Lu and Tan uses a robust connected component algorithm that will work with both western characters and Chinese characters. Thus, it is capable of solving the skew problem in a general sense. Note that the nearest-neighbor based approaches [O’Gorman, Lu and Tan, and Miao and Peng] are generalized rather than tailor-made to specific document characteristics as many of the other approaches are. This generalization provides for a more robust solution to the perspective rectification problem.

![Image of a table with data]

Table 2.9 Two Dimensional Perspective Correction Scores

<table>
<thead>
<tr>
<th>Method</th>
<th>A</th>
<th>Co</th>
<th>FI</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proj. Profiles</td>
<td>10/7.3</td>
<td>3/2.4</td>
<td>6</td>
<td>5/1.8</td>
<td>9/2.8</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>7/6.0</td>
<td>8/6.6</td>
<td>7/6.6</td>
<td>SC</td>
<td>22/20.2</td>
</tr>
<tr>
<td>Auto Pg Orient</td>
<td>8/5.8</td>
<td>4/3.2</td>
<td>5</td>
<td>4/1.4</td>
<td>6/1.9</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>5/4.3</td>
<td>7/5.7</td>
<td>7/6.6</td>
<td>SC</td>
<td>17/17.0</td>
</tr>
<tr>
<td>Improvd Hough</td>
<td>8/5.8</td>
<td>4/3.2</td>
<td>5</td>
<td>4/1.4</td>
<td>6/1.9</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>5/4.3</td>
<td>9/7.4</td>
<td>7/6.6</td>
<td>SC</td>
<td>18/19.4</td>
</tr>
<tr>
<td>Centroid</td>
<td>8/5.8</td>
<td>5/4.0</td>
<td>6</td>
<td>5/1.8</td>
<td>6/1.9</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>9/8.5</td>
<td>6/5.2</td>
<td>7/5.7</td>
<td>7/6.6</td>
<td>SC</td>
<td>16/14.6</td>
</tr>
<tr>
<td>Vertical Line</td>
<td>6/4.4</td>
<td>6/4.7</td>
<td>5</td>
<td>6/2.2</td>
<td>6/1.9</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>5/4.3</td>
<td>8/6.6</td>
<td>8/7.5</td>
<td>SC</td>
<td>16/13.2</td>
</tr>
<tr>
<td>N-Neighbor</td>
<td>8/5.8</td>
<td>6/4.7</td>
<td>4</td>
<td>6/2.2</td>
<td>7/2.2</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>7/6.0</td>
<td>4/3.3</td>
<td>8/7.5</td>
<td>SC</td>
<td>17/13.1</td>
</tr>
</tbody>
</table>

2.3.2 Three Dimensional Perspective Correction

Several methods have been developed that correct both skew and perspective distortions. These methods include the use of illusory cues, a probabilistic model approach, a vanishing points approach, a method using fuzzy sets, and a morphology-based method.

- **Perspective Rectification using Illusory Cues.** Pilu [Pilu, 2001a] presents a skew rectification approach based on horizontal illusory clues. In this approach horizontal illusory clues are derived from the arrangements of characters into words and lines. First, preprocessing (including down sampling and thresholding) turns textual constructs into blobs that represent characters, words, or lines of text. The blobs are then categorized as elongated or compact based on their size. A pairwise similarity value is then calculated for neighboring blobs. This value measures the likelihood that the pair are part of a text line. An association network is then constructed using the blobs and their association values. In the final step, the network is traversed to identify similar linear groups of blobs that are considered to be horizontal clues. These clues are utilized to calculate skew angles that can then be used for image rectification.

- **Perspective Rectification using a Probabilistic Model.** Dance [Dance, 2002] describes a process of estimating perspective in both the horizontal and vertical direction using horizontal text lines and
vertical paragraph margins. The parallel lines formed by text lines and formatted column boundaries are used to estimate vanishing points. The vanishing points are then used for perspective correction. This computationally expensive process is comprised of the following steps. With the text lines in the image represented as horizontal vanishing points, a histogram of the pixel values for each horizontal vanishing point is constructed. Next, a hidden Markov model (HMM) is constructed to determine the vertical vanishing points from the column boundaries. Each potential vanishing point in the histogram and the HMM are then evaluated and given a score representing the quality of the estimate. The vanishing point with the maximum score is regarded as the estimated perspective. The distorted document image is then rectified using two principal vanishing points, which are estimated based on the parallel lines extracted from the text lines and vertical paragraph margins.

• **Perspective Rectification using Paragraph Formats.** Clark and Mirmehdi [Clark and Mirmehdi, 2003] propose the estimation of two vanishing points based on paragraph formatting. Horizontal vanishing points are calculated using a circular search space. A projection profile of the text is generated for every possible vanishing point in the space. The projection profile can be thought of as a set of bins into which pixels are placed. The bins form a parallel line through the circular search space when the document is aligned horizontally. When a document is skewed, a vanishing point outside of the space can be found. The calculated vanishing point is verified through a set of confidence measures. The vertical vanishing point is calculated using the column boundaries of paragraphs. If the paragraph is fully justified, the left and right margins of the text provide the vertical lines to be used in the calculation of the vanishing point. For left-justified, right-justified, or centered text, at least one straight vertical line may be determined either on the margin, or through the middle of the text. The centered text requires more calculation to determine the vanishing point. Once the horizontal and vertical vanishing points have been determined, the document image can be rectified using these points.

• **Perspective Rectification using Fuzzy Sets.** The technique proposed by Lu, et al. [Lu, et al., 2005] relies on stroke boundaries and tip points (top or bottom point of a character). Structural features of the characters are used to extract tip points at the top and bottom of a row of text. Extraction of stroke boundaries on the right and left side of a character is also based on structural features.
After the tip points and stroke boundaries are extracted, each tip point is classified and the vertical stroke boundaries are identified. The vertical boundaries are then used to construct source quadrilaterals. These quadrilaterals and the document boundaries are then used to estimate target quadrilateral boundaries. The source and target boundaries provide the mapping required to move from the source to the target position. A homography matrix is used to adjust the source positions into the target positions, and thereby creating a rectified image.

- **Perspective Rectification using Morphology.** The approach proposed by Miao and Peng [Miao and Peng, 2006] for perspective rectification consists of a multi-step morphological process. In the first step the horizontal vanishing points are established through connected component analysis and a nearest neighbor chain method similar to Lu and Tan [Lu and Tan 2003]. The chains are then used to fit text lines using linear regression, which enables the horizontal vanishing point to be estimated. Next, the connected components are used to estimate the vertical vanishing points. The connected components are also used to generate a set of overlapping blocks centered at the centroid of each component. A longer horizontal block is then constructed with a length equal to the average height of the components. The vertical character stroke is then estimated using run-length opening operations.

  Pilu’s approach performs well on the estimation of horizontal clues, however it does not reliably extract vertical clues. The proposed method for vertical rectification is dependent on the quality and quantity of the vertical illusory clues obtained. In many document images vertical clues are sparse, other than left paragraph justification. Thus, this approach does not perform well in the correction of other forms of image distortion such as shear. Note also that this approach performs better when the focal length of camera is known.

  Dance used two vanishing points to rectify the distorted document image. These points are estimated based on the parallel lines corresponding to text lines and formatted column boundaries. Unfortunately, no description is provided of how the lines are originally obtained. Similar to Pilu’s approach, this method works best with fully aligned text (i.e. left and right justified paragraphs). The probabilistic model constructed in this approach may be extended to account for other artifacts within the image such as font size and interline spacing, however the maximum angle of correction is limited.
Clark introduced an extension of the 2D projection profile to locate horizontal vanishing point of the text line. The vertical vanishing point of the document is determined based on the change in line spacing due to perspective. Similar to Pilu and Dance’s methods, well-formatted paragraphs are required by Clark’s vertical rectification method. However, this method does rectify image skew in three dimensions.

Unlike the previously discussed methods, Lu used character stroke boundaries and tip points to rectify the document images. This method is not dependent on any a priori input, paragraph boundaries, column boundaries, etc., and works well for small segments of text. However, this approach does require document images to contain uniform font size.

The morphology based approach proposed by Miao and Peng extends the nearest-neighbor chain based approach and effectively rectifies document images that contain both English and Chinese characters. However, the hierarchical nature of the proposed rectification process is computationally expensive. Unfortunately, no computational speed metrics were reported.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>Co</th>
<th>Fl</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illusory Cues</td>
<td>6/4.4</td>
<td>7/5.5</td>
<td>6</td>
<td>8/2.9</td>
<td>8/2.5</td>
<td>5/1.6</td>
<td>1/0.8</td>
<td>3/2.8</td>
<td>8/6.9</td>
<td>3/2.5</td>
<td>6/5.6</td>
<td>DC</td>
<td>14/8.7</td>
</tr>
<tr>
<td>Prob. Model</td>
<td>6/4.4</td>
<td>5/4.0</td>
<td>7</td>
<td>6/2.2</td>
<td>7/2.2</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>6/5.2</td>
<td>6/4.9</td>
<td>7/6.6</td>
<td>DC</td>
<td>15/11.7</td>
</tr>
<tr>
<td>Paragr Format</td>
<td>6/4.4</td>
<td>5/4.0</td>
<td>6</td>
<td>6/2.2</td>
<td>7/2.2</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>8/6.9</td>
<td>3/2.5</td>
<td>8/7.5</td>
<td>DC</td>
<td>16/11.4</td>
</tr>
<tr>
<td>Fuzzy Sets</td>
<td>7/5.1</td>
<td>7/5.5</td>
<td>5</td>
<td>8/2.9</td>
<td>7/2.2</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>6/5.2</td>
<td>7/5.7</td>
<td>8/7.5</td>
<td>DC</td>
<td>16/12.6</td>
</tr>
<tr>
<td>Morphology</td>
<td>6/4.4</td>
<td>6/4.7</td>
<td>6</td>
<td>7/2.5</td>
<td>8/2.5</td>
<td>5/1.6</td>
<td>1/0.8</td>
<td>6/5.6</td>
<td>9/7.7</td>
<td>3/2.5</td>
<td>8/7.5</td>
<td>DC</td>
<td>17/11.6</td>
</tr>
</tbody>
</table>

*Table 2.10: Three Dimensional Perspective Correction Scores*

### 2.4 Page Curl Correction

A document image may contain text that is distorted due to page curl, or what is also called warping. Page curl is commonly found when an image of a document from a large bound book is captured on a flatbed scanner. This distortion can be minimized if the page is pressed firmly against the platen. However, if a user holds a document, such as a newspaper, and captures the document image using a digital camera, there is little that can be done to avoid page curl distortion.

*Figure 2.4 Document Image Before and After Page Curl Correction*
Several methods have been developed to rectify the page curl in a document image. A few of these methods include rectification using applicable surfaces, regression of curved text lines, a cylinder model, Hooke’s law, a Gordon surface, bilinear interpolation, a developable surface, and line estimation.

- **Page Curl Rectification Using Applicable Surfaces.** Pilu [Pilu, 2001b] proposes using applicable surfaces to correct page curl. This method mathematically maps an applicable surface to the curled page. Deformed applicable surfaces such as sheets of paper satisfy the differential geometric constraints of isometry and vanishing Gaussian curvature. Thus, given knowledge of the proper form of the shape, and these geometric constraints, the full geometric structure of the document image can be recovered. However, the implementation of such a surface is impractical. Thus, Pilu used a triangular mesh to calculate a finite element approximation of the surface. This approximation method is comprised of several steps. First, the triangular mesh is constructed and placed onto the image. A gradient descent method iteratively adjusts the position of the nodes of the mesh until convergence is achieved, thus creating an applicable state. Finally, the texture is mapped onto the surface of the document image and the image is rectified.

- **Page Curl Rectification Using Regression of Curved Text Lines.** The method proposed by Zhang and Tan [Zhang and Tan, 2003] applies a multi-step regression based approach to the page curl correction problem. First, connected components of the straight text lines are clustered together and modeled using linear regression. The connected components are also clustered into a set of warped text lines, and the warped text lines are modeled using polynomial regression. Next, the warped text lines and the corresponding straight lines are aligned. Rectification is enabled by correcting the polynomial curves based on the corresponding straight text lines, and moving the corresponding components to restore straight horizontal baselines.

- **Page Curl Rectification Using a Cylinder Model Approach.** The method proposed by Cao, et al., [Cao, et al., 2003] models a curled page as a cylindrical surface. The first step of this rectification process is to determine as many horizontal text lines as possible through a connected component searching method. These lines are then used to identify left and right boundaries of the text. Assuming that these boundaries are vertical, the X and Y directions are determined. Two text lines are then used to generate
two directrixes (used to model conic sections). Using a mapping function, the image is rectified by mapping coordinates of the directrixes in the original image to a flat surface image.

• **Page Curl Rectification Using Hooke’s Law.** The method proposed by Brown and Seals [Brown and Seals, 2004] was developed to restore portions of damaged manuscripts. Within this setting, they were able to employ a digital camera mounted on a gantry perpendicular to the manuscript to capture a 2D image of the document. They also used a structured-lighting system composed of an LCD projector and a digital camera. The camera, light source, and manuscript were calibrated such that a 3D image could be obtained. Using a mapping of the 2D image to the 3D image, an approach based on Hooke’s Law was used for rectification. Hooke's law states that the amount by which a body is deformed is related to the force causing the deformation. The proposed approach models the document image as a 3D particle system. The particles are connected by “springs” which follow Hooke’s law. The rectification process begins when “external forces” are applied to the image to cause overall structure change. “Internal forces” hold the image together and keep the particles equal distance apart as the image is restored.

• **Page Curl Rectification using a Gordon Surface.** Zhang and Tan [Zhang and Tan, 2005] propose a method in which a 3D Gordon surface model of the distorted page is constructed from a set of lines extracted from the captured document image. After connected component analysis is complete, the text lines are represented as natural cubic splines interpolated from points within the connected components. A ruled surface model based on the text lines is constructed, and represented as a Gordon surface. A second natural cubic spline is then constructed using the bottom boundaries of the connected components (characters). This spline is then used to generate a restored document image through a mapping function derived from the Gordon surface model.

• **Page Curl Rectification using Bilinear Interpolation.** Ulges, et al. [Ulges, et al., 2005] propose another method to straighten curved text lines through estimation of the horizontal baselines. In this method, the reasonable assumption is made that parallel lines have a constant separation, and thus when they get closer together, their distance from the camera has increased. Stated another way, objects of equal size appear smaller as their distance from the viewer increases. Thus, for processing document images, the assumption continues that line spacing is uniform over the original document and that the distance between text lines decreases with an increasing distance form the camera.
Beginning with connected component analysis, the text lines are estimated using the RAST algorithm [Breuel, 2002]. Using a set of character bounding boxes derived from the connected components, the RAST algorithm is used to determine the base line for the bounding boxes while accounting for the descenders in letters like ‘p’ and ‘q’. The baseline approximations are then used to estimate line spacing by measuring the distance from one row to the next. Finally, these estimates are employed by a bilinear interpolation routine to rectify the warped image.

- **Page Curl Rectification using a Developable Surface.** Recall that a surface that can be unrolled onto a plane without tearing or stretching is a developable surface. Liang, et al., [Liang, et al., 2005a] describe a method to restore a warped document image to a planar image using a developable surface model rather than a cylinder. The assumption is made that the page forms texture flow fields that constrain the underlying surface of the document.

  The first step in this method is to model the warped document image as a developable surface. The tangent vector for every point on the developable surface is then identified. A surface transformation method is employed until all of the tangent vectors of the surface are parallel. When this occurs the surface has been flattened. The texture flow fields from the textual regions are extracted and used to derive the projected textual lines. The textual lines are then used to determine vanishing points. Finally, the document image can be rectified using the estimates of the page shape based on the texture flows and vanishing points.

- **Page Curl Rectification using Line Estimation.** The method proposed by Keefer, et al. [Keefer, et al. 2009a] follows a relatively simple process based on line estimations. Similar to other methods, connected component analysis is used to identify and place a bounding rectangle around each character. The characters are clustered into words and lines based on the Euclidean distance from the center of the left side of the bounding rectangle of one character to the center of the right side of the bounding rectangle of the other components. Based on the line estimations derived from this clustering procedure, vanishing points are estimated and used to remove the perspective distortion. Finally, the image is rectified one character at a time rather than adjusting the entire line. The characters are rotated and moved to align with a RANSAC [Fischler and Bolles, 1981] estimated horizontal line.

  The applicable surfaces approach proposed by Pilu relies heavily on a custom 3D image acquisition system to obtain the document image. Thus, the approach is prone to distortions if the calibration is not
precise. This reliance on special equipment renders this technique impractical in a mobile setting or applied to images captured using only digital camera. While this approach is original and has inspired others to build upon it (the developable surface and Gordon surface approaches in particular), it is more complex than other approaches.

In the regression based approach presented by Zhang and Tan, curved text lines are straightened by first finding the text lines using connected component clustering methods, and then moving the components to restore horizontal baselines. While this approach does not specify an ending shape for the document image, the final image can be run through an OCR successfully.

Cao’s approach estimates the cylindrical shape of a page near the binding of an opened book by using the horizontal text-lines in the image. These text-lines are discovered by a bottom-up clustering of connected components. The model simulates the surface of the document image captured by the camera, and assumes that the horizontal text lines follow the direction of a directrix. While the method is described well, it is more complex than other approaches.

The approach proposed by Brown and Seales assumes a structured lighting system, which requires special image capture hardware and strict calibration for obtaining 3D data. The application of Hooke’s law primarily focuses on damaged manuscripts. Thus, the custom configuration and complexity of the model would be difficult to generalize to a mobile setting or applied to images captured using only a digital camera.

In the Gordon surface based approach presented by Zhang and Tan, the document image is represented as a 3D Gordon surface. As this surface is mapped to a 2D spline, the original shape of the document is approximated. This method relies on structured text lines that are long enough for the warped line to be mapped to the Gordon surface. Thus, this approach may prove to be problematic for correcting short, steep curves found near the inner edge of some bookbindings. This method is also computationally expensive as compared with other methods.

The RAST based method proposed by Ulges, et al., estimates line spacing derived from baseline approximations. The rectification process utilizes a bilinear interpolation routine based on these estimates. Unfortunately, this method only applies to document images containing a single column of text, rather than multicolumn; nor does it account for vertical alignment. Thus, not all characters are aligned properly after
the line has been straightened. However, even with this deficiency, the reported error rate of rectified
document images tested with an OCR was reduced by up to 90%.

Liang, et al., propose a process for computing vanishing points of a curved document image. The
assumption that textual lines are parallel is modeled as texture flow vectors on the curved page to detect the
horizontal lines and the text line spacing on the page. A developable surface is fitted to the detected lines
and texture flow fields, and the surface is unrolled to generate a rectified document image. While the
presentation of the process is very good (including pseudo-code), the experiments presented utilized high-
resolution images, and the process is computationally expensive compared with other methods. Thus, this
process may be difficult to generalize for use in other settings.

The approach proposed by Keefer, et al., uses estimation of horizontal lines using cues in the document
image to undo page curl distortion of the image. The rectification process first groups the connected
components into lines and uses the RANSAC algorithm to approximate the base line, estimating the slope
of the base line for dewarping each character. The method assumes that text lines are left and right justified,
and parallel. While this process is not as computationally expensive as others, it is less accurate than other
methods reporting an OCR accuracy rate of 97%.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>Co</th>
<th>Fl</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appl. Surfaces</td>
<td>6/4.4</td>
<td>8/6.3</td>
<td>5</td>
<td>8/2.9</td>
<td>8/2.5</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>7/6.0</td>
<td>3/2.5</td>
<td>2/1.9</td>
<td>CC</td>
<td>10/5.2</td>
</tr>
<tr>
<td>Curved Lines</td>
<td>1/0.7</td>
<td>1/2.4</td>
<td>1</td>
<td>1/1.1</td>
<td>1/2.5</td>
<td>1/0.3</td>
<td>1/0.8</td>
<td>1/0.9</td>
<td>1/0.9</td>
<td>1/0.8</td>
<td>1/0.9</td>
<td>-</td>
<td>4/4.2</td>
</tr>
<tr>
<td>Cylinder Model</td>
<td>7/5.1</td>
<td>4/3.2</td>
<td>6</td>
<td>8/2.9</td>
<td>6/1.9</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>6/5.6</td>
<td>6/5.2</td>
<td>3/2.5</td>
<td>7/6.6</td>
<td>DC</td>
<td>14/9.9</td>
</tr>
<tr>
<td>Hooke’s Law</td>
<td>6/4.4</td>
<td>9/7.1</td>
<td>7</td>
<td>8/2.9</td>
<td>7/2.2</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>7/6.0</td>
<td>7/5.7</td>
<td>2/1.9</td>
<td>CC</td>
<td>10/5.9</td>
</tr>
<tr>
<td>Gord. Surface</td>
<td>7/5.1</td>
<td>5/4.0</td>
<td>5</td>
<td>7/2.5</td>
<td>6/1.9</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>6/5.6</td>
<td>6/5.2</td>
<td>3/2.5</td>
<td>7/6.6</td>
<td>DC</td>
<td>14/10.0</td>
</tr>
<tr>
<td>Bilinear Inter.</td>
<td>6/4.4</td>
<td>3/2.4</td>
<td>5</td>
<td>4/1.4</td>
<td>6/1.9</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>6/5.6</td>
<td>6/5.2</td>
<td>8/6.6</td>
<td>7/6.6</td>
<td>DC</td>
<td>19/20.2</td>
</tr>
<tr>
<td>Dev. Surfaces</td>
<td>8/5.8</td>
<td>4/3.2</td>
<td>6</td>
<td>8/2.9</td>
<td>6/1.9</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>8/6.9</td>
<td>3/2.5</td>
<td>5/4.7</td>
<td>DC</td>
<td>12/8.8</td>
</tr>
<tr>
<td>Line Estimation</td>
<td>6/4.4</td>
<td>3/2.4</td>
<td>5</td>
<td>4/1.4</td>
<td>6/1.9</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>6/5.2</td>
<td>3/2.5</td>
<td>7/6.6</td>
<td>DC</td>
<td>16/13.8</td>
</tr>
</tbody>
</table>

Table 2.11: Page Curl Correction Scores

2.5 Document Image Segmentation

Document image segmentation is often the first step in document image analysis and understanding.
Many methods have been developed for segmentation. Although, fully automatic segmentation is yet to be
achieved, the techniques are mature enough to support commercial OCR systems. In this evolving field,
new algorithms for segmentation are continuously being proposed. A few methods that have been
developed include segmentation using an RLSA method, the invention of the Docstrum, an OCR approach,
selection of Voronoi edges, use of Kruskal’s algorithm, use of soft decision integration, a parameter-free method, and a neuro-fuzzy approach.

