A Document Reconstruction System for Transferring Bengali Paper Documents into Rich Text Format

Anirban Ray Chaudhuri   Debnath Singh   Mita Nasipuri   Dipak Kumar Basu

Abstract

The transformation of a scanned paper document into an editable form suitable for further processing such as desktop publishing or archiving in a digital library is a complex process. It requires solutions to several problems – document analysis by acquiring knowledge of document layout by a Page Layout Analyzer (PLA), followed by document recognition, which mainly comprises text recognition by Optical Character Recognition (OCR). Besides these two, another important problem is document reconstruction by transforming content into an electronically editable format by keeping the original layout intact. Core OCR modules exist on different Indian scripts, but no such document reconstruction system is available for Indian scripts. The document reconstruction system reported in this paper is the first of its kind on Indian scripts and it addresses document reconstruction for Bengali document images. The system makes use of the knowledge of both document layout extracted by a PLA in a graphical user interface (GUI) and the results of text recognition steps performed by OCR for transformation of paper documents into Rich Text Format.

Keywords : Indian Scripts, Desktop Publishing, Page Layout Analysis, Optical Character Recognition, Document Reconstruction, Encoding Standard, Indian Language.

0. Introduction

For the digital archiving, the most rudimentary way to make scanned documents accessible is to insert the document images in an MSWord document or as an attachment to HTML pages, after having converted into supportable digital format (e.g., GIF or JPEG). In this way, the information present in the input document could be preserved. However, this approach presents several disadvantages.

1. Compressed raster images are still quite large and their transfer can be unacceptably slow.
2. The content can only be viewed but is not editable.
3. Information retrieval based on ‘keywords query and searching’ is not possible.
4. For desktop publishing, to access the content in editable form, manual entry of the textual content (e.g., title, abstract and even the whole document) is required.
5. For multi page documents, pages can be presented only in a sequential order, thus missing the advantages of a hypertext structure necessary for document browsing in a digital library.

In view of the recent advances in information and communication technologies viz., in the frontiers of information retrieval, fast and noiseless data transfer and document archiving, there is an urgent need for tools that are able to transform data presented on paper into an editable form. This demand has been considerably met for European scripts and some of the Asian scripts [1,2]. Now there is an increasing demand for the same for Indian scripts. Looking at the growing needs of the same for the last few years, government agencies like Department of Science & Technology and Ministry of Information & Technology sponsored several turnkey projects at major R & D organizations and maximum emphasis has been given to Devanagari and Bengali, the 3rd and 4th most popular scripts of the world respectively. Recently commercial Optical Character Recognition (OCR) systems on Devnagari Script have also come up in the market. OCR systems also exist on other major North Indian scripts such as Gurumukhi, Marathi, Assamese and Oriya [3,4]. These OCR systems are still in a nascent form; particularly, document reconstruction part receives the minimum attention due to non-availability of appropriate tools such as fonts with standardized Font driver and editor support. These systems may be considered as core OCR modules as they can handle document images containing no other entities except text in a single column. These modules can save scanned documents in plain text format where no style sheet is associated. As a result, their appearance is not similar to the original document. However, for the sake of original document layout preservation, manual modification of Font-size and paragraph attributes like indentation, justification, line spacing, insertion of graphical components and regeneration of text in multicolumn are very tedious and time consuming.

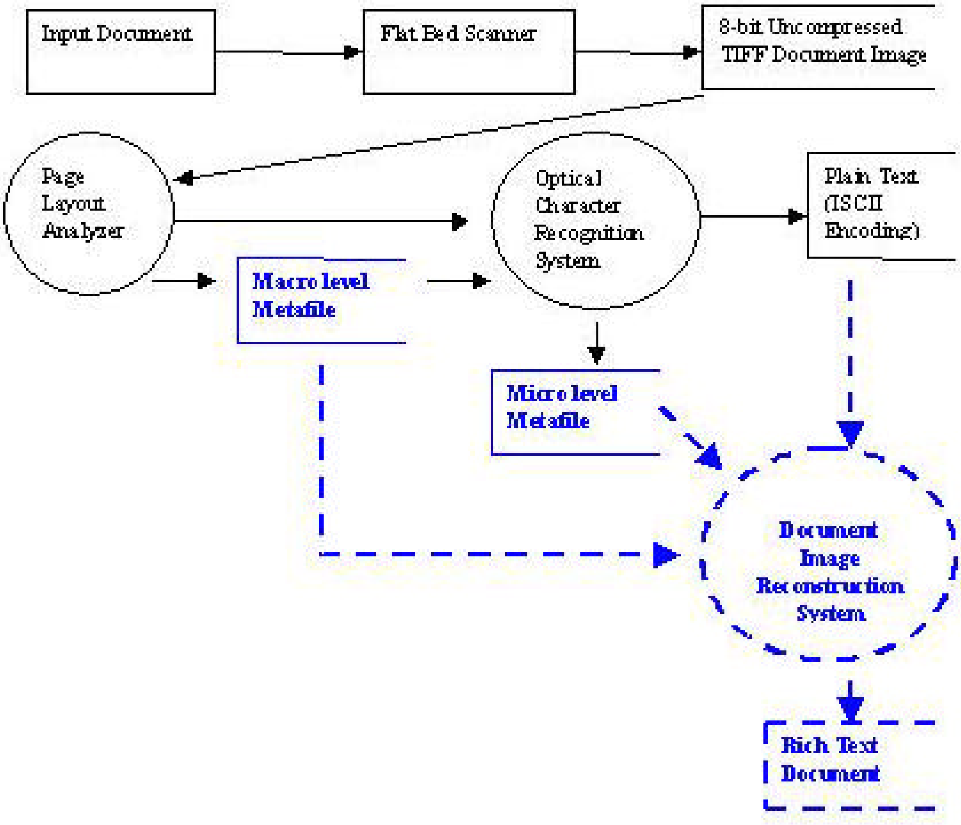
Input Document
Flat Bed Scanner
8-bit Uncompressed TIFF Document Image
Page Layout Analyzer
Optical Character Recognition System
Plain Text (ISCII Encoding)
Macro level Metafile
Micro level Metafile
Document Image Reconstruction System
Rich Text Document

Figure 1: The automatic generation of the structured rich text documents from document images.

It may be noted that most of the North Indian scripts such as Devnagari, Marathi, Bengali, Gurumukhi and Assamese are based on Brahmi based scripts and have very similar characteristics. As a result, extension of one OCR system in any of these scripts could be easily adopted by the OCR systems of the other scripts.

In this paper, we report the status of the ongoing project, “Anulikhan” on Brahmi based script in the context of Bengali document reconstruction. The objective of the project is to upgrade the core OCR into an advanced document image processing system where original layout of the input document image, including document formatting could be preserved in an editable electronic version. The reported system on document reconstruction transforms the document images into Rich Text Format (RTF), the most acceptable format that can represent document of arbitrary complexity. The system preserves the document structure in the final output as a RTF file by aggregating textual, graphical, and document formatting layout that is extracted by the PLA and the core OCR.

The paper is organized as follows. Section 2 presents an overview of the complete document image processing system. In Section 3, implementation details of document reconstruction system are presented. Performance evaluation of the proposed system is provided in Section 4. Section 5 concludes the paper and indicates the directions of the future work.

1. The Complete System Overview

The Figure 1 illustrates the overall process of the automatic generation of the structured rich text documents from document images. Prior to document reconstruction, the Page Layout Analyzer (PLA) performs layout structure analysis on the input document image and classifies document entities into texts, images and rulers/separators. It first coverts the image from gray tone to two tone and corrects the skew, if required. For gray tone to two-tone conversion, we use a simple histogram based thresholding approach [5,6]. A small amount of skew in the document image can be automatically corrected with projection profile and Hough transformation techniques [7]. To handle arbitrary skew in multiple directions an intelligent Graphical User Interface (GUI), based on mouse controls, is designed. From the deskewed binary image, PLA localizes and extracts text regions as rectangular blocks containing moderate sized connected components satisfying homogeneity or compatibility in terms of headline, vertical line and other (textual) features particularly available in Brahmi based scripts [8]. Note that the headline is a horizontal line present in the upper part of the characters, establishing word-level connectivity; whereas, the vertical line lies just below the headline and spreading over the middle zone. These two features occur in most of the characters. Each homogenous textual block is then fed into the core OCR system. The core OCR system performs the following tasks: line-word-character segmentation, character recognition, followed by word formation. For line, word as well as character segmentation, a projection profile based technique is adopted, as shown in Figure 3 [9]. The character recognition module primarily uses a run-length encoding based template and closest target matching for character level recognition as described in [10]. The block-level output of the core OCR is coded in ISCII and stored in a plain text file. The plain text file is displayed in a plain text editor, embedded in the system and is used for online correction of the OCR output. To carryout the document reconstruction, two metafiles are primarily generated. One metafile is designed to store the information evolving out of the PLA module. Since it contains information at image/block/region level, this metafile may be designated as a macro level metafile. Each block generated by the PLA is described by the positional information of the two diagonally opposite corners. Also, based on the content type (textual or graphical information), each block is labeled as text, picture (halftone images or line drawings), horizontal line or vertical line. To store the character level and paragraph formatting information that are evolving out from OCR output, a metafile that might be designated as micro level metafile is designed. The ISCII encoded plain text file together with the two metafiles, described above, are fed into the document reconstruction system. Within the document reconstruction system, these two metafiles are integrated into a single metafile in a more compact and structured format. By using this metafile and the plain text file, the reconstruction system generates the final output in RTF. Figure 2 illustrates an overall process description of the system for document reconstruction.