- **Segmentation using the RLSA method.** The traditional approach for document image segmentation developed by Wong, *et al.* [Wong, *et al.*, 1982] uses a nonlinear run length-smoothing algorithm (RLSA) to identify textual regions on a page. The first step of this approach is to identify sub-regions of a document image using an extension of a RLSA algorithm. This extension is used to create a bitmap of white and black regions that represent areas of content in the image. This process is applied in both a horizontal and vertical direction resulting in two binarized images. A segmented image is produced when these two images are combined using a logical AND operation. This image is then analyzed further for text regions resulting in a document image that only contains text.

- **Segmentation using the Docstrum.** The method proposed by O’Gorman [O’Gorman, 1993] uses only the neighborhood between connected components to locate the text areas in an image. In this approach, connected components in close proximity to one another are considered a text block. Two connected components (characters) are considered neighbors if the distance between them is lower than a maximum threshold.

Page segmentation is then accomplished through a multistep process. After connected component analysis is performed and the bounding box of each component is noted, the nearest neighbors of each component are discovered. The nearest neighbors are then clustered based on horizontal separation of neighbors. Two spacing histograms are then compiled using the clusters: one representing nearest-neighbor distances for all angles within a specified angular range of the orientation estimate, and one for angles within the same angular range of the perpendicular to the orientation estimate. Using the histograms, text lines and skew orientation are determined.

One or more text lines are grouped forming a structural block when two text lines are nearly parallel and close together. The orientation of the blocks is determined using a bottom-up, k-nearest neighbor approach. Independent components are grouped when a path can be found from one component to another through a nearest neighbor path. Page segments are then discovered by assembling groups of components with similar orientation.

- **Segmentation using a Labeled X-Y Tree.** The method proposed by Krishnamoorthy, *et al.* [Krishnamoorthy, *et al.*, 1993] for image segmentation is comprised of two primary steps. First, the
document image is decomposed into blocks of text using a nested X-Y tree data structure. Second, a string of text extracted from the text-block and examined using a context-free grammar. The results of this analysis lead to the classification of the contents of the blocks into logical components.

- **Segmentation using an OCR Approach.** Bourbakis [Bourbakis, 1996] presents a top-down method for separating text from images. This method begins with a top-down pyramidal method to isolate a first order approximation of the text and image regions. Next, the potential textual regions are scanned for characters. A Chain Code method is used to identify potential characters. The shape representing a potential character is extracted and analyzed to verify that it is a character. Non-character shapes are treated as images. Shapes that are found to be characters are classified according to size and font. A bounding box is then created for each character. Characters sufficiently close are clustered and considered to be words. If a gap between two characters is greater than a user defined threshold, the new character is considered to start a new word. Horizontal groups of words are clustered together as lines of text. Finally, these lines of text are identified as textual regions.

- **Segmentation using selection of Voronoi edges.** Kise, et al. [Kise, et al., 1997] proposed a method for identifying text blocks by constructing a point Voronoi diagram to determine segment boundaries based on the distance between the points. This method begins with an area Voronoi diagram that is generated based on connected component analysis within the document image. The excess Voronoi edges are then removed from the diagram. This removal is based on a threshold measure determined for each document image. The remaining edges in the diagram form the column boundaries and potential paragraph or image boundaries.

- **Segmentation using an augmented Kruskal’s algorithm.** Simon, et al., developed an efficient bottom-up segmentation method based on the rectilinear physical layout of technical journals [Simon, et al., 1997]. In the first step of this method, characters are identified using connected components analysis. Next, the distances between pairs of characters are calculated using a minimal-cost spanning tree, built with Kruskal’s algorithm. Similarly, a minimal-cost spanning tree is constructed for the words, text lines, and column segments of a document image. Document image segments are identified based on evaluation of this spanning tree.

- **Segmentation using soft decision integration.** Etemad, et al. [Etemad, et al., 1997] propose a document image segmentation method based on wavelet analysis and a neural network decision guide. The first
step in this method is to analyze the document image using a pyramidal wavelet transform and a separation based wavelet packet tree. Features are extracted from this analysis and overlapping sub-blocks are identified. The extracted features form the input parameters to a neural network that was developed to automatically determine the segmented regions within the overlapping sub-blocks.

- **Segmentation using a parameter-free method.** Lee and Ryu propose a parameter-free geometric document segmentation method [Lee and Ryu, 2001]. This method begins with the construction of a pyramidal structure of the document image. This is followed by connected component analysis, and the extraction of bounding boxes from the connected components. The periodicity of each sub-region is then estimated in the horizontal and vertical directions for each level in the pyramidal structure. The regions with inconsistencies are analyzed for places in which they could be split into two regions. The larger regions are divided into smaller and smaller segments until each becomes a single periodical region. Segmentation is then determined through the identification of homogeneous regions.

- **Segmentation using a neuro-fuzzy approach.** The segmentation approach proposed by Caponetti, et al. [Caponetti, et al., 2008] begins with the classification of each pixel as a part of a text region, an image region, or background region based on a neuro-fuzzy learning approach. Pixels are then merged together to form groups of coherent regions using a set of morphological operators. These groups are then verified by an analysis procedure of each region. This regional analysis is also based on a neuro-fuzzy learning approach. The result of this multi-step approach is a classification of each region of a segmented document image.

The text recognition method described by Wong employs a pre-classification technique comprised of a decision network. The network accepts a pattern array and produces a list of possible matching prototype patterns. By implementing such a sub-module, pattern matching is reduced to the identification of prototypes. This pattern-matching scheme is simple to implement and supports a number of fonts and sizes.

While the *Docstrum* approach presented O’Gorman only applies to textual regions, there are several advantages that have made this method popular. First, the *Docstrum* method is independent of page orientation, which enables the method to be performed without a skew correction preprocess, and makes this approach more robust. Secondly, this approach is also independent of character size and line spacing. These two advantages lead to a third, namely that the *Docstrum* method can be applied to regional analysis. One disadvantage of this approach is the user required threshold parameters.
Though the model-driven approach presented by Krishnamoorthy, *et al.* is quite novel, it requires the user to construct grammars for the segment classification and has a low tolerance for skew distortion. This approach is useful for verifying the contents of small regions that may otherwise be classified as noise.

The method proposed by Bourbakis is a simple and easy to implement approach based on the human visual system of processing a document through global recognition of patterns rather than a detailed analysis. It is robust in that it processes both typed and handwritten text in a region. However, it is also slow because it is dependent upon character recognition prior to the framing process. This method adjusts for skew and page curl, and is robust for both typed text and handwritten text.

While the approach presented by Kise, *et al.* is limited to document images in which column regions are dominant, it is relatively robust in that it is able to extract document components of arbitrary shape regardless of the two-dimensional skew angle of the document image. Unfortunately, this method has a tendency to fragment textual headings, and thus does not work on arbitrary font size. As with many of the methods discussed in this survey, the speed of this approach is predominantly bound by the connected component analysis.

The approach developed by Simon, *et al.* is found by traversing a minimum spanning tree that is constructed in such a way that the edge lengths are chosen from the larger of the horizontal and vertical distances between the bounding boxes of connected components. The sub-trees of the spanning tree correspond to physical layout units, such as paragraphs. The model assumes horizontal text lines, and is robust enough to tolerate up to 5 degrees of document image skew.

To create a layout-independent document image segmentation algorithm, Etemad, *et al.* derived rotation-invariant features at different scales from wavelet packets. A neural network was then used to differentiate the features, and determine segment blocks. The neural network was trained with a set of manually segmented image samples. Once the blocks have been determined, neighboring blocks are then combined using a weighting scheme until a threshold is met. A post-processing step is then applied to recover the layout of the page. This method is computationally expensive, and was created specifically for texture based document image segmentation. Thus, for highly structured document images, other methods are more appropriate.

Lee and Ryu use a pyramidal quad-tree structure, a periodicity measure, and texture analysis to determine the page segments of a document image. The segments identified are based on homogeneous
regions such as text, images, and tables. This method is not dependent on any user parameters and supports large font sizes and text line spacing. This method is computationally expensive and is focused specifically on geometric segmentation.

The neuro-fuzzy approach to document image segmentation proposed by Caponetti, et al. is also computationally expensive, and produces results similar to those reported by other methods. However, this segmentation process is not affected by document image skew, and in fact determines the skew angle during processing. This method effectively classifies text and image portions of an image and is not limited by font size.

This method scores low in availability, as does the Etemad method, due to the cursory level description of their processes. This value is not intended to diminish the value of the work, only to note that a large amount of information was presented in a small amount of space, and thus much of the detail was left out. Without detail it would be difficult to implement the proposed solutions directly from the information presented in the paper.

While there is not a standard method to measure the performance of a segmentation technique, some empirical measures have been proposed for the evaluation of proposed methods [Das, et al., 2002]. However, this survey is based on the results of the industry weights of the various aspects of the proposed methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>A</th>
<th>Co</th>
<th>FI</th>
<th>MC</th>
<th>O</th>
<th>P</th>
<th>RP</th>
<th>Re</th>
<th>Ro</th>
<th>Sp</th>
<th>U</th>
<th>ST</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLSA Method</td>
<td>8/5.8</td>
<td>3/2.4</td>
<td>5</td>
<td>2/0.7</td>
<td>8/2.5</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>6/5.2</td>
<td>3/2.5</td>
<td>3/2.8</td>
<td>SC</td>
<td>16/13.5</td>
</tr>
<tr>
<td>Docstrum</td>
<td>7/5.1</td>
<td>7/5.5</td>
<td>3</td>
<td>5/1.8</td>
<td>9/2.8</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>8/6.9</td>
<td>6/4.9</td>
<td>5/4.7</td>
<td>SC</td>
<td>18/15.3</td>
</tr>
<tr>
<td>X-Y Tree</td>
<td>8/5.8</td>
<td>5/4.0</td>
<td>6</td>
<td>7/2.5</td>
<td>8/2.5</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>5/4.7</td>
<td>5/4.3</td>
<td>5/4.1</td>
<td>5/4.7</td>
<td>SC</td>
<td>14/8.8</td>
</tr>
<tr>
<td>OCR</td>
<td>7/5.1</td>
<td>3/2.4</td>
<td>3</td>
<td>4/1.4</td>
<td>7/2.2</td>
<td>5/1.6</td>
<td>1/0.8</td>
<td>6/5.6</td>
<td>8/6.9</td>
<td>3/2.5</td>
<td>5/4.7</td>
<td>SC</td>
<td>17/16.8</td>
</tr>
<tr>
<td>Voronoi Edges</td>
<td>7/5.1</td>
<td>7/5.5</td>
<td>3</td>
<td>7/2.5</td>
<td>9/2.8</td>
<td>9/2.8</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>8/6.9</td>
<td>5/4.1</td>
<td>7/6.6</td>
<td>SC</td>
<td>18/14.1</td>
</tr>
<tr>
<td>Kruskal’s Alg.</td>
<td>6/4.4</td>
<td>6/4.7</td>
<td>4</td>
<td>5/1.8</td>
<td>7/2.2</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>7/6.0</td>
<td>9/7.4</td>
<td>7/6.6</td>
<td>SC</td>
<td>19/18.8</td>
</tr>
<tr>
<td>Soft Decision</td>
<td>5/3.7</td>
<td>9/7.1</td>
<td>6</td>
<td>9/3.2</td>
<td>9/2.8</td>
<td>6/1.9</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>7/6.0</td>
<td>6/4.9</td>
<td>2/1.9</td>
<td>SC</td>
<td>12/6.1</td>
</tr>
<tr>
<td>Param Free</td>
<td>7/5.1</td>
<td>6/4.7</td>
<td>4</td>
<td>7/2.5</td>
<td>7/2.2</td>
<td>8/2.5</td>
<td>1/0.8</td>
<td>8/7.5</td>
<td>7/6.0</td>
<td>5/4.1</td>
<td>7/6.6</td>
<td>SC</td>
<td>16/12.8</td>
</tr>
<tr>
<td>Neuro-Fuzzy</td>
<td>5/3.7</td>
<td>9/7.1</td>
<td>6</td>
<td>9/3.2</td>
<td>8/2.5</td>
<td>7/2.2</td>
<td>1/0.8</td>
<td>7/6.6</td>
<td>7/6.0</td>
<td>3/2.5</td>
<td>6/5.6</td>
<td>SC</td>
<td>14/8.9</td>
</tr>
</tbody>
</table>

Table 2.12: Document Image Segmentation Scores

2.6 Conclusion

In this chapter a comparative survey of document image processing techniques based on several aspects of an overall system design was presented. The purpose in performing such a survey was to determine which methods were most useful in the development of techniques to support a mobile reading
device for the visually impaired. Thus, the intent was to determine the current state-of-the-art and understand areas in which improvements are required to support our efforts.

By conducting this survey it was discovered that there is no system or combination of techniques that has fully matured from a user’s perspective. However, it is also very difficult to develop a usable system that supports the important user centered design qualities, while keeping the implementation simple.

In this cooperative comparative survey, aspects that indicate the maturity and usefulness of a method were considered. The evaluations were categorized by functionality, and based on the overall maturity and potential for improvement. Thus, if a method has a higher evaluation score than another method, there is no inference that the first method is somehow ‘better’ than second method. There are too many aspects of a technique to determine a generalized ‘best’ solution. Thus, each application of a technique will depend on the problem at hand.
Chapter 3 Document Image Processing Pipeline

As described in Chapter 2, several image processing techniques must be combined in order for a device to read a printed document. There are several challenges that must be addressed to construct such a system. First, the resolution of the document image that can be obtained using inexpensive cameras that are small enough to be mounted onto a pair of glasses is not sufficient for obtaining acceptable OCR performance. The document image captured by these low-resolution cameras must be enlarged and enhanced for acceptable text recognition. Second, since a user could potentially hold a document for the camera, the resulting image could suffer from both perspective distortions and the page curl. Many of the methodologies present in the literature only deal with either 2-D skew distortion or with perspective distortion in the document image, and even fewer systems can account for perspective distortions and page curl. However, given the surface nature of the pages in newspapers and books, the non-linear distortion due to page curl is inherently present. State-of-the-art OCR systems have a very low performance on recognizing such distorted text, and commercial and prototype mobile readers do not account for these distortions at all.

In order to accommodate robust user interaction with the reader, a document image segmentation method must be incorporated that will separate images from text, identify headings within the document image, and identify article content within the document image. Another aspect of the user interaction with the reader will guide the user in positioning the device in order to capture a quality image. Thus, the user will capture more than one image and the guidance system of the device must track the location of each image. The processing to support this interaction is described in Chapter 4. This chapter will focus strictly on the document image-processing pipeline that is employed to correct a document image and identify the headings and article body within the document image, namely those steps illustrated in Figure 3.1.

Figure 3.1 Document Image Processing Pipeline

The first step in the TYFLOS image-processing pipeline is binarization, the process that transforms a color image into a black and white image (or binary image). Section 3.1 describes the binarization processes employed by the TYFLOS prototype. As will be demonstrated, the binarization process alone
does not produce satisfactory results with the low resolution images captured by the TYFLOS cameras. Thus, the low resolution of the document image must be enhanced.

Simply enlarging an image blurs it and may introduce noise. Section 3.2 describes a neural network-based image enhancement technique that removes artifacts in the high-resolution image and improves the OCR of the document image. As noted above, the document image captured by a camera may suffer from both perspective distortion and page-curl. Thus, Section 3.3 describes a method to remove the perspective distortion and Section 3.4 describes a page curl correction technique. When these two techniques are combined a flattened and straightened view of the text is produced.

In order to identify page headings and textual content of a page, the final three steps of the document image-processing pipeline must be completed. The first step of these three is the segmentation of the document image. A pyramidal segmentation approach is described in Section 3.5. The next step in the process is to identify the headings on the page. Section 3.6 describes a process for this identification based on the font sizes of the text segments within the document image. Section 3.7 describes the final step in this pipeline, which is the process that frames, or groups, related text segments into an article.

This chapter concludes with experimental results and further discussion.

3.1 Document Image Binarization

Binarization generally refers to the conversion of a grey-scale image into a binary image. TYFLOS captures a color image, thus the image is first converted to a grey-scale image using a common conversion technique. The color values of each pixel are used to calculate a new grey value using equation 3.1 where the color values range from 0 to 255.

$$GreyValue = (0.3)(RedValue) + (0.59)(GreenValue) + (1.1)(BlueValue)$$ (3.1)

Researchers have developed different global and local thresholding methods for binarization, many of which are described in Section 2.2. TYFLOS employed three different global thresholding methods in order to produce comparative results as well as to maximize the OCR accuracy. These methods include a simple global average technique, Niblack’s adaptive thresholding technique, and Otsu’s maximum entropy technique.

In the simple global average technique, the document image is scanned and the total value of each grey-scale pixel is summed. This total is then divided by the number of pixels in the image, thus producing an average pixel value for the entire image. In order to produce the binarized image, the grey-scale image is
scanned again and each pixel value that falls below the average is mapped to a value of 0 in the binarized image. Similarly, each pixel value that is equal-to or greater than the average is mapped to a value of 255 in the binarized image.

Niblack developed the most common thresholding algorithm used in document image analysis [Niblack 1986]. This algorithm can be understood by considering a rectangular window slid over a grey-scale image and calculating threshold values within the window. The local threshold values are calculated for each image pixel by using the intensity of the pixels within a small neighborhood window. The threshold T is computed using the mean $m$ and standard deviation $s$ of all the pixels in the window:

$$T = m + (k)(s)$$

(3.2)

where $k$ ($0 < k < 1$) is a user-defined constant that determines the thickness of the binarized stroke that is retained. This local threshold is then used to binarize the pixels within the window, and as the window is slid over the image, the complete document image is binarized. Clearly, the size of the sliding window and the value of $k$ affect the quality of the resulting image.

Otsu’s method [Otsu 1979] separates the pixels of a normalized grey-scale document image into two classes: foreground and background. An initial intensity threshold is calculated based on the mean and variance of each class. Subsequent threshold values are calculated for each intensity level until one is found that minimizes the variance within a class. This method reduces the thresholding problem to a search problem (i.e. search for an optimized threshold value), which has more potential for identifying an optimal threshold than the global thresholding method. Figure 3.2 shows the original image and the corresponding binary image of a newspaper page using these three different binarization techniques.

![Figure 3.2 Binarized Document Image using Average, Niblack’s Adaptive, and Otsu Binarization Methods](image)

### 3.2 Document Image Enhancement

As is illustrated in Figure 3.2, binarization techniques alone may not produce an image that can be readily processed by other document image processing techniques. Many binarization techniques do not explicitly consider the variations in illuminations, shadows, and text bleed through, and hence leave noise
in the non-text areas. (Text bleed through is a pattern observed when the characters on the reverse side of the page become a part of the document image.)

An effective method for removing the effects of illumination and other noise from document images is to divide each input pixel with the local mean calculated from a large neighborhood area. The assumption for considering a large neighborhood is that text occupies a larger portion of the image than the non-text area. Note that this overall method of scaling each pixel with the mean intensity of the neighborhood acts similar to a Retinex filter [Pilu and Pollard, 2002]. The three thresholding techniques described above are then applied on the mean subtracted image to obtain a binary output.

Given a gray-scale document image, the first step in the process is to separate the page from the background. For this, assume that the input image always contains a simple background and the image largely contains text. Based on this assumption, the text portion is extracted as the largest rectangular object in the image. The text is assumed to be in the up-right position and hence no rotations are needed. Once the page is separated from background, the image can be binarized. Binarization helps in identifying the connected components (characters) necessary for the base line fitting described in Section 3.3.

OCR systems often fail when the text font of capital case letters is less than 8 pixels. The resolution of the document image captured by the TYFLOS cameras is very low (640 X 480), so in order for an OCR system to accurately recognize text found within a document image, the resolution of the text must be large enough to distinguish one character from another. Hence, the low-resolution images obtained must be enlarged and enhanced for the OCR to reliably recognize text. The process of obtaining high-resolution image from one or multiple low-resolution images is called super resolution. [Chaudhuri, 2001 and Park, et al., 2003] An image simply obtained by up-sampling, by either pixel replication or interpolation, contains a greater number of pixels than the corresponding low-resolution image, but does not contain any more detail. On the other hand, super resolution techniques improve the perceived detail of an image compared with that of the low-resolution images. These techniques typically involve the restoration of the high frequency contents.

Many approaches have been proposed for the super resolution enhancement of text including a maximum likelihood approach [Capel and Zisserman, 2000 and Capel and Zisserman, 2001], an iterative back projection algorithm [Irani and Peleg, 1991], a Bayesian approach incorporating a prior on the high resolution image [Schultz and Stevenson, 1996], an estimator that uses total variation norm [Vogel and
Oman, 1998], a Bayesian algorithm that use text-specific bimodal prior [Donaldson and Meyers, 2005], and a hallucination algorithm [Baker and Kanade, 2002]. The TYFLOS approach to enhance the text in the images uses a neural network to estimate the high frequency contents of the low-resolution images.

First, the initial high-resolution image is estimated from the low-resolution image by using a standard bi-cubic interpolation method. This bi-cubic interpolation blurs the image to a great extent, which reduces the high frequency details of the image. A neural network is used to estimate the pixel content corresponding to the high-resolution image. The network learns to estimate the high frequency components of the high-resolution pixel given a pixel with certain neighborhood area.

The network considered in this research is a multilayer perceptron with two layers. The input layer consists of 25 neurons (corresponding to a 5 * 5 neighborhood), the hidden layer consists of 10 neurons, and the output layer has 1 neuron corresponding to the center pixel of the neighborhood. All the input and output values are normalized in the range [-1 1]. The input to the network is the normalized pixel gray value along with its neighborhood, and the output is the expected enhanced normalized gray value. The network is trained using a conjugate gradient back propagation algorithm. The error function is estimated as the distance between the network’s estimate and the expected pixel gray value estimate (obtained from the corresponding high resolution image). The activation functions used are hyperbolic tangent functions at the input, and the identity function at the output layer. It should be noted that the conventional neural network that uses a sigmoid function at the output layer is typically used for binary classification purposes, and thus was not utilized in this effort. The network was trained on a set of text-page images. For training purposes, the high-resolution images are first down sampled using a factor of 3 and the corresponding bi-cubic up scaled images are used as input to the system. Figure 3.3 shows an original image and the enhanced images when processed with the average, Niblack adaptive, and Otsu binarization techniques.

Figure 3.3 Enhanced Image using Average, Niblack’s Adaptive, and Otsu Binarization Methods
3.3. Perspective Correction

When a document image has an appropriate resolution, any distortions due to perspective or page-curl can be rectified. The base lines of text on the surface of the page serve as cues inherent in the document image that are used to remove both of these distortions and produce a flattened view of the text.

In order to correct the perspective distortion, the TYFLOS prototype follows a multistep process. A connected component analysis is performed on the binarized image to identify letters/characters in the document image. Ideally, each connected component will correspond to a letter or a character. However, these components might not correspond exactly to a single character due to noise and binarization errors. Figure 3.4 shows the characters obtained after identifying connected components, each marked with a bounding rectangle.

![Figure 3.4 Bounding Rectangles of Connected Components Before Correction](image)

From these connected components, characters are identified and grouped that are present in a single line. For each character, the successor is identified based on the distance between these characters. The distance is calculated as the Euclidean distance between the center of the left side of the bounding box of the current component and the center of the right side of the bounding box of the other components. To account for various page-curl distortions, the distance of all the components in the surrounding rectangular neighborhood is calculated and the component that has a minimum distance as the successor to that component in that line is chosen. This neighborhood area is chosen depending upon the input image resolution and the format of the text. It is assumed that the format of the text on a page does not vary much and the line spacing in a page remains mostly constant.

Due to perspective distortion, parallel lines as perceived by humans are not seen as parallel lines in the document image. One method for removing perspective distortion is to calibrate the cameras and use this calibration matrix to transform the image. However, in a wearable environment, the camera calibration is not feasible. To overcome this problem, the horizontal and vertical vanishing points are computed and the
distortion is removed using these vanishing points. For this, a robust approximation of the horizontal and vertical lines is needed. Estimation of horizontal and vertical lines is obtained using cues present in the document image. The letters contained within each line are used to compute the horizontal and vertical lines in the rectified image based on those known to be parallel in the original document image.

In general, the base lines of the text are assumed to be parallel. Hence, it is crucial to estimate the base lines of the text present in the image. To obtain an accurate estimate of the base lines, RANSAC modeling [Fischler and Bolles, 1981 and Forsyth and Ponse, 2003] is used. Note that an ordinary least squares fit or using splines will not do the task as some of the characters such as ‘p’, ‘q’, ‘y’, etc. will extend below the base line. On the other hand, the RANSAC method will consider these as outliers and then form the baseline using inliers only.

The input to the RANSAC method is a set of points on the bounding boxes belonging to same line and the output is the baseline approximation that passes through the base point of each of the letters present in that line. Figure 3.5 shows an example. Clearly, the characters such as ‘p’ and ‘y’ fall below the base line. Under normal circumstances, we can assume that all the baselines of text in a page are parallel to each other and hence can be used to obtain horizontal vanishing point. Since, there is an estimate of more than two horizontal lines in the image, the vanishing points from all the possible pairs are calculated and the average of these points is used as an estimate of the horizontal vanishing point. Similarly, the starting and ending points of each line can be used to obtain an estimate of the parallel vertical lines.

Figure 3.5 Baseline Fitting using RANSAC

If the text in a page has multiple columns and is not justified, estimating vertical lines can be difficult. When the text contains more than one column, a multi-column detection method is first applied and then the vertical lines are detected for each column. Given an estimate of the vertical and horizontal vanishing points, the vanishing line and the corresponding rotation matrix can be estimated. Let $H_r$ and $V_r$ represent the horizontal and vertical vanishing points in homogeneous coordinates, $V_z$ represent the equation of the vanishing line and the rotation matrix $M_{rot}$ can be expressed as follows:
\[ M_{Rot} = [H_v, V_v, V_z] \]

where
\[ H_v = [H_x, H_y, 1] \]
\[ V_v = [V_x, V_y, 1] \]
\[ V_z = H_v \otimes H_v \] (3.3)

where \( H_x, H_y, V_x \) and \( V_y \) represent the \( x \) and \( y \) coordinates of the horizontal and vertical vanishing points respectively, and represents the cross product. Figure 3.6 shows the estimated vertical vanishing points.

Figure 3.7 shows the bounding rectangle of the connected components obtained after applying the rotation matrix.
Figure 3.8 shows an example of the original image and the corresponding perspective rectified images when processed with the average, Niblack adaptive, and Otsu binarization techniques. Clearly, the lines of text in a perspective-rectified image appear to be parallel. However, note that the distortion due to page curl is still present and must be removed to improve the performance of the OCR.

3.4 Page Curl Correction

To remove the distortion due to page curl, a line-by-line dewarping of the entire page is performed. The slope of each text line is required to recover each character individually. Thus, the slope is obtained by considering the slope of the baseline of each character as a whole. To obtain a better estimate, only the characters present in each word are used in the calculation. A RANSAC approximation of the base line is then obtained for each word.

To rectify the document image, the characters in each line are rotated with an angle obtained from its baseline. In this way, all of the characters present in a single line will appear in the same line in the transformed image. Note that the start of each line in the original image is maintained in the rectified image. This enables an approximate indentation of the paragraphs and text from the original image to be preserved.

Figure 3.9 shows an example of the original image and the corresponding dewarped images when processed with the average, Niblack adaptive, and Otsu binarization techniques. As can be seen, the proposed method is able to maintain the characters in the same line as in the original image and also remove the distortion due to page curl. Note that at the present stage, special characters such as quotes, hyphens, etc. present in the document image are ignored.
3.5 Page Segmentation

To facilitate the find-ability and spatial understanding for a visually impaired reader, the section headings of a document are identified. When headings can be identified, they can then be read to the user rather than the whole document image as is standard for other reading devices. Support for the identification of headings and other framing techniques begin with the segmentation of a page.

To reduce the information required to process a document image, a pyramidal scheme has been implemented using a pyramid data structure and the Zeta Scanning Algorithm [Bourbakis and Klinger, 1989]. This methodology iteratively reduces the size of an image through an averaging of groups of pixels within the image. Consider that the original image is of size \(N^2 \times (X \times Y)\). After each iteration through the Zeta Scanning Algorithm the image is reduced to a size of \((4N^2-1)/3\). This reduction method is used to produce a low-resolution representation of the structure of the document image.

In a revised approach to this pyramidal reduction scheme every level of the image is split into non-overlapping kernels of 2x2 pixels. A lower-resolution image is produced by correlating every kernel to only one pixel in the reduced image, thus reducing the resolution of the new image by 2 in every dimension. This reduction also included the rule that if one or more pixels of the kernel are black (i.e. text or image) then the corresponding pixel in the reduced image will also be black.

As shown in figure 3.10, headline and paragraph segments are identifiable in the 128x128 image that was created using the revised pyramid method. This reduction method is compared with images reduced using a linear interpolation [Bradski and Kaehler, 2008] method and a nearest neighbor [Lu and Tan, 2003] method, which tend to eliminate structural information from the image rather than preserve it.

While the headline and paragraph segments may be identifiable, pixels may be set to black that should be set to white due to residual effects of the earlier document image-processing methods. For example, the perspective correction method may straighten the document enough for OCR, but not enough to fully support the identification of paragraph and headline boundaries. Thus, a boundary acceleration method is applied in order to ensure that block boundaries are clean and clearly identifiable to the boundary discovery algorithm.
The boundary acceleration method considers the length of a potential horizontal or vertical boundary and searches in the same row/column for potential boundary pixels by evaluating the distance from the potential pixel to the confirmed boundary. If this distance falls within a configurable threshold, the boundary is extended to include the identified pixel. In this way legitimate boundaries are expanded and the boundary discovery process can proceed. In Figure 3.11 the result of the boundary acceleration method is presented. The green lines indicate confirmed vertical boundaries and the red lines show where the vertical boundary was extended. The yellow lines indicate the confirmed horizontal boundaries and the purple lines show where the horizontal boundary was extended.
The boundary discovery process selects a starting pixel and tracks the boundary until it comes back to the starting pixel. Special processing must be performed for the positions where the boundary falls on the edge of the document image. Pseudo-code for the boundary following routine is shown in Figure 3.12.

```plaintext
set startPosition
set currentPosition = startPosition
set b = 4-neighbor to the west of startPosition
while (currentPosition != startPosition) {
    let currentPosition = 8-neighbor on boundary starting search with b
    let b = 4-neighbor pixel counter-clockwise from currentPosition
    if (b is on imageEdge)
        search clockwise for a b that is not on the imageEdge

    leftSide = (leftSide < current.x) ? leftSide : current.x;
    rightSide = (rightSide > current.x) ? rightSide : current.x;
    top = (top < current.y) ? top : current.y;
    bottom = (bottom > current.y) ? bottom : current.y;
}
```

After the boundary discovery process has completed on the reduced image, the list of boundaries is mapped back to the original image. The text inside these boundaries will be further analyzed in the headline identification process described in the next section. Figure 3.13 shows the original image and the
corresponding segments when processed with the average, Niblack adaptive, and Otsu binarization techniques.

Figure 3.13 Region Boundaries using Average, Niblack’s Adaptive, and Otsu Binarization Methods

3.6 Headline Identification

Once the boundaries around regions of interest have been identified, the headlines and supporting text found in a document image can be detected. Three important features of a document image are used in this discovery process: the headlines, the text itself, and the spacing used to separate one portion of text from the next. Black portions of the textual image are framed, and the portions are grouped together based on proximity. Headlines are identified by the font size and position relative to identified portion of text.

The first step in this process however, is to identify image regions and textual regions. To accomplish this, each identified segment of the rectified image is passed through an OCR. If the OCR identifies the region as text (regardless of font size) the region is tagged as TEXT. The regions identified as image are tagged as IMAGE. Some regions cannot be identified as text or image, and these are tagged as OTHER. In Figure 3.14 an example is presented with textual regions outlined in red, and image regions outlined in purple.

Figure 3.14 Separated Region Boundaries

Once the textual regions have been identified, the headline classification process labels the segments as headlines or body text. This process begins with the identification of the bounding rectangles of each character within a textual segment. The height of the tallest bounding rectangle is noted for each segment.
These heights (or font sizes) are gathered into a histogram and features of the histogram are analyzed in order to determine which segments should be considered headlines and which should be considered as body text. Figure 3.15 contains the histogram for the regions identified in Figure 3.14. The zero size regions are those identified as IMAGE, and the green vertical line indicates where the headline regions begin.

![Figure 3.15 Histogram of Font Sizes in a Document Image](image)

The analysis of the histogram involves a search of the values that will discover an ideal candidate font size discriminator. This search includes several empirically derived rules, which are outlined below. A potential headline marker is evaluated against all of the rules until an ideal candidate is discovered. The ideal candidate is the discriminator with the lowest value that meets all of the criteria. Note that this technique assumes that the images are 640 x 480 pixels.

- There must be more than two textual segments in the document image in order to identify a segment as a headline.
- Font size must be greater than 10 pixels.
- For a textual region to be identified as a headline in which the maximum font size is greater than 10 and less than 15 pixels, 70% of the textual regions must be smaller than the potential discriminator.
- For a textual region to be identified as a headline in which the maximum font size is between 15 and 20 pixels, 50% of the textual regions must be smaller than the potential discriminator.
- Identify a sequence of maximum font sizes that that are “close” together (i.e. differ by less than 2 pixels). Place the potential discriminator one value lower than the lowest value in the sequence.
- For the values in the histogram that are greater than 20 and do not fall in a sequence, identify the distance between the value and its nearest neighbor. If the distance between the neighbors on the left is
greater than the distance to the neighbor on the right then place the potential discriminator just to the
left of the value.

Figure 3.16 shows the original image and the corresponding headline identification when processed
with the average, Niblack adaptive, and Otsu binarization techniques. The headline regions are identified
with a green bounding rectangle. The textual regions are identified with a red bounding rectangle, and the
unknown regions are identified with a yellow bounding rectangle.

Figure 3.16 Headline Identification using Average, Niblack’s Adaptive, and Otsu Binarization Methods

3.7 Segment Aggregation

The final step in the image processing pipeline utilizes the information discovered in the headline
classification step to associate body text segments with the appropriate headline segments and create an
aggregated group of segments called an article. The classified textual regions are processed by a regional
synthesis method that combines body segments and headline segments based on their regional proximity.
Related body segments are closer to each other vertically than the related horizontal body segments, which
are closer than other non-related body segments. Thus, an initial set of textual regions is examined for
regions within a close vertical proximity. These regions are combined to form a first level synthesized
region, or a column of text. After the columns have been synthesized, a similar process is used to join
regions horizontally. However, the horizontal regions must be of similar height in order to be joined. A
similar process is used to aggregate the headline segments.

After the textual segments and the headline segments have been joined, a process for joining the two
segment types into articles can begin. Beginning with the bottom most textual region on the document
image, the corresponding headline is discovered by examining the headline regions above it. If none are
found, the region is assumed to not have a headline. When a headline is found above the textual region,
they are matched based on proximity, namely the closest headline is matched with the textual region. There
are incidents when this rule is incomplete, and thus the closest headline must still fall within a ‘reasonable’
distance from the textual region. ‘Reasonable’ here is empirically defined to be within 10% of the textual
region’s height. Much of this work is derived from the application to newspapers, and becomes highly simplified when applied to document images from a book.

After the regions have been identified and textual regions synthesized into articles, the textual regions are framed. A frame is created around a region by examining the pixels surrounding a region to ensure that they contain whitespace. The outermost whitespace locations are marked as the frame of the region. These framed regions and their corresponding headline and textual regions form the basis of the navigation of the document for the blind reader. This process will be developed further in Chapter 4.

Figure 3.17 shows the original image and the corresponding article identification when processed with the average, Niblack adaptive, and Otsu binarization techniques. The frame is bound with a blue rectangle. The headline and textual segments are also identified with the green and red bounding rectangles. Pseudo-code for the synthesis routine is presented in Figure 3.18.
get listOfClassifiedSegments
set headlineList = getHeadlineList(listOfClassifiedSegments)
set textualList = getTextualList(listOfClassifiedSegments)

for each headlineSegment in headlineList
    group headlineSegments with close vertical proximity
    group headlineSegments with close horizontal proximity and similar size

for each newHeadlineSegment
    create article
    set article->headline to headlineSegment

for each textualSegment in textualList
    group textualSegments with close vertical proximity
    group textualSegments with close horizontal proximity and similar size

for each newTextualSegment
    find corresponding headlineSegment
    add textualSegment to article->textList

for each article
    validate article boundaries

Figure 3.18 Method to Frame Headline and Textual Regions

3.8 Experimental Results

In order to evaluate the entire document-image processing pipeline, 25 images were captured using the TYFLOS cameras. These cameras capture images with a resolution of 640 x 480. While camera technology continues to improve over time, and image-processing techniques perform better with high-resolution images, it is also noteworthy that the performance of this image-processing pipeline will improve when images with greater resolution are used.

Twenty-five document images were taken of the September 4, 2010 edition of the Wall Street Journal and the New York Times. The papers were held at various angles and distances in order to test the performance of the TYFLOS correction techniques. An artificial way to simply improve the reported performance of the system would be to capture more document images closer to the cameras. The goal of this study was to measure the improvement in OCR accuracy on a set of images, and thus the document images represent a variety of distances and distortions.
The first stage of the evaluation, illustrated in Figure 3.19, includes the steps in the document image correction phase; the second stage includes the steps in the document image analysis phase. These stages were evaluated independently.

![Diagram](image.png)

*Figure 3.19 Document Image Correction Evaluation Steps*

The perspective and page curl correction methods described above do not work with document images that contain a picture or image. Thus, the first pre-processing step in this experiment was to run all 25 document-images through OmniPage-16, a commercial OCR that can determine whether a portion of the page contains an image. Based on the results of the OmniPage analysis images were removed from the document images.

The first step in the experiment was to binarize the 25 preprocessed images. These images were all binarized using the three methods described in Section 3.1. Each image was then evaluated for OCR accuracy with OmniPage-16. These results provide a baseline for the performance evaluation of the remaining steps in the process. The average accuracy based on the three different forms of binarization alone were 44.8% for the average method, 52.8% for the Niblack adaptive method, and 24.5% for the Otsu method.

The enhancement techniques described in Section 3.2 combine the document image enhancement with the binarization process, and thus cannot be treated as a separate step in the process. The document image enhancement was performed with the three different binarization techniques as part of the process, and treated individually for evaluation purposes. Each image was then evaluated for OCR accuracy with OmniPage-16. These results provide a baseline for the performance evaluation of the remaining steps in the
pipeline, but can also be compared to the binarization process. The average accuracy based on the three different forms of binarization alone were 15.5% for the average method, 56.2% for the Niblack adaptive method, and 60.3% for the Otsu method. Thus, the enhancement had a notable detrimental effect on the average method and a notable positive effect on the Otsu method.

The next step in the document image correction stage of the pipeline is the perspective correction. The 75 images that were binarized without enhancement and the 75 images that were processed with the document image enhancement technique were corrected with the perspective correction method described in Section 3.3. Each image was then evaluated for OCR accuracy with OmniPage-16. Changes in the OCR accuracy are noted when these results are compared to the binarization process and the enhanced process. The perspective correction based on the three different forms of binarization alone had a detrimental effect on the OCR accuracy, going down to 15.7% for the average method, 19.0% for the Niblack adaptive method, and 20.8% for the Otsu method. However, the perspective correction had a positive effect on the OCR accuracy for the enhanced document images: 17.4% for the average method, 63.6% for the Niblack adaptive method, and 79.3% for the Otsu method.

The final step in the document image correction portion of the pipeline is the page curl dewarping. The 75 images that were binarized and perspective corrected and the 75 enhanced images that were perspective corrected were dewarped with the method described in Section 3.4. Each image was then evaluated for OCR accuracy with OmniPage-16. Changes in the OCR accuracy are noted when these results are compared to the binarization process and the enhanced process. The page curl correction based on the three different forms of binarization alone continued to have detrimental effect on the OCR accuracy, going down even further to 9.2% for the average method, 18.5% for the Niblack adaptive method, and 17.8% for the Otsu method. However, the page curl correction had a continued positive effect on the OCR accuracy for the enhanced document images: 20.0% for the average method, 71.2% for the Niblack adaptive method, and 86.4% for the Otsu method.

Figures 3.20 and 3.21 present a comprehensive view of the OCR accuracy results for the various points on the pipeline. Figure 3.20 presents the accuracy measures for the headlines in the document image. Figure 3.21 presents the accuracy measures for the body text in the document image. Note that the left side of each figure contains the results for the binarization only path through the pipeline and the right side presents the results for the enhancement path through the pipeline.
These experimental results were analyzed using analysis of variance (ANOVA), with an \( \alpha \) value of 0.05. Each step in the processing pipeline had a statistically significant change with \( p < 0.0001 \) for both the Headlines and the Body text. Further, a Tukey-Kramer Honestly Significant Difference (HSD) reveals that the application of the perspective correction and page curl correction significantly improve the OCR accuracy of the document image. The full results of this analysis are presented in Appendix C.

Once the document image has been corrected it continues to pass through the rest of the document image-processing pipeline in order to be converted into an XML document. This portion of the process includes the segmentation, separation, classification and aggregation steps described in Sections 3.5 - 3.7.
The evaluation of the layout analysis stage of the document image-processing pipeline is illustrated in Figure 3.22.

While the document image segmentation and separation steps were not evaluated for accuracy, the headline classification and segment aggregation steps were. In order to perform this evaluation, the 75 enhanced document images that had been fully corrected by the corrective processes were passed through the latter stage of the pipeline.

This second stage begins with the document image segmentation process described in section 3.5. For evaluation purposes, the identified segments were then classified as TEXT, IMAGE or OTHER by using both a manual process and an automatic process (using OmniPage-16). The manual classification presented each document image segment to a human evaluator who classified the segment. For automatic classification, each segment was evaluated by OmniPage-16. These two evaluations resulted in 150 classified images that were then evaluated by the headline classifier.

In order to evaluate the headline classifier, the 75 images that had been separated manually and the 75 images that had been separated automatically passed through a manual and automatic classification process. The 150 separated images were presented to a human classifier who classified each TEXT segment as a headline or body text. The 150 separated images were also presented to the automatic classifier described in Section 3.6. In this way, 300 images were classified, the results of which are presented in the Receiver Operating Characteristic (ROC) curve in Figure 3.23.
A ROC curve is a plot of the true positive rate vs. the false positive rate for a binary classifier. In this case, the classifier is binary in that it determines whether a TEXT segment is a headline or not. The dotted line through the middle of the graph represents the performance of a random classifier. Results that are closer to the X-axis or Y-axis are considered farther from random.

As is demonstrated in Figure 3.23, the headline classifier described in Section 3.6 performs well. The two points of interest that are closer to random than the other clustering of points represent the values for the Manually Separated/Automatic Classified and Automatic Separated/Automatic Classified images that were processed with the average binarization technique. Table 3.1 presents a full analysis of the confusion matrix represented by the ROC curve in Figure 3.23, including the True Positive Rate (TPR), False Positive Rate (FPR), Accuracy (Acc), Positive Predictive Value (PPV), Negative Predictive Value (NPV), and the False Discovery Rate (FDR).

Figure 3.23 ROC Curve of Headline Classifier Results
The segmentation aggregation step described in Section 3.7 was used to aggregate the 300 classified images produced from the classification step. The results of the aggregation step are presented in the ROC curve in Figure 3.24. Evaluation of this curve reveals that the documents processed by the Otsu binarization technique (the four markers closest to the y-axis) perform better than those processed by the average binarization technique (the four markers closest to random). Table 3.2 presents a full analysis of the confusion matrix represented by the ROC curve in Figure 3.24.