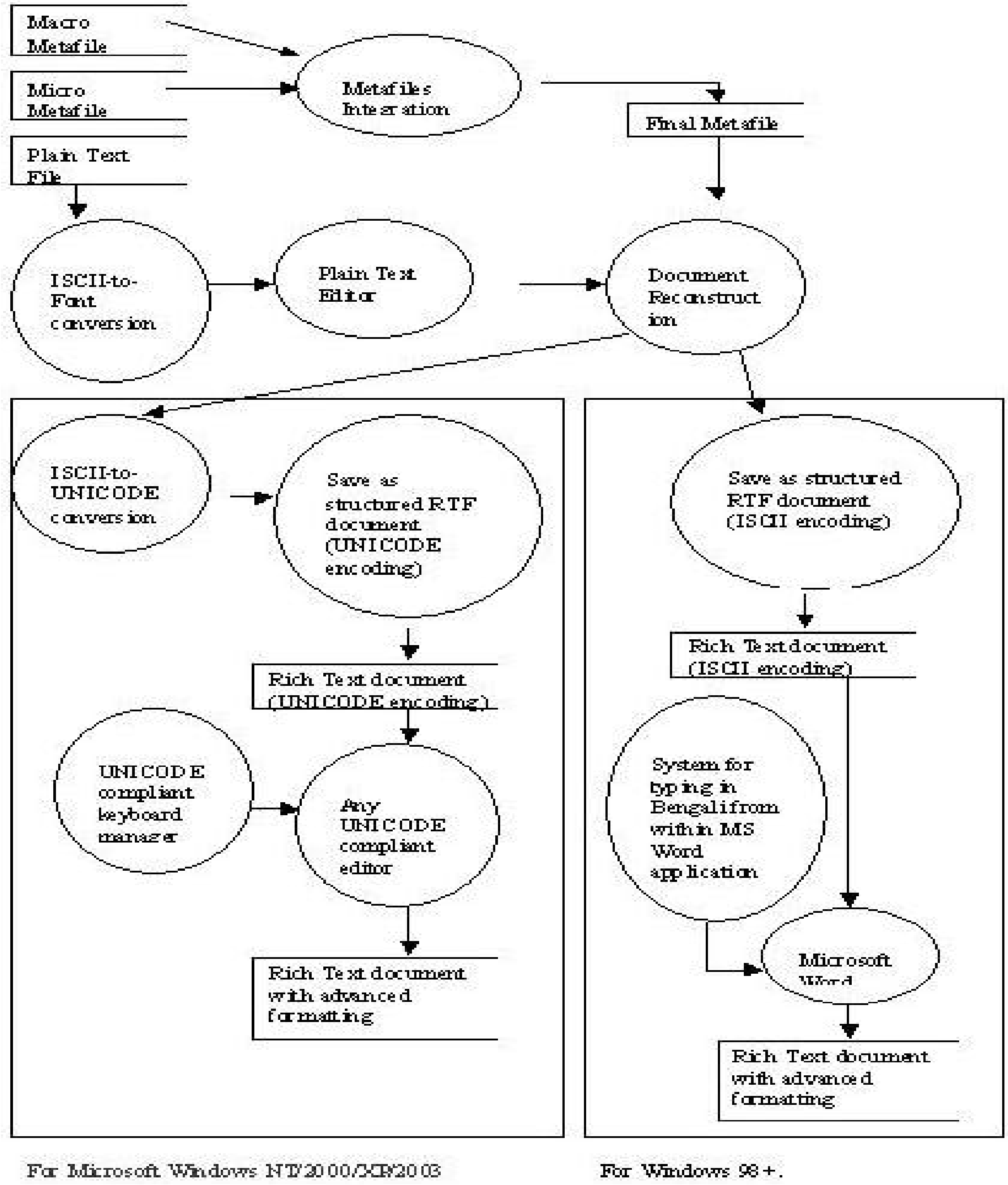
Macro Metafile
Micro Metafile
Plain Text File
Metafiles Integration
Final Metafile
ISCII-to-Font conversion
Plain Text Editor
Document Reconstruction
ISCII-to-UNICODE conversion
Save as structured RTF document (UNICODE encoding)
Rich Text document (UNICODE encoding)
UNICODE compliant keyboard manager
Any UNICODE compliant editor
Rich Text document with advanced formatting
Save as structured RTF document (ISCII encoding)
Rich Text document (ISCII encoding)
System for typing in Bengali from within MS Word application
Microsoft Word
Rich Text document with advanced formatting
For Microsoft Windows NT/2000/XP/2003
For Windows 98+.