![Segment Aggregation ROC Curve](image)

**Table 3.1 Confusion Matrix Results for the Headline Classifier**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TPR</td>
<td>0.85</td>
<td>0.96</td>
<td>0.98</td>
<td>0.54</td>
<td>0.58</td>
<td>0.79</td>
<td>0.69</td>
<td>0.74</td>
<td>0.98</td>
</tr>
<tr>
<td>FPR</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.18</td>
<td>0.02</td>
<td>0.04</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
</tr>
<tr>
<td>Acc</td>
<td>0.93</td>
<td>0.99</td>
<td>0.99</td>
<td>0.69</td>
<td>0.83</td>
<td>0.88</td>
<td>0.85</td>
<td>0.91</td>
<td>0.98</td>
</tr>
<tr>
<td>SPC</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.82</td>
<td>0.98</td>
<td>0.96</td>
<td>1.00</td>
<td>1.00</td>
<td>0.99</td>
</tr>
<tr>
<td>PPV</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.73</td>
<td>0.95</td>
<td>0.95</td>
<td>1.00</td>
<td>1.00</td>
<td>0.99</td>
</tr>
<tr>
<td>NPV</td>
<td>0.87</td>
<td>0.98</td>
<td>0.98</td>
<td>0.67</td>
<td>0.80</td>
<td>0.83</td>
<td>0.76</td>
<td>0.88</td>
<td>0.98</td>
</tr>
<tr>
<td>FDR</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.27</td>
<td>0.06</td>
<td>0.05</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
</tr>
</tbody>
</table>

**Figure 3.24 ROC Curve of the Segmentation Aggregation Results**
<table>
<thead>
<tr>
<th></th>
<th>Man Sep / Man Class</th>
<th>Man Sep / Auto Class</th>
<th>Auto Sep / Man Class</th>
<th>Auto Sep / Auto Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPR</td>
<td>0.81</td>
<td>0.92</td>
<td>0.96</td>
<td>0.81</td>
</tr>
<tr>
<td>FPR</td>
<td>0.53</td>
<td>0.35</td>
<td>0.12</td>
<td>0.44</td>
</tr>
<tr>
<td>Acc</td>
<td>0.75</td>
<td>0.86</td>
<td>0.94</td>
<td>0.76</td>
</tr>
<tr>
<td>SPC</td>
<td>0.47</td>
<td>0.65</td>
<td>0.88</td>
<td>0.56</td>
</tr>
<tr>
<td>PPV</td>
<td>0.86</td>
<td>0.90</td>
<td>0.97</td>
<td>0.89</td>
</tr>
<tr>
<td>NPV</td>
<td>0.38</td>
<td>0.71</td>
<td>0.85</td>
<td>0.40</td>
</tr>
<tr>
<td>FDR</td>
<td>0.14</td>
<td>0.10</td>
<td>0.03</td>
<td>0.11</td>
</tr>
</tbody>
</table>

Table 3.2 Confusion Matrix Results for the Segmentation Aggregation Classifier

3.9 Conclusion

In this chapter a document image-processing pipeline was presented that performs document image correction and page layout analysis. The correction stage of the pipeline improves the OCR accuracy of 25 documents from a low of 40% to a high of 85% for headlines, and from a low of 5% to a high of 35% for the body text of document images. The automatic classification techniques in the page layout analysis stage of the pipeline are shown to perform as well as a human classifier.

While this chapter presents the methods required to process a document image, these methods by themselves do create a usable textual document of the image. Chapter 4 describes the remaining steps required to take the OCR and layout information received from this pipeline and construct an XML document that can be utilized in a voice user interface.
Chapter 4 Creating an Interactive Document

The document image processing pipeline described in the previous chapter will convert a document image into ASCII text, classify segments as image, headline, and body text, and aggregate related segments into articles. However, simply performing this process on one document image may not produce a reliable version of the text of the article. This is due to the correlation between OCR Accuracy and the distance between the camera and the document when the image is captured. Thus, an interactive scheme has been devised to improve the accuracy of the OCR and create a document that a user can interact with using a Voice User Interface (VUI). This scheme is illustrated in Figure 4.1.

**Figure 4.1 Creation of an Interactive Document**

At the conclusion of the segment aggregation process described in Chapter 3, the text found in the document image is stored along with positional information. The positional information can then be used to correlate one document image with another in order to improve the interaction with the document. Thus, the first step in the creation of an interactive document is to create an XML representation of the document image (Figure 4.1a). Step two is to collect a set of images and their XML representations (Figure 4.1b) from different views of the same document. Step three is to calculate an image-based registration measure, which will be used to guide the combining of multiple XML files into a composite comprised of the best information from all available related images. The fourth step in the process involves regular expression
matching of the segments with a high probability of being part of the same article. If the regular expression matching process discovers a correlation between two XML files, the two are combined into a composite XML file, which is used for interaction with the user (Figure 4.1c). As more document images are captured, the composite interaction file grows in accuracy.

This chapter describes each of these steps in detail. However, before the details of each step are presented, Section 4.1 discusses the need for this process and demonstrates that the distance between the document and the cameras affects OCR accuracy. Section 4.2 describes the process of converting the results of the document image pipeline described in Chapter 3 into an XML document. This is followed by the presentation of a targeted registration process in Section 4.3, which attempts to find related document segments based on the image characteristics of each segment. Section 4.4 presents the final step in the process, which is to perform regular expression matching on potentially related segments.

4.1 OCR at a Distance

The cameras incorporated into the TYFLOS system are inexpensive web-cams that capture 640 x 480 pixel resolution images. These cameras must be calibrated with a fixed focal length, so in order to determine the proper distance to calibrate the cameras, 14 sighted people (7 male, 7 female) were surveyed. The subjects were presented with a newspaper and asked to sit in a comfortable position to read it (see Figure 4.2). The distance of the newspaper to the subject’s eyes was measured, as well as the arm length (distance from the top of the shoulder to the wrist), and their height. Table 4.1 presents the findings from this survey.

![Figure 4.2 Sighted Readers Sitting in a Comfortable Position to Read a Newspaper](image)

These findings indicate that an optimal focal length for the TYFLOS cameras is 43 cm. An experiment was designed to test whether this setting would provide an optimal performance of an OCR system. For this experiment, a rack was assembled to ensure a flat surface and an accurate distance from the camera (see Figure 4.3). Two camera systems were used to capture the document images: the cameras of the TYFLOS
prototype and a Fuji FinePix 602s. The focal lengths of both camera systems were set to 20 cm and 43 cm, and document images were captured at 20, 30, 40, and 50 cm away from the camera for each focal length setting.

<table>
<thead>
<tr>
<th>Gender</th>
<th>Height</th>
<th>Eye-Paper Distance</th>
<th>Arm Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>181 cm</td>
<td>46 cm</td>
<td>53 cm</td>
</tr>
<tr>
<td>Female</td>
<td>165 cm</td>
<td>40 cm</td>
<td>46 cm</td>
</tr>
<tr>
<td>Average</td>
<td>173 cm</td>
<td>43 cm</td>
<td>50 cm</td>
</tr>
</tbody>
</table>

*Table 4.1 Average Metrics of Sighted Newspaper Readers*

*Figure 4.3 Rack to Accurately Capture Document Images*

*OmniPage-16* was used to perform the OCR processing. OmniPage-16 is tuned to perform OCR on document images that have been captured using a flatbed scanner. When images are captured this way, *OmniPage-16* will produce results with up to 100% accuracy for a multicolumn newspaper document image.

As part of the OCR process, *OmniPage-16* applies image enhancement algorithms. It will also segment the document image and categorize these segments as textual or image segments. It is important to note the accuracy of the *OmniPage-16* document image segmentation process, as well as the accurate recognition of headline characters and article text characters in order to compare with the accuracy of the TYFLOS reader.

The results of these tests are shown in Tables 4.2-4.5. These results indicate that headlines may be processed accurately at distances less than 30 cm away from the camera; however, the body text of the
article will need to be captured with the document much closer to the camera. Thus, the constraint of utilizing the low resolution images requires a user interaction scheme to be devised that will facilitate the capture of portions of the document that are of interest to the user. An image registration process must also be devised that will ensure that images captured at different distances are of the same area on the document.

<table>
<thead>
<tr>
<th>Camera &amp; Focal Length</th>
<th>Segment Accuracy</th>
<th>Headline Accuracy</th>
<th>Body Text Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuji: 20 cm</td>
<td>44%</td>
<td>100%</td>
<td>86%</td>
</tr>
<tr>
<td>Fuji: 43 cm</td>
<td>100%</td>
<td>100%</td>
<td>1%</td>
</tr>
<tr>
<td>TYFLOS: 20 cm</td>
<td>50%</td>
<td>100%</td>
<td>74%</td>
</tr>
<tr>
<td>TYFLOS: 43 cm</td>
<td>100%</td>
<td>100%</td>
<td>3%</td>
</tr>
</tbody>
</table>

*Table 4.2 OCR Accuracy when Document is 20 cm From the Camera*

<table>
<thead>
<tr>
<th>Camera &amp; Focal Length</th>
<th>Segment Accuracy</th>
<th>Headline Accuracy</th>
<th>Body Text Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuji: 20 cm</td>
<td>0%</td>
<td>64%</td>
<td>2%</td>
</tr>
<tr>
<td>Fuji: 43 cm</td>
<td>71%</td>
<td>99%</td>
<td>3%</td>
</tr>
<tr>
<td>TYFLOS: 20 cm</td>
<td>50%</td>
<td>96%</td>
<td>3%</td>
</tr>
<tr>
<td>TYFLOS: 43 cm</td>
<td>91%</td>
<td>99%</td>
<td>5%</td>
</tr>
</tbody>
</table>

*Table 4.3 OCR Accuracy when Document is 30 cm From the Camera*

<table>
<thead>
<tr>
<th>Camera &amp; Focal Length</th>
<th>Segment Accuracy</th>
<th>Headline Accuracy</th>
<th>Body Text Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuji: 20 cm</td>
<td>6%</td>
<td>36%</td>
<td>1%</td>
</tr>
<tr>
<td>Fuji: 43 cm</td>
<td>13%</td>
<td>60%</td>
<td>2%</td>
</tr>
<tr>
<td>TYFLOS: 20 cm</td>
<td>19%</td>
<td>63%</td>
<td>2%</td>
</tr>
<tr>
<td>TYFLOS: 43 cm</td>
<td>20%</td>
<td>69%</td>
<td>2%</td>
</tr>
</tbody>
</table>

*Table 4.4 OCR Accuracy when Document is 40 cm From the Camera*

<table>
<thead>
<tr>
<th>Camera &amp; Focal Length</th>
<th>Segment Accuracy</th>
<th>Headline Accuracy</th>
<th>Body Text Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuji: 20 cm</td>
<td>5%</td>
<td>13%</td>
<td>0%</td>
</tr>
<tr>
<td>Fuji: 43 cm</td>
<td>11%</td>
<td>2%</td>
<td>0%</td>
</tr>
<tr>
<td>TYFLOS: 20 cm</td>
<td>5%</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td>TYFLOS: 43 cm</td>
<td>11%</td>
<td>1%</td>
<td>0%</td>
</tr>
</tbody>
</table>

*Table 4.5 OCR Accuracy when Document is 50 cm From the Camera*
The interaction scheme (described in Chapter 5) enables a user to capture a document image that is up to 30 cm from the camera and TYFLOS will read the headlines to the user. The user can then request a specific article to be read and TYFLOS will guide the user to properly position the article in front of the cameras in order to accurately process and read the requested article. The image registration process required to guide the user is described in the next section.

4.2 Conversion of an Aggregated Image to XML

Recall that the goal of converting a document image into XML is to enable the combination of representative XML files into one composite XML file for a specific article. This composite XML file is then used for the voice interaction described in Chapter 5 and Chapter 6. The first step in this process is to convert a document image into XML. This conversion is performed by the segment aggregation step in the document image-processing pipeline described in chapter 3. A schema of the XML file is presented in Figure 4.4.

The schema contains a few elements, namely the DocumentImage element, which is comprised of an Article element, which is comprised of a Headline element and a BodyText element.

The DocumentImage element in the schema is the outer-most element and contains a document number, the page number that the image was captured on, and one or more Article elements. The document number is a random number between 0 and 2,147,483,647 (long type in C++), and is used to uniquely identify the document image that created the XML file. While it is possible for two document images to be assigned the same document number it is unlikely given that only a few images will be taken of a given page. If this were to prove problematic, this number could easily be changed to use a Globally Unique IDentifier (GUID). The document number, combined with the page number, facilitates the regular expression comparison and the creation of the composite XML described in Section 4.4.

To populate the page number element, the user is prompted to move the paper so that the upper left or upper right corner is in the camera view. The document image is captured and processed, and the segment found at the highest and furthest left/right position on the image is examined for the number. If no number is found the user is prompted again until a number is found. This number is then recorded as the page number for the document.
Figure 4.4a Document Image Schema
The primary element in the `DocumentImage` is the `Article` element, which contains values for the article’s height, width, number of text columns that comprise the article, and the average distance between them. These values play an important role in the registration process described in Section 4.3. The `Article` element also contains a `Headline` element and one or more `BodyText` elements.

A `Headline` element contains the maximum character height in the headline, the average character height in the headline, and the position within the document image that the headline is located. It is this position that is used to guide the user in the capture of closer more readable images of the article (described in Chapter 5). The position is comprised of values for the top, bottom, right, and left of the bounding rectangle surrounding the headline. The `Headline` element also contains the ASCII text of the headline.
The **BodyText** element essentially contains the same information as the **Headline** element. However, the information is segmented by column in order to preserve the visual structure of the document. Thus, the maximum character height, average character height, position, and number of spelling errors of each column that comprise the **BodyText** are preserved. An additional element is the text element that breaks the columns into paragraphs and sentences. This information is important to provide the document navigation at the paragraph and sentence level that is described in Chapter 5.

All of the information preserved in the XML for one document image can be used in the comparison of one document image to another. This comparison process then leads to a combination of XML documents in order to create a composite XML representation of the best information available from the sampled document images. The next two sections describe the processes used to form this composite XML.

### 4.3 Targeted Registration

A primary directive for the TYFLOS system identified in Chapter 1 is to provide spatial information to an eyes-free user. In order to guide the user in the interaction with the spatial components of the document, multiple images of the page must be captured, and TYFLOS must track the location of these images relative to each other. This section presents a targeted registration process that supports this activity.

When a document image is captured at a distance it can provide an overview of the page, but a second document image must be processed and registered with the first in order to guide the user to properly position the document relative to the cameras. This type of registration process typically involves an analysis of both the reference and target images in order to detect notable features, match these features with invariant descriptors, and transform the target image based on the correspondence between the two images [Niblack, 1986]. However, by exploiting features identified during the segmentation and aggregation process, optimizations in the matching and registration phase of registration can be implemented. The first step in this process is to calculate the similarity between the article frames of two document images using attributes obtained during the segmentation and aggregation process; secondly, the article frame boundary can be utilized as the invariant descriptor in the correlation calculation. All of the information required to perform these steps is stored in the XML document created by the Segmentation Aggregation step of the Document Image Processing Pipeline.
For clarity, the first image captured, assumed to be further from the cameras (approximately 30 cm away), is referred to as the reference image. The second captured image, assumed to be closer to the camera, is referred to as the target image. The goal of the targeted registration process is to map the target image onto the reference image. In doing so, the relative spatial information can be mapped and preserved.

Since the target image goes through the segmentation and framing process, the attributes of the frames identified within the target image can be compared with the attributes of the reference image. Each of the frames within the reference image are then scored and ranked based on similarity of attributes. Regular expression matching and correlation calculations are then limited to the frames that are more likely to contain the target image. The ranking is based on similarity measures which include the difference between the ratios of the height and width of each article-frame, the difference between the number of columns found in an article-frame, and the difference of the average distance between constituent frames. In order to calculate a composite similarity score for each frame in the target image relative to the reference image, the height/width ratio is multiplied by a combination of the other equally weighted similarity measures. Thus, the similarity measure \( S_{(i,j)} \) is calculated as:

\[
S_{(i,j)} = s_{1,i,j} \cdot (s_{2,i,j} + s_{3,i,j})
\]

\[\text{where:}\]

\[
s_{1,i,j} = \text{abs}\left(\frac{\text{refHt}}{\text{refWd}} - \frac{\text{tgtHt}}{\text{tgtWd}}\right)
\]

\[
s_{2,i,j} = \text{abs}(\text{refNoCols} - \text{tgtNoCols})
\]

\[
s_{3,i,j} = \text{abs}(\text{refAvgDist} - \text{tgtAvgDist})
\]

(4.1)

An example of a similarity measure matrix for a pair of images is presented in Table 4.6. Recall that only the textual segments are processed and aggregated into an article-frame, so the images in this example are ignored. The article labeled 2 in the reference image on the left of Figure 4.4 is also labeled 2 in the target image on the right.
The article-frames with the lowest similarity score (greater than zero) are determined to be the most similar. Thus, based on the similarity scores presented in Table 4.6, the first attempt at document registration for Figure 4.4 is to register article-frame 2 from the target image with article-frame 2 of the reference image.

<table>
<thead>
<tr>
<th></th>
<th>Ref Image 1</th>
<th>Ref Image 2</th>
<th>Ref Image 3</th>
<th>Ref Image 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target 1</td>
<td>0.00</td>
<td>15.30</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Target 2</td>
<td>10.03</td>
<td>0.16</td>
<td>51.40</td>
<td>51.40</td>
</tr>
<tr>
<td>Target 3</td>
<td>0.00</td>
<td>34.37</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Target 4</td>
<td>186.95</td>
<td>65.08</td>
<td>149.72</td>
<td>149.72</td>
</tr>
</tbody>
</table>

*Table 4.6 Example Similarity Measure Matrix for Figure 4.4*

Figure 4.5 presents another example in which the target image is not aligned well with the reference image. In this example, article-frame 2 in the reference image on the left is article-frame 2 in the target image on the right. The similarity measures presented in Table 4.7 indicate that these two article-frames are most similar and should be examined first in the regular expression matching process.

<table>
<thead>
<tr>
<th></th>
<th>Ref Image 1</th>
<th>Ref Image 2</th>
<th>Ref Image 3</th>
<th>Ref Image 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target 1</td>
<td>0.00</td>
<td>14.55</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Target 2</td>
<td>10.78</td>
<td>0.31</td>
<td>52.15</td>
<td>52.15</td>
</tr>
<tr>
<td>Target 3</td>
<td>0.41</td>
<td>2.63</td>
<td>37.64</td>
<td>37.64</td>
</tr>
<tr>
<td>Target 4</td>
<td>0.00</td>
<td>26.19</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

*Table 4.7 Example Similarity Measure Matrix for Figure 4.6*
Note that article-frames captured from unrelated documents may have visually similar characteristics. Thus, the similarity measure calculated between two document images is used only as a guide to direct the regular expression matching described in the next section, and should not be treated as a direct classifier.

In order to examine the effectiveness of the targeted registration method, 17 of the original 25 images used to evaluate the document image pipeline in Chapter 3 were compared. Only those document images that contained a portion of the same newspaper article were used in this evaluation. Images that had been processed by the Segment Aggregation method were compared. This led to a total of 288 different comparisons: 24 relevant comparisons of document images, each of which had been binarized using three different binarization methods, and the segment aggregation based on four different separation/classification schemes as described in Chapter 3.8. Figure 4.5 presents the results of this targeted registration method as a ROC curve.

![Targeted Registration ROC Curve](image)

*Figure 4.7 ROC Curve of the Targeted Registration Results*

As is expected, the results of the targeted registration are highly dependent on the accuracy of the segmentation aggregation method. Table 4.8 presents a full analysis of the confusion matrix represented by the ROC curve in Figure 4.5, including the True Positive Rate (TPR), False Positive Rate (FPR), Accuracy (Acc), Positive Predictive Value (PPV), Negative Predictive Value (NPV), and the False Discovery Rate (FDR). The Otsu binarization method provides the best clustering of results, with little difference between
manual and automatic processing. This indicates that the process performs well given good results from the
document image processes preceding it.

<table>
<thead>
<tr>
<th></th>
<th>Man Sep / Man Class</th>
<th>Man Sep / Auto Class</th>
<th>Auto Sep / Man Class</th>
<th>Auto Sep / Auto Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPR</td>
<td>0.48</td>
<td>0.58</td>
<td>0.75</td>
<td>0.43</td>
</tr>
<tr>
<td>FPR</td>
<td>0.32</td>
<td>0.14</td>
<td>0.06</td>
<td>0.31</td>
</tr>
<tr>
<td>Acc</td>
<td>0.62</td>
<td>0.79</td>
<td>0.90</td>
<td>0.61</td>
</tr>
<tr>
<td>SPC</td>
<td>0.68</td>
<td>0.86</td>
<td>0.94</td>
<td>0.69</td>
</tr>
<tr>
<td>PPV</td>
<td>0.42</td>
<td>0.58</td>
<td>0.75</td>
<td>0.38</td>
</tr>
<tr>
<td>NPV</td>
<td>0.73</td>
<td>0.86</td>
<td>0.94</td>
<td>0.74</td>
</tr>
<tr>
<td>FDR</td>
<td>0.58</td>
<td>0.42</td>
<td>0.25</td>
<td>0.63</td>
</tr>
</tbody>
</table>

Table 4.8 Confusion Matrix Results for the Similarity Measure

It is important that a similarity measure based on physical characteristics of the document image is
used in determining which XML files to join into a composite because the results from the OCR are often
unusable when the image is captured further from the camera. This is described further in the next section.

4.4 Creation of a Composite XML Document

Recall that the Voice User Interface (VUI) of TYFLOS interacts with a composite XML document
created from one or more document images, and not the document image directly. In this way the quality of
the text that is read aloud can be increased by the capture of a collection of document images and
processing them into the composite document. This section presents the final step in the creation of this
composite XML document.

After the similarity of two images has been calculated, combining the text from the segments that have
been identified to be the most similar creates a composite XML document. This combination process
begins with a brute-force comparison of the text in the segments, and can have one of four results: low
information content with low similarity, low information content with high similarity, high information
content with low similarity, high information content with high similarity. The content value refers to the
quality of the content within a segment, as measured by the number of misspellings in the segment. A text
segment with many misspelled words is considered to contain low information value, whereas a segment
with few misspellings is considered to contain high information value. Similarly, as two segments are
string-wise compared, and segments that contain matching strings of characters are considered to have high similarity, whereas segments with few matching strings are considered to have low similarity.

The different combinations of these characterizations of two documents result in four different methods of creating the composite XML document: Low Information/High Similarity (LIHS), Low Information/Low Similarity (LILS), High Information/High Similarity (HIHS), and High Information/Low Similarity (HILS).

- **LIHS**: Text segments with low information content and high similarity use the text associated with the image with the largest font size. The text associated with the smaller font size is discarded. This process of elimination preserves the information content that has the highest probability of being correct due to the improvement in OCR accuracy when font sizes are larger.

- **LILS**: Text segments with low information content and low similarity are simply concatenated together to form the composite segment. In this way the small bit of information that is available is preserved, yet when this composite is compared with segments from another image any segment with a larger font size will overwrite the segments that have low information content.

- **HIHS**: Text segments with high information content and high similarity typically contain portions of the same text. For these segments the text in each segment is string-wise compared to identify portions of text that are in one segment that are not in the other. Text that is in one and not the other is preserved directly. Portions of text in which a discrepancy of 10 characters or smaller is found, the segment with the largest font size is used to create the composite. Discrepancies of larger than 10 characters are considered to have low similarity, and thus the segments are concatenated together.

- **HILS**: Text segments with high information content and low similarity typically contain different portions of the same segment. Thus, the text from the segments is concatenated together to form the composite segment.

Figures 4.8 - 4.14 illustrate the process with one article. Figure 4.8 presents the fully corrected image followed by an abbreviated version of the XML from the image text produced by the OCR (Figure 4.9). The headline is clear in Figure 4.8, but the text remains unreadable. This is reflected in the XML which contains the correct text for the headline, but the segment contains no information. Similarly, the headlines in Figure 4.10 and Figure 4.12 are distorted due to the lighting. However, the text in each of these images is readable, and the OCR accuracy on each of them is 98%. Thus, by combining the three images, a composite...
XML of the text segments is created (Figure 4.14). This composite is what the user interacts with through the voice user interface described in the next chapter.
London UBS AG and several other investors are taking legal action to force UK fund manager Reade Griffith to speed up the return of their money underscoring how some frustrated investors are resorting to the courts to recoup their cash.

Mr Griffith, whose flagship fund managed $8 billion at its peak a few years ago announced in late 2006 that he would wind up the fund after facing a wave of exits amid steep investment losses. Two years on he has re-capitalised to speed up the return of cash. But earlier this year the Cayman Islands court of appeal ruled in Camulos's favor by dismissing the investor's request.

Some legal experts say that the ruling in Camulos's case was fact specific and isn't necessarily indicative of how petitions against other funds, such as Polygon may fare. A Polygon spokesman declined to comment on the specifics of the petition but said "we believe the case has no merit and will fight it vigorously."
es to Return Cash

Capital LP to speed up the return of cash. But earlier this year, the Cayman Islands court of appeal ruled in Camulos's favor by dismissing the investor's request.

Some legal experts say that the ruling in Camulos's case was fact specific and isn't necessarily indicative of how petitions against other funds, such as Polygon, may fare. A Polygon spokesman declined to comment on the specifics of the petition but said "we believe the case has no merit and will fight it vigorously."

During the course of 2008, a year in which the flagship fund lost 48% of its value due to investment losses. Polygon's management redeemed the vast majority of their shareholding in the fund, which had represented $103.8 million, according to the petition.

Petition's Claims

"It appears that the directors have allowed management..."
Griffith Pressured to Return Cash

London UBS AG and several other investors are taking legal action to force UK fund manager Read Griffith to speed up the return of their money, underscoring how some frustrated investors are resorting to the courts to recoup their cash.

But earlier this year, the Cayman Island court of appeal ruled in Camulos's favor by dismissing the investor's request.
In order to evaluate the effectiveness of this composite XML generation process, the 17 document images used in the targeted registration evaluation were used again to create the composite XML. These 17 images contain portions of six different articles: three images of article 1, three images of article 2, two images of article 3, three images of article 4, four images of article 5, and two images of article 6.

Figures 4.15-18 present the OCR accuracy measures of the composite articles that were created from the combination process. However, to fully understand these accuracy measures, the accuracy of the targeted registration method must be noted. Thus, Table 4.9 presents the accuracy of the targeted registration method for the specific images that comprised each article. This accuracy was calculated using the results of the confusion matrix in the targeted registration method above.

Note that the headlines in most of the articles were found correctly. It is also interesting to observe the number of articles that had high OCR accuracy. However, a significant factor for those that did not perform well is the targeted registration results. For example, the two images that comprised Article 6, in the Manually Separated, Auto-Classified method, did not register at all. Thus, leaving the OCR accuracy for this pair of images at 0. It is believed that by capturing more images of the selected articles at a closer distance, the OCR accuracy for those articles would be improved.

<table>
<thead>
<tr>
<th></th>
<th>Niblack Adaptive Binarization</th>
<th>Otsu Binarization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Full Manual</td>
<td>ManSep AutoClass</td>
</tr>
<tr>
<td>Article 1</td>
<td>71%</td>
<td>100%</td>
</tr>
<tr>
<td>Article 2</td>
<td>78%</td>
<td>90%</td>
</tr>
<tr>
<td>Article 3</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Article 4</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Article 5</td>
<td>92%</td>
<td>71%</td>
</tr>
<tr>
<td>Article 6</td>
<td>60%</td>
<td>0%</td>
</tr>
</tbody>
</table>

Table 4.9 Targeted Registration Accuracy
Figure 4.15 OCR Accuracy of Composite XML for Fully Manual Process

Figure 4.16 OCR Accuracy of Composite XML for Manually Separated Process

Figure 4.17 OCR Accuracy of Composite XML for Manually Classified Process
4.5 Conclusion

The process whereby a composite XML encoding of a set of document images is created was described. The steps in the process include a registration scheme and a method for combining two XML versions of the document image. Results from an evaluation of each these steps were also presented in this chapter.

When a document image has passed through the document image-processing pipeline described in Chapter 3, the information derived from the image may be incomplete or unreadable. In order to improve the readability of an article, more than one document image should be captured and the useful information derived from the set of images can then be combined to form an improved version of the information. In this chapter a scheme to improve the accuracy of the OCR was presented. The final product of this method is an XML document that a user can interact with using a Voice User Interface (VUI). This VUI is described in Chapter 5 and Chapter 6.
Chapter 5 Design of a Voice User Interface

The dramatic proliferation of mobile computing systems has increased interest in voice recognition technologies due to the small physical size and mobile nature of these devices. Use of a keyboard has often proven to be impractical, thus many mobile-devices support the use of basic speech recognition systems.

For blind users of technology, interaction with mobile devices is often limited due to the requirement of interaction with icons, buttons, and other sight-oriented expectations that designers build into such systems. The primary focus of this chapter is the development of an eyes-free interaction model that supports speech recognition and the user interaction with TYFLOS. Recall from Chapter 1, the system qualities most useful to a blind user of TYFLOS are spatial information, find-ability, regression, and a voice user interface (audible input and output). The goal of the image-processing pipeline presented in Chapter 3 and the creation of the composite XML file outlined in Chapter 4 is to capture the information required to support these qualities through a VUI. This chapter presents the development of an eyes-free model of interaction based on cognitive field studies.

In order to develop an initial model of eyes-free interaction, consider the navigation system of a bat or a starfish. Each of these animals navigates its environment without sight. A bat uses sonar signals and a starfish uses tactile input to navigate its environment. An abstract model of this navigation system is presented in Figure 5.1. Consider the sonar signal and tactile signal as input to the model. The creature must process the signal in order to understand the situation in which it finds itself. Then, based on the destination goal it will continue on the current path or adjust its path. The bat or starfish will continue in this loop of processing the input signal, evaluation of its current situation, and adjustment until the destination is reached.

![Figure 5.1 Eyes-Free Navigation Model](image)

The application of this eyes-free navigation model to the reading task leads to further understanding of the interaction with a mobile reading device. By changing the signal from a navigation input signal to a document image, the signal processing component becomes the document image processing processes.
described in Chapter 3 and Chapter 4, and the speech-to-text processing for interpretation of user commands. The output of this device is the articulation of the read document or system commands to direct the user in the capture of an image. The human responds to this articulation by continuing to listen, as long as the user expectation is met, or by issuing a new voice command back to the system.

Recall from Chapter 1 that the primary goal of reading is comprehension, thus this model of eyes-free interaction of a mobile reader must support this ultimate goal. Similar to the continual situation evaluation performed by an eyes-free navigation system, comprehension increases as the user hears more of the document read. Thus, in a human-in-the-loop model of TYFLOS (figure 5.2), the human is modeled with an integrator.

This model is comprised of two primary components: the image/voice processing device and the human user. Two forms of input are provided to the device, namely the document image that is captured with a camera and the user’s voice commands. The device processes the image through the document image pipeline, and performs speech-to-text processing in order to interpret the user’s commands. Text-to-speech processing is used to articulate the text found within the image and the available user directive system commands. The arc connecting the Device to the Human represents the articulation of both the system commands and the reading of the document image. As the human listens, an evaluation of the expectation of what this articulation should be and what the user actually hears will keep the user in a listening state, or cause a new command to be issued. When the user has finished listening, the user may issue a command that will direct the device to operate another function. This command could be a read-oriented command such as “read headlines”, or a system command such as “take picture.”

Note that the ultimate goal of the device is for the human user to comprehend the text of the page as well as the structure or layout of the page. This layout supports the users’ understanding of the spatial orientation of the articles on the page, which in turn supports the find-ability of items of interest. Background noise, as well as verbal sounds from the user, is modeled in this diagram as noise.
While the human-in-the-loop model of TYFLOS provides a strategic understanding of the system, it glosses over the important details of the interaction, and is only able to be evaluated at the conceptual level. Thus, this model is refined with a Rasmussen decision ladder. Briefly, the decision ladder is a generic representation of the steps that may be involved in decision-making [Rasmussen et al, 1994]. This model is presented in Figure 5.3.

![Figure 5.3 Rasmussen Decision Ladder for TYFLOS](image)

In this model, the system and the human are modeled together. The goal of the user is represented at the top as comprehension. The system-oriented tasks are presented on the left, and the user-oriented tasks are on the right. The system will process the document through the image-processing pipeline, read the document to the user, and issue directional commands to the user when needed. The user will listen to the document or issue commands to the system based on the evaluation toward the goal of comprehension.

Further description of the user interaction includes the detailed tasks of both the system and the user. Figure 5.4 presents these tasks and their interaction with the system and the user as an evaluator. In this model the system tasks are on the left, and the user tasks are on the right. The system captures the document image and prepares it for interaction with the user. User expectation is represented as human interest in what the user is hearing. While the user is interested in the content of the document, the system
will remain in a “Continue Reading” state and the user will remain in a “Continue Listening” state. When
the user is no longer interested in the content or the system finishes reading the section, the user will issue a
voice command. These commands include movement-oriented commands (i.e. skip sentence or repeat
paragraph), the stop command, or commands to capture a new document image.

If the user has already heard a section of the document, a skip command will be issued until the user is
pleased with the location within the document. Similarly, when the user did not fully comprehend a section
of the text, a repeat command will be issued. When the user issues one of these commands, the system must
respond by going into the System Moves state.

When the system determines that a document cannot be read accurately due to page misalignment, a
system command must be issued to the user. The user must adjust the location of the document relative to
the location of the cameras. This is represented in the model as the user Move state. Once the user is
satisfied with the reading of the document, the interaction is complete, represented in the model as the
Done state.

In order to evaluate this model, two independent user studies were performed. The first study evaluated
a set of user commands, represented as the VUI Commands arc in Figure 5.2. The second study evaluated a
set of device articulated commands in the context of the device also reading a document to the user. This
articulation is represented as the Articulation arc in Figure 5.2. The VUI Commands study is described in
Section 5.1 and the Articulation study is presented in Section 5.2. Section 5.3 provides the conclusions of
these studies and their impact on the interaction model.
5.1 VUI Commands Evaluation

Recall from Chapter 1 that two of the important qualities of a mobile reading device for the blind are regression and audible input/output. In order to support regression the system must support the ability to repeat a sentence, paragraph, or section. The design of the TYFLOS user command set supports these functions plus other document navigational commands such as the ability to skip a sentence or paragraph. In this section, the user commands are presented using a formal grammar and are evaluated with a formal user test.

5.1.1 VUI Command Design

According to Fred Brooks, a system with a solid conceptual integrity is achieved by the adoption of a familiar mental model and its consistent extension to a computer implementation. [Brooks, 1995] Having a solid conceptual model for a system also makes it easier to build and less subject to defects. The conceptual model must account for the various interests in a system, and ultimately resolve the user’s interest primarily. Thus, based on the task analysis presented in Chapter 1 and the human-in-the-loop model in Figure 5.2, three essential interests of a TYFLOS user were identified: identify articles of interest, read an article of interest, and comprehend the content of the article.

A grammar was developed to formally describe the interaction design of the user and system commands. A grammar consists of formation rules that describe which strings formed from the alphabet of a formal language are syntactically valid within the language. A grammar only addresses the location and manipulation of the strings of the language. It does not describe anything else about a language, such as the semantics.

A grammar generally consists of a set of rules with an assigned start symbol; the language described is the set of strings that can be generated by applying these rules. Thus, a grammar is usually thought of as a language generator; however, it can also be used as a string recognizer, which determines for any given string whether it belongs to the language.

An initial grammar for the user commands is formally presented in the network shown in Figure 5.5 and a formal grammar definition in Figure 5.6. This grammar supports the user interaction presented on the right side of the model in Figure 5.4.
INSTRUCTION -> VERB | VERB-PHRASE | RESPONSE
RESPONSE -> ohkay | alright | go
VERB -> start | stop | pause | help
VERB-PHRASE -> COMMAND NOUN-PHRASE | LEVEL-PHRASE | VOICE-PHRASE
COMMAND -> read | repeat | skip | begin | end | take picture
NOUN-PHRASE -> NOUN | ARTICLE-PHRASE | λ
NOUN -> paragraph | sentence | headings | headlines | heading | headline
ARTICLE-PHRASE -> (article | section) NUMBER
NUMBER -> one | two | three | four | five | six | seven | eight | nine
LEVEL-PHRASE -> (increase | decrease) (volume | speed)
VOICE-PHRASE -> change voice VOICES
VOICES -> reed | shelly | glen | sandy

Examples of valid user commands based on this grammar include:

read headlines       repeat sentence
skip paragraph       increase volume

In the grammar presented in Figure 5.6, the start symbol is INSTRUCTION. An INSTRUCTION can be comprised of a VERB, a VERB-PHRASE, or a RESPONSE. The VERB is a set of simple commands such as start or stop. The RESPONSE is also a simple set of utterances that a user may speak to the system in the course of interaction.

The VERB-PHRASE is the most interesting symbol in the grammar. This symbol is comprised of a COMMAND followed by a NOUN-PHRASE, a LEVEL-PHRASE, or a VOICE-PHRASE. The LEVEL-PHRASE and VOICE-PHRASE symbols represent the ability for a user to adjust the volume of the system, the speed at which the document is read to the user, and the ability to select a voice. These system settings were not evaluated in the test presented here.
It is the **COMMAND NOUN-PHRASE** symbol sequence that provides the robust interaction with TYFLOS. This interaction includes the ability to move throughout a document with skip and repeat commands, have section headings read, have whole sections read, or select a section by number.

To further illustrate the user command sequence within the TYFLOS system, a sequence diagram is presented in Figure 5.7. A sequence diagram, defined in the Unified Modeling Language (UML) [Fowler, 2003], illustrates different processes, and the messages exchanged between them, in the order in which they occur. Sequence diagrams enable the specification of simple runtime scenarios in a graphical manner. Thus, the sequence diagram in Figure 5.7 illustrates the system modules that would be engaged to support a few different user commands.

![Figure 5.7 User Command Sequence Diagram](image)

### 5.1.2 Evaluation of User Commands

In order to evaluate the grammar presented in Figure 5.6, a user study was conducted to determine whether the user commands defined in the grammar improved the user’s performance in the reading task. By recording a simulated voice that read the articles from the front page of two different editions of *The New York Times*, a prototype of the proposed interaction was developed. All of the articles from the first edition were recorded without any form of interaction. The articles from the second edition were recorded
in order to easily simulate the proposed interaction design: each headline, sentence, and paragraph were recorded separately.

The participants were asked to answer four questions, the answers to which were derived from similar locations within each edition of the newspaper. The answers to two of the questions were related to the headlines on the page, and two of the answers were unrelated to the headlines. The answer to the questions that were related to the headline were found in article four (Figure 5.8 R1 and R2). The answers to the questions that were not related to the headline were found in article one of both papers (Figure 5.8 U1 and U2). This structure supported the comparison of the search time between reading using the user commands and a sequential reading. Figure 5.8 illustrates the location of the answers on the page.

![Figure 5.8 Location of Articles on the Printed Page](image)

5.1.3 Method of Evaluation

Simply testing the commands by themselves is not interesting, nor does it provide a control group by which comparison can be made. Thus, a within-subject experiment was conducted to test whether the navigational commands defined in the proposed grammar improved users’ task performance and satisfaction.

5.1.3.1 Participants

Participants were recruited from the blind community at a university in the USA, and all participants were given a $5 gift card to a local coffee shop for their participation. The investigator recruited the participants personally through email. The research was conducted in the Office of Disability Services at the participating university.
Eight people, aged 19 to 52, participated in this study. Five were males and three were females, with an average age of 28. Each participant’s visual impairment was significant enough to require him or her to use alternative devices to access reading material. The visual impairment was determined by asking the participant about his/her reading habits before the experiment began. Seven of the eight participants used a long-cane for mobile navigation. The eighth participant does not use any navigational aid. The participants’ reading habits are described below:

**Note Taking Device:** 6 participants used a laptop computer, 1 participant used a BrailleNote [HumanWare, 2010], and 1 participant does not take notes. (Average use: 5 years)

**Printed Braille:** 6 participants reported using printed Braille when textbooks or other handouts could be translated. 2 participants do not use Braille. (Average use: 15 years)

**Screen Reader:** 7 participants reported using Jaws [Freedom Scientific, 2010]. 1 participant uses ZoomText [AI Squared, 2010]. (Average use: 11 years)

**Scanning/Reading:** 4 participants use OpenBook [Freedom Scientific, 2010]. 2 participants use Kurzweil 1000 [Kurzweil, 2010]. 2 participants did not use a scanner. (Average use: 6 years)

Each participant held a newspaper during the experiment in order to simulate the experience of reading the paper. The interaction was prototyped on a laptop computer.

5.1.3.2 Measures

The primary independent variable was the interaction type: navigable reading or sequential reading. Another independent variable was the type of question asked of the participant: whether the question was related to a headline or not.

There were two dependent variables in the experiment. The first was the participant’s response time, measured by the elapsed time from when the participant began to read the newspaper until the question was answered correctly. The second dependent variable was the participant’s satisfaction with the interaction process. In order to measure the satisfaction, a four-point Likert scale questionnaire was created that consisted of four questions asking for the participant’s opinions about the ease of interaction with each form of navigation, with 1 being the best and 4 being the worst.

5.1.3.3 Procedure

The experiment took 15-30 minutes to complete for each participant.
Before the test began, each participant was verbally asked a set of questions from a brief pretest questionnaire. These questions focused on understanding the tools and techniques that each participant commonly used to read various types of materials. Each participant was also given a brief introduction to the interaction prototype. Once the participant indicated that he or she was comfortable with the interaction, the formal test began.

Each participant was given a newspaper to hold, and asked a question regarding the content of the paper. The participant issued a verbal command to begin reading the content, and answered the question as soon as the answer was discovered.

Each type of question (related to headline vs. unrelated to headline) was tested with the proposed interaction and without any interaction (which is the capability of current mobile readers). A Latin-square was used to partially counter-balance the order in which different interaction mechanisms and questions were presented to the participants.

To simulate a navigable reading experience, the participant issued a verbal command (i.e. read headlines) and the investigator played the recording related to the command. For the simulation of the sequential reading experience, all of the articles were read, beginning with the first until the participant discovered the correct answer. Each test was terminated when the user answered the question correctly.

5.1.4 Evaluation of Results

The experimental results were analyzed using analysis of variance (ANOVA), with repeated measures on both the interaction and the question type, and an α value of 0.05. Table 5.1 summarizes the ANOVA results. As shown in the table, there was a significant main effect of interaction on the participants’ response time and satisfaction.

<table>
<thead>
<tr>
<th>Statistical Effect</th>
<th>Dependent Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Response Time</td>
</tr>
<tr>
<td>Interaction Mode</td>
<td>*F(1,7) = 5.59, p = 0.0499</td>
</tr>
<tr>
<td>Question</td>
<td>*F(1,7) = 3.29, p = 0.1122</td>
</tr>
<tr>
<td>Interaction Mode X Question</td>
<td>*F(1,7) = 20.36, p = 0.0028</td>
</tr>
</tbody>
</table>

Table 5.1 ANOVA Results of the Experiment (* indicates the significant effect on the dependent variable)

Tukey Honestly Significant Difference (HSD) tests reveal that compared to no interaction, the navigable reading significantly reduced response time, increased task accuracy, and increased satisfaction.
An unexpected discovery was that the mean response time to find the answer to the question that was not related to the headline was lower for the sequential reading than for the navigable reading (see Table 2). This may be due to the fact that the question for the navigable reading contained the word ‘Obama’ as did the headline for article three. Thus, three of the eight participants requested article three before requesting article one. While this is not conclusive, it does indicate that for some tasks a sequential reading may be faster than a navigable reading. Further research would need to confirm this.

<table>
<thead>
<tr>
<th>Reading Question Type</th>
<th>Mean Response Time Score</th>
<th>Std. Error Response Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigable / Related</td>
<td>71.38 seconds</td>
<td>13.53</td>
</tr>
<tr>
<td>Sequential / Unrelated</td>
<td>129.75 seconds</td>
<td>13.53</td>
</tr>
<tr>
<td>Navigable / Unrelated</td>
<td>145.25 seconds</td>
<td>13.53</td>
</tr>
<tr>
<td>Sequential / Related</td>
<td>152.00 seconds</td>
<td>13.53</td>
</tr>
</tbody>
</table>

*Table 5.2 Response Times for Each Interaction Method*

Participants were asked to rate the difficulty of answering the posed questions using a four-point Likert scale questionnaire. Each participant’s opinion was ranked with 1 being the best and 4 being the worst. Though the sequential reading may have been faster for some of the participants, all of the participants rated the navigable interaction higher than sequential reading (see Table 5.3).

<table>
<thead>
<tr>
<th>Reading Question Type</th>
<th>Mean Likert Scale Score</th>
<th>Std. Error Response Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigable / Related</td>
<td>1.00</td>
<td>0.15</td>
</tr>
<tr>
<td>Navigable / Unrelated</td>
<td>1.38</td>
<td>0.15</td>
</tr>
<tr>
<td>Sequential / Unrelated</td>
<td>3.00</td>
<td>0.15</td>
</tr>
<tr>
<td>Sequential / Related</td>
<td>3.13</td>
<td>0.15</td>
</tr>
</tbody>
</table>

*Table 5.3 Likert Scale Scores for Each Interaction Method*

During the post-test survey, each participant was asked his or her opinion of the completeness of the proposed grammar. Half of the participants suggested adding the term “back” to simply go back one sentence or paragraph as the proposed “repeat” command does. Similarly, they suggested supplementing the “skip” command with a default to advance to the next sentence rather than require the user to say the word “sentence.”