Figure 2: An overall system for document reconstruction.

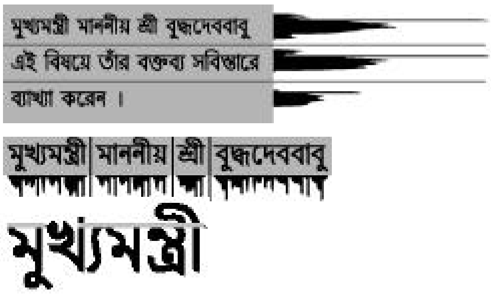

Figure 3 : Profile based method for line-word to character segmentation where matra and vertical bar features are extensively used.

2. Document Reconstruction System

Since the document reconstruction system has much functionality to perform, the entire system is designed in several stages. It reduces the complexity of the overall problem, thereby achieving an effective modular solution.

The stages are :

1. Font generation and Font driver:
   (a) Designing Indian Standard Code for Information Interchange (ISCII) compliant Bengali True Type Font (TTF).
   (b) Designing UNICODE compliant Bengali Open Type Font (OTF).
   (c) Designing keyboard layout and font driver for writing with the True Type fonts.
2. Font Conversion:
   (a) ISCII file to TTF file and vice versa.
   (b) ISCII file to UNICODE file.
3. Online correction of OCR.
4. Integration of metafiles.
5. Rich text generation.
6. Proofreading of the final output:
   (a) Designing a module for typing in Bengali from within Microsoft Word for correction of the final output (designed for Win98+).
   (b) Designing an UNICODE compliant keyboard managing system for typing Bengali.

2.1 Font Design and Font Driver Generation

(a) Designing ISCII compliant Bengali TTF

A TTF is designed for writing in Bengali. This is required due to non-availability of Bengali fonts having glyphs that support all conjuncts with standardized Font driver. A glyph set is designed for all Bengali characters. Note that in 8-bit character encoding scheme a maximum of 256 combinations are available out of which 32 are used by the system itself (e.g., tab key, alt key and enter key.). Remaining 224 combinations are not enough to accommodate all Bengali characters. Each basic character is represented by a single glyph and most of the conjuncts are displayed programmatically by concatenating two or more glyphs. Furthermore, the font is made ISCII compliant by placing the glyphs of the basic characters in appropriate or designated code point specified by ISCII.

After determining the basic Bengali character set, the dimension of the font i.e., the font attributes such as ascent, descent, leading, font name and weight are determined. For generating TTF, we use a standard font designing software, Fontographer 4^{TM}.

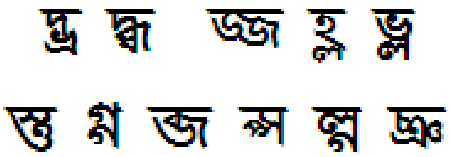

Figure 4: Some of the glyphs of conjuncts in the newly designed font that rarely

occur in other popular fonts.

(b) Designing UNICODE compliant Bengali OTF

OpenType fonts are also referred to as TrueType Open version 2.0 fonts, because they use the TrueType ‘sfnt’ font file format [11]. We design an OpenType font using Microsoft Visual OpenType Layout Tool (VOLT) that provides an easy-to-use graphical user interface to add OpenType layout tables to fonts with TrueType outlines. It supports a wide range of substitution and positioning types. It also contains a proofing tool that helps correcting the result of applying layout table lookups. We use the tool to add OpenType layout tables to our designed Bengali True Type fonts. Prior to adding the tables, the font is made UNICODE compliant by placing the glyphs of the characters in appropriate or designated code point specified by UNICODE.

(c) Keyboard Layout and Font Driver Design for writing with the designed TTF

÷ÅàÉ÷LaÏ

Figure 5. Typographical structure of a word.

Most of the keyboards available in India are designed for writing of English characters. For languages other then English, a soft keyboard layout design is necessary to determine which character of that language is assigned to which key. The keyboard layout is designed for our fonts and this becomes the foundation for writing the font driver program. Note that this font driver manages the mapping of the keyboard key to Bengali characters by assigning each Bengali character to a keyboard key. One key can be used to display two different Bengali characters with Shift key pressed and released. When a user enters a key, the key is trapped and appropriate Bengali character is displayed. Link-logic is for the identification of conjunct characters. A sequence of three or more keystrokes is necessary for display of a conjunct character. Conjunct characters are stored in a separate file and on detecting the link key at runtime, appropriate conjunct character is searched from the file and displayed.

2.2 Font conversion

In the absence of any standardization, each Bengali font has its own keyboard layout. Texts are being stored in font dependent glyph codes and the glyph-coding scheme for these fonts is not same. As a result, one cannot exchange the electronic Bengali documents from one font to another as conveniently as in English. To alleviate this problem, font conversion utilities are developed. One of the utility converts text encoded in ISCII to our system supported font and vice-versa; while the other one coverts text data encoded in ISCII to UNICODE and vice-versa.

(a) ISCII-to-Font and vice-versa

ISCII is a standard encoding scheme for Indian languages and a convenient way of exchanging information. This module performs conversion operation from ISCII to system supported font and vice-versa. The converter program accepts an ISCII file as input and for each character, it replaces the ISCII character code with the specific font character code. Next, the data is saved in a file. For font to ISCII conversion, the logic is same as above but in the reverse order.

(b) ISCII-to-UNICODE and vice-versa

The obvious disadvantage of coding an ISCII encoded data to a particular font is that the font must have its own font driver. In worst case, for N supported fonts we require N Font drivers. A more standard solution to this problem is to encode data in UNICODE. Once data is encoded in UNICODE, we can smoothly switch it between UNICODE compliant Bengali Open Type fonts, without worrying for Font driver. Therefore, a converter module has been designed to convert data between ISCII and UNICODE.

2.3 Online correction of OCR

The OCR output (the plain text file) may contain some word errors due to incorrect character and/or word recognition. A word error can belong to one of the two distinct categories viz., non-word error and real word error. A non-word error makes a word meaningless, while a real word error means an error that results a valid word but not the intended one in the sentence., thus making the sentence syntactically or semantically incorrect. Though some attempt is found on non-word error detection [12], no work is available on real word-error detection for Indian scripts.

The phrase “online correction of the OCR” implies prior to document reconstruction, support of an editor is required that enables user to correct the non-word and real word errors present in the OCR output. Using the standard Windows controls, such as buttons, menus, scroll bars, and lists we design the Editor for the above purpose. As we provide support for various text editing operations including different paragraph and character formatting, find/replace string etc., this editor is also used as a standalone ISCII compatible plain text editor. To help the new user to write in Bengali efficiently and quickly an on-screen keyboard is also designed.

2.4 Integration of metafiles

A module is designed to synthesis the macro and micro level metafiles. This module extracts necessary information for document reconstruction from the metafiles, and stores into a single metafile in a more integrated and structured format. The metafile so formed could be designated as final metafile. Besides the necessary block/region level information, the final metafile also contains important character level and paragraph formatting information for text regions. Association information needed to relate a text block with its corresponding textual information contained within the plain text ISCII file is also maintained within the final metafile.