By taking the participant feedback into consideration, the user grammar was updated with “back” and “skip” as a default. The updated grammar is presented in Figure 5.9.
5.2 Articulation Evaluation

Recall from Chapter 1 that three of the important qualities of a mobile reading device for the blind are the provision of spatial cues, find-ability, and audible input/output. In order to support the spatial cues and find-ability, TYFLOS tracks the location of one document image relative to others of the same document and therefore can direct the user with in-document navigation. This feature is supported with audible system commands. In this section, the grammar for the system commands is presented with a formal evaluation of the commands and results.

5.2.1 System Command Design

As with the user commands, a grammar for the system commands was defined. The grammar network in Figure 5.10 and the formal grammar definition in Figure 5.11 present a formal description of the grammar. This grammar supports the system interaction presented on the left side of the model in Figure 5.4.
Examples of valid system commands based on this grammar include:

- move paper forward
- move paper up
- turn to page five
- move paper down and left

In the grammar presented in Figure 5.11, the start symbol is \texttt{COMMAND}. A \texttt{COMMAND} can be comprised of a \texttt{VERB}, a \texttt{VERB-PHRASE}, or a \texttt{RESPONSE}. The \texttt{VERB} is a set of simple commands: start and stop. The \texttt{RESPONSE} is also a simple set of utterances that the system may speak to the user in the course of interaction.

The \texttt{VERB-PHRASE} is the most interesting symbol in the grammar. This symbol is comprised of an \texttt{ADJUST} symbol followed by a \texttt{DIRECTION} or a \texttt{SECTION-PHRASE} symbol. The \texttt{DIRECTION} symbol represents the direction which the user should move the paper relative to the previously captured image. The \texttt{SECTION-PHRASE} symbol represents the hierarchy of document components of which the system can provide direction to the user.

To further illustrate the system command sequence within the TYFLOS system, a sequence diagram is presented in Figure 5.12. This sequence diagram illustrates the system modules that would be engaged to support a few different system commands. For example, when a user speaks the “read article one” command, TYFLOS will determine the position of the article and check whether enough information is available to read to the user. If there is not enough information, the instruction generator will direct the user in how to move the document in order to capture an image with an increased information content (i.e. “move down and right”).

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure5_11.png}
\caption{Formal System Command Grammar}
\end{figure}
5.2.2 Evaluation of System Commands

In order to evaluate the grammar presented in Figure 5.11, a user study was conducted to determine whether the system commands defined in the system grammar improved the user’s performance in the reading task. This test utilized a commercially available reading device as the control measure, and supplemented the device interaction with feedback from the proposed grammar. The time required to find the article in the upper left of a printed page was measured in order to determine the affect of the feedback on the user’s performance.

A commercially available mobile reading device is the knfbReader Classic [knfbReading Technology, 2008]. Three different mobile devices are currently available from knfbReader, Inc. that provide the functional equivalent of a scanner/reading machine on a cell phone or PDA. Each of these devices captures a photograph of the printed page and process the document image for reading. Though the knfbReader Classic is an older model, it was used in this experiment due to its availability. The feedback of the Classic model related to alignment is limited to rotation information (i.e. “rotated 3 degrees clockwise”). Newer models provide a feature called the “field of view”, which confirms that a page edge is in view, the percentage of the captured picture that is filled by the page, and rotation information. While this additional information is useful, it does not direct the user in how to adjust the camera to accomplish the reading task more effectively.

The knfbReader Classic is engaged when the user takes a picture of the page to be read. The character recognition software in conjunction with text-to-speech processing processes the document image and
reads the contents of the document aloud. As each word is spoken, the text is presented and highlighted on
the device’s screen.

While this device provides useful access to written text to the blind community, it is not as usable as
one may expect. The device must be held approximately 15 inches above a document that is laid on a flat
surface. As mentioned above, the device also provides limited feedback to help the user account for image
distortions caused by the placement of the camera, page curl, or view perspective. Thus, it may take a few
tries for the user to guess the proper position of the camera to have the page read correctly.

The usefulness of auditory feedback to the user when taking the picture of the page to be read was
considered. A blind user of technology must rely on auditory or tactile feedback in order to interact with the
device in any way. Thus, while the knfbReader Classic provides blind users access to information that
previously was not attainable, further enhancements to provide a rich user experience will rely on robust
auditory feedback.

An approach for providing navigational cues to blind users of a GUI was the development of earcons
[Brewster, 1998]. Earcons are abstract, musical tones that can be used in structured combinations to create
sound messages to represent parts of the interface. Earcons have been shown to be an effective means of
communicating information using auditory tones. In one experiment, users were able to correctly recall the
meaning of 81.5% of the earcons. Thus, the earcons were shown to improve the usability of navigating a
menu structure in a classical GUI setting.

Referring back to the concept that conceptual integrity is achieved by the adoption of a familiar mental
model [Brooks, 1995], it is not surprising that a study of directional instruction techniques indicates that
reference body parts is the clearest technique for explaining positioning and orientation to blind users.
Ghiani, et al. [Ghiani, et al., 2008] also reported the use of audible feedback in order to improve the
navigation of a blind user of museum guides. The feedback mechanism they developed supported the
system’s conceptual integrity by providing either words or earcons to guide the user.

For the purposes of the development of the TYFLOS interface, the objective of this study was to
evaluate the proposed grammar, another quantifiable objective emerged, namely the objective of examining
whether a mobile reader for the blind would be easier to position correctly with intelligent auditory
feedback compared to the standard auditory feedback currently available.
5.2.3 Method of Evaluation

Simply testing the commands by themselves is not interesting, nor does it provide a control group by which comparison can be made. Thus, a within-subject experiment was conducted to test whether providing auditory feedback in a mobile reading device for the blind can improve users’ task performance and satisfaction.

5.2.3.1 Participants

Participants were again recruited from the blind community at a university in the USA, and all participants were given a $5 gift card to a local coffee shop for their participation. The investigator recruited the participants personally through email. The research was conducted in the Office of Disability Services at the participating university.

In this study, six people, with ages from 19 to 52, participated in this study. Three were males and three were females, with an average age of 30. Each participant’s visual impairment was significant enough to require him or her to use alternative devices to access reading material. The visual impairment was determined by asking the participant about their reading habits before the experiment began. Five of the six participants used a long-cane for mobile navigation. The sixth participant does not use any navigational aid. The participants’ self-reported reading habits are listed below:

- **Note taking device**: 4 participants used a laptop computer, 1 had a BrailleNote [HumanWare, 2010], and 1 participant does not use any note taking devices.
- **Printed Braille**: 5 participants used printed Braille when textbooks or handouts could be electronically translated, and 1 does not use Braille, but rather a magnifier.
- **Screen reader**: 5 participants use Jaws [Freedom Scientific, 2010], whereas 1 used ZoomText [AI Squared, 2010]
- **Scanning/reading**: 3 participants use OpenBook [Freedom Scientific, 2010], 2 used Kurzweil 1000 [Kurzweil, 2010] and 1 had not used a scanner.

A knfbReader Classic was used in the experiment. The interaction with the device was supplemented with directional sounds produced by a computer in order to provide a richer user experience.

5.2.3.2 Measures

The primary independent variable of the experiment was the type of auditory feedback: no feedback, word feedback, or earcon feedback, similar to those used in the auditory feedback literature. Since
TYFLOS address page skew and curvature directly, another independent variable, the degree of page curvature, was also included in this study. The goal of this measure was to determine the effect of auditory feedback on the participants’ performance of capturing a document image that did not need adjustment. Three types of documents, a newspaper, a magazine, and a thick textbook, were presented to the participants. The newspaper did not introduce any form of curvature or page distortion when laid flat on the table. The magazine with small binding introduced a slight curvature of the page. The thick textbook, however, introduced a pronounced curvature of the page.

There were three dependent variables in the experiment. The first was the participants’ response time, measured by the elapsed time from when the participant picked up the device and positioned it correctly above the documents to be read. The number of times a participant attempted to position the device was also noted, and the test was terminated after the third attempt, regardless of whether the task of identifying the first line of text was accomplished or not. The second dependent variable was the participants’ task accuracy, i.e., whether the participants successfully positioned the device after no more than three attempts. The third dependent variable was the participants’ satisfaction with the process of using the device. This was measured using a four-point Likert scale questionnaire which consisted of six questions asking for the participants’ opinions about the ease of using the device with each form of auditory feedback (including the case without feedback), with 1 being the best and 4 being the worst.

5.2.3.3 Procedure

The experiment took 45-60 minutes to complete for each participant.

Before the test began, each participant was verbally asked a set of questions from a brief pretest questionnaire. These questions focused on understanding the tools and techniques that each participant commonly used to read various types of materials. Each participant was also given a brief introduction to the knfbReader Classic, which included repeated attempts at using the device to read a newspaper. The participants were instructed that, according to the manufacturer, the device should be held 15” above the document and that the best results were obtained when the reader was aligned within 0 to 10 degrees of the page. Once the participant indicated that he or she was comfortable with the device after some practice, the formal test began.

A knfbReader Classic was placed on a mouse pad to the right of the participant. A selected document was placed flat on a table in front of the participant, and he or she was asked to pick up the device, use it to
read the first line of the text on the page, and then place the device back onto the mouse pad. The time from when the device was lifted off of the mouse pad until it was placed back onto the mouse pad was recorded.

Figure 5.13 shows a participant during the test with a newspaper and a textbook.

![Figure 5.13 Participant Using knfbReader to Read a Newspaper and a Textbook](image)

Each type of document (i.e., a newspaper, a magazine, or a textbook) was tested without positioning feedback (the default setting of the device), auditory spoken feedback, and earcon feedback. A Latin-square was used to partially counter-balance the order in which different feedback mechanisms and documents with different degrees of curvature were presented to the participants.

In order to test the usefulness of auditory feedback on using the device, a set of earcons and spoken words were developed (see Table 5.4). Just before each test began, the auditory sounds for that test were reviewed with the participant. During the test, the knfbReader processed an image of the page, which was followed by auditory feedback based on the portion of the page that the device started to read. The investigator determined the location on the page that was being read by the device and then provided feedback by playing the appropriate sound on his computer. If adjustment was required in more than one direction, feedback for each direction was provided (i.e. “backward left” was a common auditory feedback). The participant would then adjust the position of the device to a new location and make another attempt to read the first line of text on the page again. When this task was accomplished, the investigator indicated success to the participant, and the device was placed back to the starting position on the mouse pad.
<table>
<thead>
<tr>
<th>Direction</th>
<th>Earcon</th>
<th>Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>Revving car engine</td>
<td>“Forward”</td>
</tr>
<tr>
<td>Backward</td>
<td>Construction truck backing up</td>
<td>“Backward”</td>
</tr>
<tr>
<td>Up</td>
<td>Whistle with tone going up</td>
<td>“Up”</td>
</tr>
<tr>
<td>Down</td>
<td>Whistle with tone going down</td>
<td>“Down”</td>
</tr>
<tr>
<td>Right</td>
<td>Rattle snake</td>
<td>“Right”</td>
</tr>
<tr>
<td>Left</td>
<td>Knock at door</td>
<td>“Left”</td>
</tr>
</tbody>
</table>

Table 5.4 Types of Auditory Feedback

It is interesting to note that all six participants assumed that the first line of text was at the top of the page and felt for the top of the page in order to take the first picture. However, the text on many sample pages started further down the page and thus the camera needed to be moved closer to the participant.

5.2.4 Evaluation Results

The experiment results were analyzed using analysis of variance (ANOVA), with repeated measures on both the auditory feedback and the degree of page curvature, and an α value of 0.05. Table 5.5 summarizes the ANOVA results. As shown in the table, there was a significant main effect of auditory feedback on the participants’ response time, task accuracy, and satisfaction, and a significant interaction effect between the auditory feedback mechanism and the degree of page curvature in the participants’ task accuracy.

<table>
<thead>
<tr>
<th>Statistical Effect</th>
<th>Task Accuracy</th>
<th>Response Time</th>
<th>Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auditory Feedback</td>
<td>*F(2,10) = 13.57, p = 0.001</td>
<td>*F(2,10) = 61.48, p &lt; 0.0001</td>
<td>F(2,10) = 26.85, p &lt; 0.0001</td>
</tr>
<tr>
<td>Page Curvature</td>
<td>F(2,10) = 2.00, p = 0.18</td>
<td>F(2,10) = 0.54, p = 0.6</td>
<td>NA</td>
</tr>
<tr>
<td>Feedback X Curvature</td>
<td>*F(4,20) = 4.17, p = 0.013</td>
<td>F(4,20) = 1.72, p = 0.18</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 5.5 ANOVA Results (* means the significant effect on the dependent variable)

Tukey Honestly Significant Difference (HSD) tests reveal that compared to no auditory feedback, the earcon and the word prompts significantly reduced response time and increased task accuracy and satisfaction (Table 5.6). However, there was no significant difference between the use of earcons and words in these three measures.

117
Table 5.6  Tukey HSD of Prompts

<table>
<thead>
<tr>
<th></th>
<th>Success</th>
<th>Response Time</th>
<th>Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>None X Earcon</td>
<td>5.9017 &gt; 3.61</td>
<td>13.3425 &gt; 3.61</td>
<td>8.1667 &gt; 3.61</td>
</tr>
<tr>
<td>None X Word</td>
<td>6.7535 &gt; 3.61</td>
<td>13.8058 &gt; 3.61</td>
<td>9.6078 &gt; 3.61</td>
</tr>
<tr>
<td>Earcon X Word</td>
<td>0.3477 &lt; 3.61</td>
<td>0.4633 &lt; 3.61</td>
<td>1.4411 &lt; 3.61</td>
</tr>
</tbody>
</table>

Figure 5.14 illustrates the interaction effect between auditory feedback and the degree of page curvature on the task response time. From the figure, we can see that adopting auditory feedback led to the largest improvement in response time when the page had a slight curve. This may be due to the “ceiling” and “floor” effects. When the page is flat, it is relatively easy to correctly position the reader even without any auditory feedback. When the page has a deep curve, on the other hand, it is challenging to capture the document image even with auditory feedback. However, it is surprising to see that participants actually performed better on the deeply curved page than on the slightly curved page when they did not receive any auditory feedback. This could be due to a number of factors including the performance of the device on the highly reflective material of a magazine page, and that the layout of a magazine page does not typically have text at the top, in contrast to a textbook page, which generally does have text at the top.

Figure 5.15 demonstrates an improvement in the successful completion of the task with audible prompts over no prompts.
During the post-test survey participants indicated that the audible prompts were useful in completing the task and provided an improved interaction. One interesting note is that three of the six participants wanted to know how far to move the device when prompted with a movement direction. The ability to provide this feature is beyond the scope of this research. The system command grammar withstood the test of the users and therefore did not need to be updated. Since the earcons and the audible words were not statistically different, the TYFLFOS prototype includes the audible words rather than the earcons.

5.3 Conclusion

In this chapter an eyes-free interaction model was presented. From this model, a grammar was defined for the user commands and the system commands that comprise the VUI of the TYFLOS system. These grammars were evaluated in two separate studies with visually impaired participants. The results of these studies influenced changes in the user commands grammar. In the next chapter the two evaluated grammars will serve as the basis for the development of a Stochastic Petri Net model of the interaction.
Chapter 6 Modeling and Verification of a VUI

While the interaction models in the previous chapter led to the development of two grammars, the models themselves remain strategic in nature (Figure 5.3 and Figure 5.4). The grammars developed in Chapter 5 can be modeled using a Stochastic Petri-Net (SPN), which provides a tactical model that can be used to guide the implementation of a voice user interface (VUI). In this chapter, the development of a model of interaction is presented based on a combination of the two grammars. After a brief introduction to Stochastic Petri-Nets (SPN) (Section 6.1), a SPN model for the user commands is presented with verification results in Section 6.2. This is followed in Section 6.3 with a revised SPN that is evaluated. In the Section 6.4, the unified model of interaction is presented with results from a final user study to verify the model, which is presented in Section 6.5.

6.1 Stochastic Petri-Nets

Statistical speech recognition enables the recognition of a word, phrase, or sentence pronounced when matched with a finite set of possibilities. This technique is typically used in command-and-control settings, such as those found in issuing commands to TYFLOS. The development of a stochastic model of the user commands will enable improved performance of the actual user input due to the fact that modern speech recognition systems are built with a stochastic model as part of the framework.

There are a variety of methodologies used for system modeling, such as formal languages, directed graphs, classical mathematical models, queuing models, and Stochastic Petri-Nets. The stochastic modeling framework utilized for command-and-control systems is essentially a technique for describing complex probabilistic systems in a mathematically tractable way. The major reasons for using the Stochastic Petri-Net model rather than a Hidden Markov Model (HMM) are:

- SPN is an efficient modeling tool for the functional description and analysis of complex systems
- SPN is able to simultaneously describe concurrency, parallelism, and synchronization of events that take place in a complex system, especially when other methodologies lack adequate results
- SPN can be used as a modeling tool for hierarchical and abstracted (top-down or bottom-up) processes
- SPN provides timing during the execution of various events
- SPN presents compatibility with neural networks
• SPN is an efficient interface for control and communication

Formally, a generalized Petri-net model is defined as

\[ \text{SPNG} = \{P, T, A, I, O, M, X, C, L, D, S \}, \]

where

- \( P \): a finite set of places \( \{P_i, i \in Z\} \) that represent a particular state of a physical component. In the model presented here, other letters are used for clarity: \( P \) for system states (or places) or pseudo-states, and \( U \) for user states (or places)
- \( T \): a finite set of transitions, \( \{t_j, j \in Z\} \) that represent a process performed between two states
- \( A \): a finite set of arcs \( \{a_{rij}, r,i,j \in Z\} \) that represent relationships among places \( \{P_i, P_j\} \)
- \( I \): \( I_i \subset (P \times T) \), represents the input function
- \( O \): \( O_j \subset (T \times P) \), represents the output function
- \( M \): a vector of marking (tokens \( T \)) \( \{m_{ij}, i,j \in Z\} \) that represent the status of the places
- \( X \): a vector of time values \( \{x_i, i \in Z\} \) related with the time required by a process to be performed
- \( C \): the alphabet \( \{c_i, i \in Z\} \) of communication
- \( L \): a finite set of possibly marking-dependent firing rates \( \{l_i, i \in Z\} \) associated with the transitions
- \( D \): a finite set \( \{d_i, i \in Z\} \) of delays associated with the transitions
- \( S \): a finite set of structural properties \( \{s_i, i \in Z\} \) associated with places

In this section, a modified version of a Stochastic Petri-Net (SPN) will be used to model the user command grammar of the functionality of TYFLOS.

### 6.2 User Commands SPN

The user command grammar developed in Chapter 5 can be modeled as a SPN. For the purposes of the model, the RESPONSE, LEVEL-PHRASE, VOICE-PHRASE, and NUMBER symbols have been removed. Also, the pause, help, back, begin, end, and take picture tokens have been removed. While this greatly simplifies the grammar, the primary components and directives of the grammar are left in tact, which simplifies the model and its evaluation.

Figure 6.1 and Table 6.1 present the initial set of user commands and their related probabilities modeled as an SPN.
Figure 6.1 Original Stochastic Petri-Net Model

<table>
<thead>
<tr>
<th>No.</th>
<th>State</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>U0</td>
<td>Read Headings Command</td>
<td>0.42</td>
</tr>
<tr>
<td>U1</td>
<td>Repeat Paragraph Command</td>
<td>0.06</td>
</tr>
<tr>
<td>U2</td>
<td>Skip Paragraph Command</td>
<td>0.06</td>
</tr>
<tr>
<td>U4</td>
<td>Skip Article Command</td>
<td>0.02</td>
</tr>
<tr>
<td>U5</td>
<td>Repeat Article Command</td>
<td>0.02</td>
</tr>
<tr>
<td>P0</td>
<td>System Ready</td>
<td>0.56</td>
</tr>
<tr>
<td>P1</td>
<td>System Perform Rewind</td>
<td>0.08</td>
</tr>
<tr>
<td>P2</td>
<td>System Perform Fast Forward</td>
<td>0.08</td>
</tr>
<tr>
<td>P3</td>
<td>System Finished Rewind</td>
<td>0.08</td>
</tr>
<tr>
<td>P4</td>
<td>System Finished Fast Forward</td>
<td>0.08</td>
</tr>
<tr>
<td>P5/U3</td>
<td>System Reading/User Listening</td>
<td>0.42</td>
</tr>
<tr>
<td>P6</td>
<td>System Finished Rewind</td>
<td>0.03</td>
</tr>
<tr>
<td>P7</td>
<td>System Perform Fast Forward</td>
<td>0.03</td>
</tr>
<tr>
<td>P8</td>
<td>System Perform Rewind</td>
<td>0.03</td>
</tr>
<tr>
<td>P9</td>
<td>System Finished Fast Forward</td>
<td>0.03</td>
</tr>
</tbody>
</table>

Table 6.1 Probability Values for Each Place in Figure 6.1
In this model the primary action place is P5/U3, the System Reading/User Listening. Before TYFLOS begins reading it is in the Ready place (P0). When the user issues the Read Headings command (U0) the system moves into the reading place and the user is in a listening place (P5/U3). The system and the user remain in that place until the section has been read, or it is interrupted by a user command. When TYFLOS is in the reading state it can be interrupted by any one of three commands: Repeat Paragraph (U1), Skip Paragraph (U2), or Skip Article (U3).

The probabilities associated with each place in the SPN are presented in Table 6.1. These probabilities were derived from the understanding that sighted readers backtrack, called regression, in their reading approximately 15% of the time [Carpenter and Dahneman, 1981]. In order to further understand the probabilities associated with each place, a study was conducted to evaluate these probabilities, and enhance the model based on user experiences.

6.2.1 Evaluation of Model Design

The primary purpose of this study was to evaluate the initial Stochastic Petri-Net model. However, this evaluation was performed in the context of another important objective, namely that of determining the effect that a default mode of interaction has on user performance. It is important to configure the initial interaction of a user with a device correctly as this can affect the adoption rate of the device. [Kintsch and DePaula, 2002]

For the purposes of this study, a set of the user commands was prototyped and tested in a Wizard-of-Oz scenario. The hypothesis was that a user who is presented with the section headings of a document by default will be more apt to use the other available commands, and thus complete a task more quickly than a user who simply had the document read to them by default.

In this test two documents were audibly presented to the participants in two different ways. In the first method the entire document was presented audibly to the participants from top to bottom similar to other common reading devices. In the second method, all of the headlines of the document were presented to the participants followed by the content of the document from top to bottom without any pause. However, in this second method the participant could interrupt the reading at any time, just as is modeled in Figure 6.1.
6.2.2 Method of Evaluation

Participants were asked to listen to a body of text and answer four questions regarding the content of what they heard. A computer was used to present the text audibly to the participant. A moderator interpreted the commands and controlled the correct computer responses.