2.5 Rich text generation

A module is designed to restore the physical layout and logical structure of the original document, based on labeled regions and recognized texts that we obtain as the result of layout analysis and text recognition. Using the block/region information contained within final metafile and as per RTF specification 1.7 [13] we reconstruct pictures and rulers/separators. To reconstruct text frame/block, the primary task is to recognize the font and its size. As shown in Figure 5, text line images are composed of three typographical zones – The ascender (upper), the x height (mid) and the descender (lower) zones, which are delimited by four virtual horizontal lines. While the ascender and descender zones depend on the text content, the x height zone is always occupied regardless of the characters that occur. The x height is commonly called midzone distance, and its proportion in the text height differs from one typeface to another.

A statistical classifier is designed, that makes use of typographical attributes such as ascenders, descenders, midzone distance, matra thickness and stroke width obtained from a word image (stored in the final metafile) to recognize font-shape-size. Another crucial issue is the identification of paragraphs along with the paragraph indentation and justification. Once the font-shape-size and paragraph attributes are identified, the system reconstructs text frames from the final metafile and the plain text file. Note that at present, no algorithm is designed and implemented to find the reading order of the text regions in Indian documents. We adopt a GUI based approach to tackle this issue for any complex layout. We generate the reading sequence by placing and clicking the mouse in the text blocks in an order.

2.6 Proofreading of the final output

Once the image is converted to rich text document, there might be two problems that need to be solved.

? There might still be some errors in text recognition results even after correction in online.

? Users might need to append/modify texts as well as layout of the original document and export it to another data processing software.

To address the above issues we design:

? A module to correct the OCR results. It supports typing in Bengali using one of the supported fonts, from within Microsoft Word to correcting the text recognition results. It is designed for Win98+ and operates on ISCII encoded text.

? An UNICODE compliant keyboard managing system for typing Bengali anywhere on UNICODE compliant Windows, to correct the UNICODE encoded text recognition results.

3. Performance Analysis

To assess the absolute performance of the document reconstruction system, a large number of document images of Bengali pages having multi-column layout with pictures rulers etc., from various resources should be tested. We are currently testing with a few samples of one hundred and twenty pages. Fifty scanned pages are directly generated from computer. Twenty pages are scanned from “Desh”, the most popular Bengali literature magazine. About fifteen pages of cutting are taken from “Anaandabazar Patrika’, which has world-wide the maximum readership as a daily Bengali newspaper; whereas, about ten pages are derived from “Bartaman”, another daily Bengali newspaper with the second largest readership. Rest is taken from popular Bengali books and other type of publications. Pages from hardcopies are scanned into 8-bit uncompressed tiff image at 300 dpi resolution. The input and output of the document reconstruction system at different stages of the algorithm are shown in Figure 6.

As far as document reconstruction system is concerned, the results are quite satisfactory. Text blocks, pictures and rulers are correctly reconstructed wherever PLA and OCR provide correct information. In a very few cases incorrect character level and paragraph formatting information result by the reconstruction module. The error evolving out of the PLA and text recognition system produces erroneous font metric estimation besides wrong font and font size recognition.

For about 3% of the data, incorrect blocks are generated by PLA mostly due to small paragraphs of two-three lines with poor text alignment. So far, we do not pay much attention to the core OCR. Only about ten percent of the sample document images are learnt for the template generation. Only our designed fonts are learned thoroughly. Consequently, the accuracy of OCR falls rapidly for documents whose character/ fonts are not well represented by the template database. Particularly, OCR results for old print documents (less than 3% of the pages) are unacceptable due to several problems including poor object background segmentation, touching characters, noise and lack of representation in the template database.

4. Conclusions and Direction of the Future Work

The presented advanced document image processing system transforms Bengali printed documents with its original layout into Rich Text Format. To the best

of our knowledge this is the first initiative on any Indian script. There are several significant benefits to this transformation: the transformed documents can be edited, reformatted, appended into other documents or converted into HTML version for the access via Internet more quickly than the original bitmap image. The user can manipulate the original document and be able to make hypertext document and automated information retrieval that is necessary for document browsing. The potential beneficiaries of our system are – newspapers (printed in Bengali script) and other publishing houses, libraries (digital library generation), offices looking for office automation (document archiving), linguistic community (for creating corpus) and blind people (as automated reading aid). As the development is based on VC++, all the code is easily portable. In addition, since the object-oriented concepts are used for development, the system can be easily expanded as per requirement. System encodes the output in 8-bit ISCII and UNICODE encoding format and hence, outputs can be viewed or edited with any third party editor that supports ISCII and/or UNICODE. However, the system is still far from its final shape. The limitations and future scopes of the present system are :

? Due to reduction in cost and rapid modernization, present publication media uses multiple colors for text and shading in the background. Simple thresholding techniques that are currently being used by OCR modules are not appropriate for these documents. Designing of a more robust segmentation module for object background detection that could take care of degraded documents as well is started.