6.2.2.1 Participants

Eight sighted people, ages 22 to 48, participated in this study, four males and four females, with a mean age of 34. During a pre-test interview, seven of the participants indicated that they were familiar with computer-generated voice and that they had listened to a book-on-tape, indicating that the concept of audibly interacting with textual content was not foreign to most of the participants.

6.2.2.2 Measures

The primary measure for this study was the elapsed time from when the participant first issued a read command until four questions regarding the content of the material were answered correctly. A timestamp was recorded for each command that a participant issued. Other measures include a pre-test and post-test questionnaire. The pre-test questionnaire was developed to gather information regarding the participants’ familiarity with audible text and computer generated voice. The post-test questionnaire was developed in order to understand the satisfaction level of each participant with two different default answers. Thus, the three dependent variables were response time, successful completion of task, and satisfaction.

6.2.2.3 Procedure

The experiment took 30-45 minutes to complete. Participants were recruited from the moderator’s place of employment. The moderator recruited the participants personally through email. The research was conducted at the moderator’s place of employment.

Before the test began, each participant was verbally asked a set of questions from a brief pretest questionnaire. These questions focused on understanding each participant’s familiarity with audible text and with computer generated voice. Each participant was also given a brief introduction to the available voice commands. Once the participant indicated that he or she was comfortable with the commands, the test was begun.

The test was comprised of a few steps. A participant was either presented with 1.) the list of section headings of the text, or 2.) the beginning of the text, and instructed to issue voice commands to navigate
through the text. At the end of each text were four questions that were to be answered by the participant. When the four questions were answered correctly, the test was completed. The elapsed time from when the participant first heard the computer begin to speak until all four questions had been answered correctly was recorded. The time at which a participant issued a verbal navigational command was also noted.

The documents presented to the participants contained text regarding the Bubonic Plague and William Wallace. Both documents were historical in nature and the questions regarding the content of the documents were found directly in the text.

Each document was tested reading the section headings as the default and the text as the default. Each default was counter-balanced by changing the order in which a specific default was tested with different participants. The document used with each default was also altered with each participant. Table 6.2 demonstrates the order in which the tests were presented to the participants of both tests.

<table>
<thead>
<tr>
<th>Participant</th>
<th>Presentation 1</th>
<th>Presentation 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WW - Text</td>
<td>BB - Section Headings</td>
</tr>
<tr>
<td>2</td>
<td>WW - Section Headings</td>
<td>BB - Text</td>
</tr>
<tr>
<td>3</td>
<td>WW - Section Headings</td>
<td>BB - Text</td>
</tr>
<tr>
<td>4</td>
<td>WW - Text</td>
<td>BB - Section Headings</td>
</tr>
<tr>
<td>5</td>
<td>BB - Text</td>
<td>WW - Section Headings</td>
</tr>
<tr>
<td>6</td>
<td>BB - Section Headings</td>
<td>WW - Text</td>
</tr>
<tr>
<td>7</td>
<td>BB - Section Headings</td>
<td>WW - Text</td>
</tr>
<tr>
<td>8</td>
<td>BB - Text</td>
<td>WW - Section Headings</td>
</tr>
</tbody>
</table>

*Table 6.2 Test Presentation (BB - Bubonic Plague Article, WW - William Wallace Article)*

### 6.2.3 Evaluation Results

This study addressed two main questions. The first was to determine if providing section-headings as a default interaction would lead to an optimal path for navigating a document, versus presenting the text as a whole. Secondly, the study provided insight into the frequency of the use of the verbal commands provided by the user in order to adjust a Stochastic Petri-Net model of the interaction.

The results of the tests were analyzed using Analysis of Variance (ANOVA), with a $p$-value of 0.05. The results of the test indicate that there is no significant contribution of the default interaction between simply reading the text from the top and presenting the headlines first followed immediately with a
continued reading of the text (\( f \)-ratio of 0.0027 and a \( p \)-value of 0.9596). However, the results also show that as the participant becomes more familiar with the interaction, the time to complete the task improves. The average amount of time required to complete the task for the first document (regardless of interaction default or document) was 1008 seconds compared with 367 seconds for the second document.

Participants were asked a series of questions at the conclusion of each test to determine a satisfaction score related to the use of the commands and their ability to complete the given task. Participants were asked to rate the experience with each type of default as “Easy”, “Fairly Easy”, “Fairly Difficult”, or “Difficult”. The results of this survey indicated that the default presentation did not affect the satisfaction. However, there was a correlation between the participants experience and the satisfaction; every participant indicated that the second task was easier than the first. Thus, the time measurements and the satisfaction scores support the notion that experience with the device improves the users ability to interact with it.

The average number of times that a participant issued a particular verbal command was used to adjust the Stochastic Petri Net model of the commands. The percentage of each command issued influenced the updates to the probability values presented in Table 6.1 (see Table 6.3). Figure 6.2 presents the adjusted Stochastic Petri-Net, which accounts for the changes discovered from the test. The new model not only has updated probability values for the places, but also has added two nodes. The two nodes represent navigation at the sentence level, adding to the article and paragraph level navigation in the original model.
The interaction between the participant and the moderator during the “Wizard-of-Oz” scenario may have had an impact on the frequency of commands issued by the participant. The moderator observed several cases where the participant appeared confused on the clarity of the audio or a desire to issue a command but instead continued listening. When asked after the experiment was complete, participants indicated that they did not consciously realize their hidden motivations, but speculated that they were
unsure if it was the correct time to issue a command. Issuing commands to a human may have produced a lack of confidence or an embarrassment factor of doing something ‘wrong’ for the participant, whereas if they were interacting with a machine they may have been more apt to act on their intuitions immediately.

In this study, all of the headings of the document were read followed immediately with a reading of the document, or only the first heading was read followed with a reading of the document. Since different results may be obtained if the headings are read and then the user is required to issue another command to continue the interaction, another test was performed to determine the proper default.

6.3 Revised User Command

Due to the discovery that the participants’ experience was a key factor in the improvement of scores in the previous study, a second study was developed and executed in order to determine the usefulness of the voice commands. In the second test, the same two documents that were used in the first study were presented to the participants, and the participants were required to issue each command separately and the responses they heard were specific to the command. For example, in this test the participant would say “Read Headlines” and only the headlines would be read aloud, not the entire document as in the first study.

6.3.1 Method of Evaluation

Participants were asked to listen to the same body of text and answer the same four questions regarding the content as those in the previous experiment. Just as in the earlier experiment, a computer was used to present the text audibly to the participant, and a moderator interpreted the commands and controlled the correct computer responses.

6.3.1.1 Participants

Eight sighted people, ages 27 to 66, participated in this study, four males and four females, with a mean age of 51. During a pre-test interview, six of the participants indicated that they were familiar with computer-generated voice and that they had listened to a book-on-tape, indicating that the concept of audibly interacting with textual content was not foreign to most of the participants.

6.3.1.2 Measures

The primary measure for this study was the elapsed time from when the participant first issued a read command until four questions regarding the content of the material were answered correctly. A timestamp was recorded for each command that a participant issued. Other measures include a pre-test and post-test
questionnaire. These questionnaires were the same as those used in the previous experiment. Thus, the three dependent variables in this study were response time, successful completion of task, and satisfaction.

6.3.1.3 Procedure

The experiment took 15-30 minutes to complete. Participants were recruited personally through email from the moderator’s list of contacts. The study was conducted at the moderator’s university.

Before the test began, each participant was verbally asked a set of questions from a brief pretest questionnaire. These questions focused on understanding each participant’s familiarity with audible text and with computer generated voice. Each participant was also given a brief introduction to the available voice commands. Once the participant indicated that he or she was comfortable with the commands, the test was begun.

The test was comprised of a few steps. Each participant was instructed to issue voice commands to navigate through the text. The system responses provided to the participant were specific to the command, and therefore at no time was the entire text read from one command as in the previous experiment. At the end of each text were four questions that were to be answered by the participant. The test was completed when the four questions were answered correctly. The elapsed time from when the participant first issued a voice command until all four questions had been answered correctly was recorded. The time at which a participant issued a verbal navigational command was also noted.

The documents presented to the participants were the same as those used in the first experiment. Each document presentation was counter-balanced by changing the order in which a specific document was tested with different participants.

6.3.2 Evaluation Results

This study fulfilled a dual purpose. The first was to gather statistical measures used to further update and refine the SPN model composed at the end of Section 6.2 (Figure 6.2). Secondly, the study provided timing measurements to use in comparison with the first study.

The results of the tests indicate that the user experience does have an impact on the task performance ($t$-test - 0.0006). The average time to complete the first task was 686 seconds as compared with 463 seconds for the second task, regardless of the document.

Participants were asked a series of questions at the conclusion of the test to determine a satisfaction score related to the use of the commands and their ability to complete the given task. Participants were
asked to rate the experience as “Easy”, “Fairly Easy”, “Fairly Difficult”, or “Difficult”. The results of this survey reinforce the correlation between the participants experience and the satisfaction; every participant indicated that the second task was easier than the first, regardless of the document used.

Figure 6.3 presents a modified model of the Stochastic Petri-Net. This model presents the change in user interaction from the system returning to a Read place as it does in the models above, to the system only reading until it has completed the selected section and then returning to a Ready place. In this model, the specific movements within a document, such as skip or repeat, have been combined into an aggregate Move place. An analysis of this SPN confirms that the user commands SPN is bounded, safe, and has no deadlocks.

Figure 6.3 User Commands Stochastic Petri-Net Model

In order to fully describe the SPN presented in Figure 6.3, consider that the read-oriented tokens in the user grammar such as read followed by a qualifier, such as sentence, paragraph, section, and article (and potentially a NUMBER symbol), are represented in the SPN as one place labeled U0. Similarly, the move oriented tokens of the grammar such as skip, repeat, begin and end, which are also followed by a qualifier, are represented in the SPN as one place labeled U2. When the user is listening to the system read text, the user’s state is modeled as place U1, and when the user is waiting for the system to perform a task, such as to respond to a command, the user’s state is modeled as place U3. From the system perspective, the state in which it is ready and waiting for a user command is P0 (note also that the Response symbol from the system grammar is modeled with this place). When the system is reading (regardless of headlines or body text) it is in place P1. The state in which the system is moving within a document (forward or backward) is modeled as P2.
The average number of times that a participant issued a particular category of verbal commands (read or move) was used to set the probability values of the updated user commands SPN (Table 6.4).

<table>
<thead>
<tr>
<th>No.</th>
<th>State</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>U0</td>
<td>Read Command</td>
<td>0.21</td>
</tr>
<tr>
<td>U1</td>
<td>User Listening</td>
<td>0.21</td>
</tr>
<tr>
<td>U2</td>
<td>Move Command</td>
<td>0.05</td>
</tr>
<tr>
<td>U3</td>
<td>User Waiting</td>
<td>0.05</td>
</tr>
<tr>
<td>P0</td>
<td>System Ready</td>
<td>0.21</td>
</tr>
<tr>
<td>P1</td>
<td>System Reading</td>
<td>0.21</td>
</tr>
<tr>
<td>P2</td>
<td>System Moving</td>
<td>0.05</td>
</tr>
</tbody>
</table>

*Table 6.4 Probability Values for Each Place in Figure 6.3*

In this study, only the requested section was read until the participant interrupted with a new command, or the text was entirely read. For example, the system response to the user command “Read Headline 1” was only the text of the first headline, nothing more. The participant had to request each section individually. It is likely that this requirement of the use of the commands enabled the participant to become familiar with the VUI more quickly, leading to a decrease in time to answer the questions.

Table 6.5 presents the average time required for the participants to answer the questions in each experiment (Section 6.2 and Section 6.3). Note that there is a statistically significant difference between the first trial of each experiment (322 seconds), and that there is a statistically significant difference between the first and second trial of both experiments, 642 seconds in the first experiment and 223 seconds in the second experiment. However, there is not a significant difference between the second trial in the two experiments (103 seconds). These results indicate that experience has an impact on task performance and that the interaction form only impacts new users. (See Appendix D for T-Test results.)

<table>
<thead>
<tr>
<th></th>
<th>Trial 1</th>
<th>Trial 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment 1</td>
<td>1008 sec</td>
<td>366 sec</td>
</tr>
<tr>
<td>Experiment 2</td>
<td>686 sec</td>
<td>463 sec</td>
</tr>
</tbody>
</table>

*Table 6.5 Average Time to Complete Task in Each Experiment*

Because the mode of operation impacts new users, but does not impair experienced users, the TYFLOS interface was implemented by reading only the requested section as was tested in the second study.
6.4 Unified Grammar SPN

Recall from the original model (Section 6.2) the RESPONSE, LEVEL-PHRASE, VOICE-PHRASE, and NUMBER symbols, and the pause, help, back, begin, end, and take picture tokens were removed from the user command grammar in order to simplify the model. By continuing with the model simplification performed in Section 6.3, which reduced the read oriented tokens into one place (Read), simplified the move oriented tokens into one place (Move), and reduce the system grammar VERB-PHRASE symbol into one place (Command), the system states and user states can be modeled in a unified SPN.

In order to ensure that each symbol in the grammars presented in Chapter 5 are mapped to a place in the unified SPN, Table 6.6 presents a list of all of the grammar symbols and their related place in the unified SPN (Figure 6.4). Analysis confirmed that the unified SPN is bounded, safe, and has no deadlocks.

<table>
<thead>
<tr>
<th>Grammar: SYMBOL</th>
<th>Place</th>
</tr>
</thead>
<tbody>
<tr>
<td>User: RESPONSE</td>
<td>U6, U8</td>
</tr>
<tr>
<td>User: VERB</td>
<td>U4, U5</td>
</tr>
<tr>
<td>User: COMMAND NOUN-PHRASE</td>
<td>U0, U1, U7</td>
</tr>
<tr>
<td>User: LEVEL-PHRASE</td>
<td>U9</td>
</tr>
<tr>
<td>User: VOICE-PHRASE</td>
<td>U10</td>
</tr>
<tr>
<td>System: VERB</td>
<td>P5</td>
</tr>
<tr>
<td>System: VERB-PHRASE</td>
<td>P6</td>
</tr>
<tr>
<td>System: RESPONSE</td>
<td>P0</td>
</tr>
</tbody>
</table>

Table 6.6 Mapping Grammar Symbols to Unified SPN Places
Figure 6.4 Unified Grammar Stochastic Petri-Net Model

<table>
<thead>
<tr>
<th>System State</th>
<th>User State</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0 Ready</td>
<td>U0 Read Command</td>
</tr>
<tr>
<td>P1 Reading</td>
<td>U1 Move Command</td>
</tr>
<tr>
<td>P2 Process Image</td>
<td>U2 Listening</td>
</tr>
<tr>
<td>P3 Paused</td>
<td>U3 Waiting</td>
</tr>
<tr>
<td>P4 Moving</td>
<td>U4 Stop Command</td>
</tr>
<tr>
<td>P5 Low Information</td>
<td>U5 Help Command</td>
</tr>
<tr>
<td>P6 Command</td>
<td>U6 Pausing</td>
</tr>
<tr>
<td>P7 Help</td>
<td>U7 Take Picture Command</td>
</tr>
<tr>
<td>P8 Waiting</td>
<td>U8 Moving</td>
</tr>
<tr>
<td>P9 Change Level</td>
<td>U9 Change Level Command</td>
</tr>
<tr>
<td>P10 Change Voice</td>
<td>U10 Change Voice Command</td>
</tr>
</tbody>
</table>

Table 6.7 System and User States in SPN
The remaining places in the unified SPN represent states in which the human and the system can be found during the use of the TYFLOS system. For example, when a user issues a Read command (U0) and the system is in the Ready state (P0), transition T6 can be fired, and the tokens from these states are moved to the System Reading state (P1) and the Human Listening state (U2). When the reading is complete, transition T8 is fired, which moves the tokens back to their original positions (U0 and P0). A similar sequence of events is modeled throughout the SPN. Table 6.7 presents a mapping of the places to their state.

A Stochastic Petri Net can be analyzed with a coverability graph, which is derived from a coverability set. A coverability set is a set $S$ of markings such that: 1) it covers all the markings of the reachability set of a graph and 2) for each marking in $S$ that is not in the reachability set, there is an infinite strictly increasing sequence of reachable markings converging to $m'$. A coverability set $S$ is said to be minimal if no proper subset of $S$ is a coverability set. [Finkel, 1990]

Thus, a coverability graph of a Stochastic Petri Net $P$ is a graph such that its set of labels are the coverability set of the SPN and there is an arc labeled by the transition between two nodes if the transition is fireable from $m$ and reaches the marking $m'$. The minimal coverability graph of a Petri net is the unique coverability graph such that its set of nodes is the minimal coverability set. Figure 6.5 presents the coverability graph of the Unified Grammar SPN in Figure 6.4.

![Figure 6.5 Coverability Graph of Figure 6.4](image)
Another method used to analyze a SPN is to simulate token traversal throughout the SPN. In order to accomplish this, the Unified Grammar SPN was modeled using Pipe 3.0 [Bonet, et al., 2007]. In this software, the number of times the SPN transitions fire can be set in order to simulate the token movement throughout the SPN. Thus, the Unified Grammar SPN was modeled and run through the simulator with 100 firings of the transitions within the SPN. The simulation was run 10 times, and the average number of tokens per place was tracked in order to understand the flow of the tokens throughout the SPN. Figure 6.6 presents a graph of the average number of tokens per place for the 10 simulations.

![Figure 6.6 Average Number of Tokens per Place in 10 Simulations](image)

In these simulations, the system Ready state (P0) contained a token half of the time, whereas the user command places (U0, U1, U5, U7, U9, U10) contained a token much of the time. This is a result of the way the commands were modeled, with each containing a token to simulate the potential for a command to be executed. Note that place U4 represents the user stop token; this is modeled as a dynamic place due to user preferences and interactions. Other interesting places are the Low Information place (P5) and the User Waiting place (U3) each of which are more dynamic than the other places. As will be shown in the next section, this models the actual user quite well in that some users are more apt to guess correctly where the document should be placed relative to the cameras, and others are not.

### 6.5 Evaluation of Unified Grammar SPN

A study was conducted for the purpose of evaluating the complete grammar and the Unified Grammar SPN model. This study was performed in the context of an overall evaluation of the TYFLOS system as an alternative reading device. Participants were given the task of answering three questions from two different newspapers with the TYFLOS prototype using voice commands and responding to the commands issued by the prototype. The time required to answer the questions was recorded in order to determine the degree to which experience with the prototype improved the users performance.
6.5.1 Participants

Six visually impaired participants were recruited from a university in the USA. Participants were given a $10 gift card to a local grocery store for their participation. The investigator recruited the participants personally through email. The research was conducted in the Office of Disability Services at the participating university.

Of the participants, aged 19 to 53 (average 29), three were male and three were female. Each participant’s visual impairment was significant enough that he or she used alternative devices to access reading material. The visual impairment was determined by asking the participant about their reading habits before the experiment began. Five of the six participants used a long-cane for mobile navigation. The sixth participant did not use any navigational aid.

The participants’ self-reported reading habits are listed below:

**Note taking device:** 4 participants used a laptop computer, 1 had a BrailleNote [HumanWare, 2010], and 1 participant does not use any note taking take devices.

**Printed Braille:** 5 participants used printed Braille when textbooks or handouts could be electronically translated, and 1 does not use Braille, but rather a magnifier.

**Screen reader:** 5 participants use Jaws [Freedom Scientific, 2010], whereas 1 used ZoomText [AI Squared, 2010]

**Scanning/reading:** 3 participants use OpenBook [Freedom Scientific, 2010], 2 used Kurzweil 1000 [Kurzweil, 2010] and 1 had not used a scanner.

6.5.2 Measures

The two dependent variables in this study were response time and satisfaction. The participant’s response time was measured by the elapsed time from when the participant began to read the newspaper until the three questions were answered correctly. The second dependent variable was the participant’s satisfaction with the interaction type. This was measured using a four-point Likert scale questionnaire, which consisted of four questions asking for the participants’ opinion regarding the ease of interacting with each system, with 1 being the best and 4 being the worst.

6.5.3 Procedure

The experiment took approximately 30 minutes to complete for each participant. Before the test began, each participant was verbally asked a set of questions from a brief pretest questionnaire. These questions
focused on understanding the tools and techniques that each participant commonly used to read various types of materials. Each participant was also given a brief introduction to the TYFLOS prototype, which included repeated attempts at using the device to read a newspaper. Once the participant indicated that he or she was comfortable with the device after some practice, the formal test began.

In order to test the TYFLOS prototype, the glasses, which contain the webcams mounted in them, were placed over the participant’s eyes (see Figure 6.7). A newspaper was placed on the table in front of the participant, and he or she picked up the newspaper and held it in front of the glasses. The participant would then interact with the document using the voice commands of the TYFLOS grammar. The time required for the participant to answer two questions regarding the content of the document was recorded. This measure was repeated using two different newspapers for each participant, providing two time measurements per participant. A Latin-square was used to partially counter-balance the order in which the newspapers were presented to the participants. Figure 6.7 shows a participant using the TYFLOS prototype.

Figure 6.7 Participant Using TYFLOS to Read a Newspaper

During the test, TYFLOS would provide auditory feedback to the participant in order to help them position the newspaper such that an image could be processed. It would also respond with the requested audible text. For example, when the participant issued the voice command “take picture”, audible feedback was provided to note that the picture was taken, that the system was processing the document image, and that the system was ready. Then when the participant issued the voice command “read headlines”, TYFLOS would respond with the headlines found in the picture. When the user requested an article to be read, by saying, “read article one”, TYFLOS would either read the article or direct the participant to move the
document closer to the glasses. When the user was ready, the “take picture” command would be spoken and the process would start over again.

When the participant heard the answer to the task question, he or she would say “stop” and TYFLOS would stop. The question would be answered and either the answer to the next question would be sought, or, when the third question was answered, the test was terminated.

6.5.4 Evaluation Results

Once again the primary purpose of this study was to provide a final set of probabilities to support the unified SPN model of interaction presented in Figure 6.4. However, other useful information can be derived from the study, including participant response time for each task and the satisfaction measures for users interacting with the TYFLOS prototype. By recording the number of times each participant spoke a specific user command and the number of time that the system spoke a system command, the probabilities associated with each place in the Unified Grammar SPN (Figure 6.4) can be specified empirically as shown in Table 6.8.

<table>
<thead>
<tr>
<th>System State</th>
<th>User State</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0 Ready</td>
<td>U0 Read Command</td>
<td>0.16</td>
</tr>
<tr>
<td>P1 Reading</td>
<td>U1 Move Command</td>
<td>0.16</td>
</tr>
<tr>
<td>P2 Process Image</td>
<td>U2 Listening</td>
<td>0.05</td>
</tr>
<tr>
<td>P3 Paused</td>
<td>U3 Waiting</td>
<td>0.07</td>
</tr>
<tr>
<td>P4 Moving</td>
<td>U4 Stop Command</td>
<td>0.02</td>
</tr>
<tr>
<td>P5 Low Information</td>
<td>U5 Help Command</td>
<td>0.18</td>
</tr>
<tr>
<td>P6 Command</td>
<td>U6 Pausing</td>
<td>0.18</td>
</tr>
<tr>
<td>P7 Help</td>
<td>U7 Take Picture Command</td>
<td>0.00</td>
</tr>
<tr>
<td>P8 Waiting</td>
<td>U8 Moving</td>
<td>0.18</td>
</tr>
<tr>
<td>P9 Change Level</td>
<td>U9 Change Level Command</td>
<td>0.00</td>
</tr>
<tr>
<td>P10 Change Voice</td>
<td>U10 Change Voice Command</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Table 6.8 System and User States in SPN with Probabilities

Analysis of the response times for each participant reveals that experience improved the participants’ ability to answer the questions. The response time was an average of 700 seconds for the first set of questions and improved to an average of 402 seconds for the second set of questions. Satisfaction also
improved from an average score of 3 for the first set of questions to an average score of 1 for the second set. Thus, the last experiment provided a SPN model of the system based on empirical information and demonstrated that the use of the interface improved with experience. (See Appendix D for details of statistical significance.)

6.6 Conclusion

In this chapter the two evaluated grammars that were developed in Chapter 5 served as the basis for the development of a Stochastic Petri-Net model of the interaction. This model began as a simple model of the User Command Grammar and grew to include states that represented the system and the user, as was modeled in the Human-in-the-Loop model (Figure 5.2). The changes of the model and the associated probabilities with each place within the model were based on three user studies, two with sighted participants and a final one with blind participants. The result of these studies is a model that begins to represent the complete interaction of a user with an eyes-free mobile reading device.
Chapter 7 Conclusion and Future Work

This final chapter of this dissertation provides a few closing thoughts on the work presented here. Section 7.1 summarizes the conclusions of the work presented in this dissertation. This is followed in Section 7.2 with a description of further research in both document image processing and user interface enhancements that could be incorporated into the TYFLOS prototype. The chapter concludes in Section 7.3 with a summary of the contributions made in the research performed for this dissertation.

7.1 Conclusions

The research described in this dissertation is focused in two separate fields that are brought together in the context of a mobile reading device for the visually impaired. One field of focus is on document image processing techniques that correct a document image in order to improve OCR results, as well as the segmentation, classification, and registration methods used to create an XML representation of the text of a document image.

The other field of focus is on the modeling and prototyping of a voice user interface. Clearly the five user studies described in Chapters 5 and 6 directed the development of the model and prototype. However, the user analysis and reading task analysis described in Chapter 1 also played an important role in identifying the source for significant contribution. It is this observation that leads to a significant conclusion of this research, namely that assistive devices cannot be designed in a vacuum, but must be guided by user studies and supported by target user feedback.

7.2 Future Work

TYFLOS is an assistive prototype and an evolving system. It must continue to adapt to the needs of the users. This is possible only through a continued emphasis on user studies that include blind and visually impaired users. The development of the prototype should only be constrained by technological limitations. Feedback from people interacting with the system should serve as the ultimate guide in the continued research and development of this project.

7.2.1 Document Image Processing

While much work has been performed over the past 20 years in the various components of document image analysis, there is still work to be done in the application of such components to understanding
infographics, finding text in images and video, optimizing techniques for use on mobile devices, and creating robots that can read [Letourneau 2004].

An application of document image analysis techniques that is open for extensive research is the development of systems that recognize and understand information presented in infographics (i.e. charts, graphs, tables). Infographics are effective visual representations because they can explain information simply and quickly using a combination of text and graphic symbols. Two approaches to this problem have been proposed, one based on understanding the graphic itself, the other based on analysis of the context around the graphic.

Carberry, et al. [Carberry, et al. 2003] propose that the understanding infographics can be treated as a discourse problem in the sense that it requires assimilating information from multiple knowledge sources such as text and graphics. In the method proposed by Huang and Tan [Huang and Tan 2007] the text in the graphic is associated with textual content in the rest of the document image. Thus, the graphic is interpreted in light of supporting text.

Often, authors refer to images without explanation in the text, and thus charts and graphics are relatively inaccessible to the blind community. The addition of this feature to the TYFLOS prototype would provide information to the visually impaired that is currently mostly unavailable.

Another related area of research is finding text in images and video. Finding text within images and video scenes has applications in robotics, vehicle navigation, and aids for the visually impaired. The identification and extraction of textual information from images and video involves the detection, tracking, and recognition of text within a given image. Jung, et al. [Jung, et al. 2004] presents a survey of approaches to this problem.

Clark and Mirmehdi [Clark and Mirmehdi 2002] present two different methods to locate text within an image. The first approach identifies rectangular boundaries around text in order to locate a surface that contains text. The rectification of this surface then provides a fronto-parallel view of the text. The second approach utilizes a neural network to identify regions of text in the image. Similar to the first approach, the rectification of a region provides a view of the text.

Finding text in video not only has the same problem as still images with the identification of text within the scene, but also must match the same text from multiple frames to create a complete view of the text. Uchida et al. [Uchida et al. 2008] present a method for the registration and recognition of text in
video. This approach utilizes a dynamic programming-based algorithm to optimize the rectification process required to perform intra-frame matching of text regions.

An important finding from the user studies with the TYFLOS system presented in Chapter 6 is that it is difficult for the user to hold the document steady from one picture to the next. Thus, a very relevant focus of research is to optimize the document image pipeline such that it can be performed real time when processing video. In this way the feedback can be provided to the user when it is most relevant.

Applications such as identifying text in scenes and translation from one language have a broader application to anyone who travels internationally, as well as to the blind. Thus, an additional area of research in the field of document image analysis is the optimization of these algorithms for use on mobile devices. Watanabe et al. [Watanabe et al. 2003] propose a camera that identifies Japanese text in a scene and translates the recognized text into English.

Another motivation for optimizing techniques for use on mobile devices is the development of mobile reading devices for the visually impaired. knfb Reading Technology, Inc. [knfb 2008] offers three mobile reader products on the market. Each of these products offers a set of features based on document image analysis techniques. However, advancements to the usability, reliability and robustness of these products will improve the user experience and account for image distortion caused by image resolution, illumination, perspective defects, and page curl.

To this end work continues on mobile image processing devices. For example, Thillou, et al. [Thillou, et al. 2005] propose optimized text segmentation, characterization, and region clustering algorithms for use on a personal digital assistant (PDA). This method identifies text in a scene based on three observations: 1.) characters inherently contrast with their background, 2.) the spatial cohesive nature of text, and 3.) latent information found within text such as frequency and orientation. Once textual regions have been identified within a scene, the region is cleaned and sent to an OCR for further processing.

Hannaksela, et al. [Hannaksela, et al. 2007] have begun to address image resolution issues by proposing an interactive image capture method. This approach utilizes a motion estimation method to enable user control of the device. High-resolution images are captured and stitched together based on knowledge of the movement of the camera.
7.2.2 User Interface

As outlined in Chapter 1, user interaction with document image analysis systems is an important factor when considering the implementation of a device for the blind community. Unfortunately, many of the techniques used for document image rectification require the user to understand the lighting, contrast, or other characteristics of the captured image. One method for making such systems more usable would be the development of adaptive systems. An adaptive system should identify potential problems with an image and work with the user to either capture a higher quality image or adjust required parameters without user input to generate the best image available.

Another interesting problem identified during the user studies described in Chapters 5 and 6 is that TYFLOS may be able to direct the user on the direction to move the document, but it cannot tell the user how far to move the document. Further research directed at enhancing the targeted registration technique may lead to a method for supplementing directional information with a value.

A comparative study of the TYFLOS prototype with other available devices such as OpenBook [Freedom Scientific, 2010] and the knfbReader would also be interesting to discover how the feedback of the TYFLOS prototype would improve the document reading experience for the visually impaired.

In Chapter 1 it was identified that visually impaired users of a mobile reading device want to interact with the device through a tactile interface. The tactile interface is important for situations when audible commands are inappropriate (classroom, conference room, workplace, etc.). Thus, an important field of research is the development of a model of a tactile interface. As noted in Chapter 1, Gaudissart, et al. [Gaudissart, et al. 2004] present a novel tactile human-device interface, but more research could be directed toward the growing field of eyes-free tactile interaction.

7.3 Contributions

Several contributions were made through the research performed to complete this dissertation. Specific contributions were made in the context of one underlying contribution, namely the modeling and prototype of a mobile reading device that supports regression and find-ability, provides spatial cues, and enables a user to interact through a Voice User Interface (VUI). As demonstrated in Chapter 1, no commercial or research prototype currently available addresses these important features.
The contributions to the field of document image processing include a survey of the current state of
document image processing techniques used to convert text to speech for the visually impaired (Chapter 2)
[Keefer and Bourbakis, 2011a], a document image enhancement method (Section 3.2) [Keefer, et al.,
2009c], a document image three dimensional perspective correction and dewarping technique (Sections 3.3
and 3.4) [Keefer, et al., 2009c], a document image segmentation method (Section 3.5) [Keefer, et al.,
2009b], a headline classification technique (Section 3.6), a segmentation aggregation method (Section 3.7)
[Keefer and Bourbakis, 2011c], a targeted registration method (Section 4.3) [Keefer and Bourbakis, 2009a],
and the creation of a composite XML document from a set of document images (Section 4.4).

The contributions to the field of user interaction are the development of an eyes-free interaction model
for interacting with a mobile reading device. Specific contributions include the evaluation of a user
command grammar (Section 5.1) [Keefer, et al., 2010], the evaluation of system commands grammar
(Section 5.2) [Keefer, et al., 2011b], and a voice user interface modeled with a Stochastic Petri Net
(Chapter 6) [Bourbakis, et al., 2008].
Appendix A - Personas

A persona is a short description of a fictitious user of a system. The description illustrates the personality of a user and his or her goals in such a way that a developer of a technology is able to infer a user’s course of action. As a design tool, personas are a powerful way to communicate behaviors, goals, wants, needs, and frustrations.

The creation of the following personas was based on analysis of the goals, desires, needs, and frustrations of visually impaired readers interviewed early in the development of the TYFLOS prototype. These personas were used to guide the design decisions made throughout the development of the TYFLOS prototype.

A.1 Student - Michelle Green

Michelle is a full-time student at Covington University in Covington, Minnesota. She is a senior majoring in International Studies, with hopes of going to law school upon graduation.

Michelle lost her sight as the result of a bout with meningitis when she was 12 months old. She lives with her aging parents who are very supportive and make sure that she is able to get to school on days that the public transportation is not operating.

Michelle uses a long cane as her primary navigation aid, and carries a medium sized saddlebag for her books and personal belongings, including her BrailleNote note-taking device.

Due to the demands of her classes, Michelle spends a lot of her time reading material related to political and historical interests. She uses several devices to facilitate this reading, including Jaws (a screen reader on her computer), a BrailleNote (a portable device that will read text files), and sometimes the Kurzweil 1000 scanning software for books or papers that she scans into her computer. Michelle has over 5 years of experience using each of these technologies, and has worked hard to master the long list of key combinations in JAWS in particular, and she is proud of her speed and accuracy.

While Michelle has never used a mobile reading device, she is very interested in learning how to use such a device, and does not mind taking time to learn to use new technology that will help her read more quickly.
A.2 Senior - Richard Browning

Richard was an engineer at NCR Corporation during the early development of automatic teller machines. He is the primary inventor listed on 6 patents that were critical for the first ATMs to work properly.

Through living a frugal lifestyle, and a few choice investments Richard was able to retire in 1990 at 54 years old. While he does not develop new technology any longer, Richard has maintained an avid interest in technology for the past 20 years. As his Macular Degeneration worsened, Richard was able to learn to use different reading aids such as Jaws screen reading software and OpenBook scanning and reading software, both available from Freedom Scientific. Richard has never learned to read Braille, so his ability to read independently is limited to what can be read to him by his computer.

Richard’s son Anthony takes him out to eat one night a week, and to run errands on the weekend. However, because Richard is able to live independently, he walks daily around the neighborhood and through the nearby park. He uses a long cane and a guide dog named Scout as his primary navigational aids.

Richard has never used a mobile reading device, but likes the idea of learning to use a new device. He hopes the device would help him read the mail, the newspaper, and menus at restaurants. He is willing to spend a little time to learn how to make use of a device if the device performs well and is easy to use.
## Appendix B - Reading Alternatives Comparisons

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Description</th>
<th>Price</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cicero Text Reader ReadingMadeEasy.com</td>
<td>Scans printed documents and turns them into speech or braille. Begins speaking as soon as the page scan is complete. Documents can be displayed in large print on screen and can be edited, saved, and printed.</td>
<td>$395</td>
<td>FBS</td>
<td>B</td>
</tr>
<tr>
<td>Complete Reading System ReadingMadeEasy.com</td>
<td>Scans printed documents and turns them into speech. User interaction includes spoken menus and does not require a screen reader.</td>
<td>$200</td>
<td>FBS</td>
<td>S</td>
</tr>
<tr>
<td>ezVIP JBliss.com</td>
<td>A simplified version of VIP that can be set up to scan and save a single page or up to 10 pages. Scans printed documents/pictures and turns them into speech.</td>
<td>$175</td>
<td>FBS</td>
<td>S</td>
</tr>
<tr>
<td>Kurzweil 1000 KurzweilEdu.com</td>
<td>Scans printed documents and turns them into speech. Scanned text can be saved for future reference and modification.</td>
<td>$995</td>
<td>FBS</td>
<td>M</td>
</tr>
<tr>
<td>Kurzweil 3000 KurzweilEdu.com</td>
<td>Scans printed documents and turns them into speech or displays them on the computer screen.</td>
<td>$1,495</td>
<td>FBS</td>
<td>M</td>
</tr>
<tr>
<td>OpenBook FreedomScientific.com</td>
<td>Scans and converts printed documents into an electronic format. Provides two text-to-speech software synthesizers.</td>
<td>$995</td>
<td>FBS</td>
<td>S</td>
</tr>
<tr>
<td>Optical Braille Recognition IndexBraille.com</td>
<td>Scans braille documents and converts them to text with a Hewlett-Packard scanner. The text can then be used in various applications including screen readers.</td>
<td>$1,395</td>
<td>FBS</td>
<td>P</td>
</tr>
<tr>
<td>Text Cloner Pro ReadingMadeEasy.com</td>
<td>Scanning software designed to work with a user’s existing screen reader. Has two different scanning methods: high-speed and high-detailed.</td>
<td>$100</td>
<td>FBS</td>
<td>S</td>
</tr>
<tr>
<td>VIP – Jbliss.com</td>
<td>Scans printed documents and turns them into speech. Scanned text can be edited, or used with CCTV with PC connection to display split screen camera image.</td>
<td>$295</td>
<td>FBS</td>
<td>S</td>
</tr>
</tbody>
</table>

*Table B.1 PC-Based Reading Systems (FBS-Flat Bed Scanner, SFS-Sheet Fed Scanner, B-Braille, EP-Emboss/Pntr, M-Monitor, P-Printer, S-Speech)*
<table>
<thead>
<tr>
<th>Product Name</th>
<th>Description</th>
<th>Price</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extreme Reader ER1</td>
<td>Four-button operation, high-contrast tactile keys and AT&amp;T's Natural Voices speech engine. A built-in CD player can read both DAISY and music CDs.</td>
<td>$3,125</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>GuerillaTechnologies.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extreme Reader XR1</td>
<td>A high-contrast keypad, and AT&amp;T's Natural Voices speech engine. A built-in CD player can read both DAISY and music CDs.</td>
<td>$2,925</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>GuerillaTechnologies.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extreme Reader XR10</td>
<td>A 31-operation keypad, high contrast tactile keys and AT&amp;T's Natural Voices speech engine. A built-in CD player can read both DAISY and music CDs.</td>
<td>$3,500</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>GuerillaTechnologies.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ovation</td>
<td>Scans and stores text of any kind and converts it to adjustable audio-output. Provides audio cues to signal the beginning and end of a successful scan.</td>
<td>$2,895</td>
<td>E</td>
<td>P</td>
</tr>
<tr>
<td>Telesensory.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portset Reader</td>
<td>Scans and reads text with British English speech incorporating a male and two female voices. Built-in floppy disk drive that allows for the transfer of any scanned document to a computer.</td>
<td>$2,750</td>
<td>P</td>
<td>S</td>
</tr>
<tr>
<td>Portset.co.uk</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SARA</td>
<td>Scans and reads in 29 human-sounding voices, 19 language dialects and 12 languages. The buttons are big and colorful with tactile shapes to differentiate one from another.</td>
<td>$2,795</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>FreedomScientific.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ScannaR</td>
<td>Scans text and reads it aloud. Includes speech and volume controls, storage for 500,000 pages at a time, and direct connection to the BrailleNote.</td>
<td>$2,995</td>
<td>A</td>
<td>LP</td>
</tr>
<tr>
<td>Baum.de</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simon Reading Machine</td>
<td>Scans and reads text. Tactile user controls include adjustments for start, stop, volume, speed and pitch.</td>
<td>$2,295</td>
<td>P</td>
<td>S</td>
</tr>
<tr>
<td>SensoryTools.com</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Table B.2 Stand-alone Reading Systems (A - Audio, E-Electronic, P-Print, S-Speech, M-Monitor)
<table>
<thead>
<tr>
<th>Product Name</th>
<th>Description</th>
<th>Price</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>knfbReader</td>
<td>Combines an off-the-shelf PDA with a digital camera. This handheld unit scans text and reads it aloud with synthetic speech. Minimal feedback is provided to the user in order to aid in aiming the camera. This feedback includes a spoken Field of View report, informing the user which of the edges of a document are within the camera's field of view and how large the image is in the viewing area.</td>
<td>$2,595</td>
<td>E</td>
<td>S</td>
</tr>
<tr>
<td>knfbReader Mobile</td>
<td>Utilizes the Nokia N82 or 6220 cell phones each of which contains a digital camera. This unit scans text and reads it aloud with synthetic speech. Minimal feedback is provided to aid the user in aiming the camera. This feedback includes a spoken Field of View report, telling the user which of the edges of a document are within the camera's field of view and how large the image is in the viewing area.</td>
<td>$995</td>
<td>E</td>
<td>S</td>
</tr>
<tr>
<td>MobilEyes</td>
<td>Combines an off-the-shelf hand-held computer with a digital camera. This portable reading machine scans text and reads it aloud with synthetic speech.</td>
<td>$3,500</td>
<td>E</td>
<td>S</td>
</tr>
<tr>
<td>Intel Reader</td>
<td>Device specifically designed as a mobile reading device. This handheld unit scans text and reads it aloud with a speech synthesizer. The feedback includes a screen that displays the text, and highlights the words as they are read.</td>
<td>$1,499</td>
<td>E</td>
<td>S</td>
</tr>
</tbody>
</table>

*Table B.3 Commercial Mobile Reading Devices (E-Electronic, P-Print, S-Speech, M-Monitor)*
## Appendix C - Analysis of OCR Accuracy Results

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Ratio</th>
<th>Prob &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProcType</td>
<td>17</td>
<td>255624.45</td>
<td>15036.7</td>
<td>18.7254</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>Image</td>
<td>22</td>
<td>179321.98</td>
<td>8151.0</td>
<td>10.1505</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>Error</td>
<td>374</td>
<td>300326.78</td>
<td>803.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C. Total</td>
<td>413</td>
<td>735273.21</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Table C.1 Analysis of Variance of Headline OCR Accuracy*

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Ratio</th>
<th>Prob &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProcType</td>
<td>17</td>
<td>51254.59</td>
<td>3014.98</td>
<td>8.8206</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>Image</td>
<td>24</td>
<td>130959.06</td>
<td>5456.63</td>
<td>15.9639</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>Error</td>
<td>408</td>
<td>139458.69</td>
<td>341.81</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C. Total</td>
<td>449</td>
<td>321672.34</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Table C.2 Analysis of Variance of Body OCR Accuracy*
Appendix D - T-Test Results of User Studies

Figure D.1 Eight Sighted Person Study Results

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean of Trial 1</td>
<td>685.875</td>
<td>t-Ratio</td>
<td>-5.5002</td>
</tr>
<tr>
<td>Mean of Trial 2</td>
<td>463.625</td>
<td>DF</td>
<td>7</td>
</tr>
<tr>
<td>Mean Difference</td>
<td>222.25</td>
<td>Prob &gt;</td>
<td>0.0009</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>t</td>
</tr>
<tr>
<td>Std Error</td>
<td>40.4076</td>
<td>Prob &gt;</td>
<td>0.9995</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>t</td>
</tr>
<tr>
<td>Upper 95%</td>
<td>126.7</td>
<td>Prob &lt;</td>
<td>0.0005</td>
</tr>
<tr>
<td>Lower 95%</td>
<td>317.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correlation</td>
<td>0.75412</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table D.1 Eight Sighted Person Study Results
Table D.2 Six Visually Impaired Person Study Results

<table>
<thead>
<tr>
<th></th>
<th>Value</th>
<th></th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean of Trial 1</td>
<td>700.333</td>
<td>t-Ratio</td>
<td>-2.99079</td>
</tr>
<tr>
<td>Mean of Trial 2</td>
<td>402.667</td>
<td>DF</td>
<td>5</td>
</tr>
<tr>
<td>Mean Difference</td>
<td>279.67</td>
<td>Prob &gt;</td>
<td>t</td>
</tr>
<tr>
<td>Std Error</td>
<td>99.5278</td>
<td>Prob &gt; t</td>
<td>0.9848</td>
</tr>
<tr>
<td>Upper 95%</td>
<td>41.822</td>
<td>Prob &lt; t</td>
<td>0.0152</td>
</tr>
<tr>
<td>Lower 95%</td>
<td>553.51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correlation</td>
<td>0.55915</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure D.2 Six Visually Impaired Person Study Results
References

AI Squared. 2010. www.aisquared.com


Kurzweil Educational Systems. 2010. www.kurzweil.edu

I2S. 2008. www.i2s-bookscanner.com