? For better representation and easy editing, PLA module as well as the macro metafile specification should be upgraded by merging of block level information to column level so that all blocks in a single column could be merged. In addition, by its very design, PLA is not effective for scripts where characters within word are not connected as in case of English. Note that in modern

Bengali publications, embedded English text of a few lines often presents. In such a situation, a script identifier is also required with the PLA.

? More effort is required for upgrading the core OCR for better accuracy. Segmentation of touching characters should be provided with the module. In addition some word level error correction by post processing should be developed. For easy learning of new templates and efficient template database management, we are planning to provide a GUI.

? For better representation of the reconstructed document, more fonts should be designed.

? More enhanced GUI for online correction as well as block level correction would be very helpful for the end user.

? We have also started designing a similar advanced document reconstruction system for Devnagari script.

5. Acknowledgement

Science and Engineering Research Council of Department of Science and Technology, Govt. of India is duly acknowledged for the “DST Fast Track” awarded to the first author and the work presented in this paper is the outcome of that project.

6. References

1. Nagy G. (2000), Twenty years of document image analysis in PAMI, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No.1, pp.38-62, 2000.
2. Ding X., Wen D., Peng L., Liu C. (2004), Document Digitization technology and its application for digital library in China, Proc. 1st International Workshop on document image analysis for Libraries (DIAL’04), IEEE Press.
3. Ministry of Communications & Information Technology, Government of India (2003), ViswaBharat January 03, http://tdil.mit.gov.in
4. Bansal V. and Sinha RM.K., (2001), A Devanagari OCR and A Brief Overview of OCR Research for Indian Scripts in Proceedings of STRANS01, held at IIT Kanpur.
5. Jain A. K. and Yu. B (1998), Document Representation and Its Application to Page Decomposition, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 20, No. 3, pp. 294-308.
6. O’Gorman L, and Kasturi R (1997) Document image analysis. IEEE Computer Society Press Executive Briefing Series, Los Alamitos, CA.
7. Cattoni, R., Coianiz, T., Messelodi, S., and Modena CM, (1998), ITC-IRST, Povo,Trento, Italy, Geometric Layout Analysis Technique for Document Image Understanding: A review,http://tev.itc.it/people/modena/Papers/DOCSEGstate.pdf
8. Ray Chaudhri A,. Mandal A, and Chaudhuri B. B., (2002), Page Layout Analyzer for Mulitingual Indian Documents, Proc. Language Engineering Conference, IEEE CS Press.
9. Chaudhuri B . B., Garain U. and Mitra M., Indian Statistical Institute (2003), On OCR of the most popular indian scripts: Devnagari and Bangla, TR/ISI/CVPR/03/2003.
10. Garain U. and Chaudhuri B.B., (1998), Compound Character Recognition by Run Number Based Metric Distance, Proc. SPIE Annual Symposium on Electronic Imaging, San Jose, USA, pp.90-97.
11. Microsoft (2003) Microsoft typrgraphy, http: //www.microsoft.com/typography/default.mspx
12. Chaudhuri B. B.and T. T. Pal (1998), Detection of word error position and correction using reversed word dictionary, Int. Conf. on Computational Linguistics, Speech and Document Processing, Calcutta
13. Microsoft(2003) RTF Specification 1.7, http://support.microsoft.com/kb/q86999/

About Authors

Dr. Ray Chaudhuri, Anirban born in Shatiniketan ('69), received Masters in Pure Mathematics ('91) and Ph.D. in Computer Science ('2000) from Visva-Bharati (A Central University, Shantiniketan) and Indian Statistical Institute (Kolkata) respectively. He started service as a Teacher at Patha-Bhavana, the school section of Visva-Bharati and afterwards in the capacity of a Lecturer at the same University, Research Scientist at Indian Statistical Institute and Post-Doctoral Research Associate at Coordinated Science Laboratory as well as at the Department of Electronics and Computer Engineering, University of Illinois at Urbana Champaign. In 2001, to have a more independent research and business-consultancy career, he left the permanent job. At present with a Fast Track Fellowship from Department of Science & Technology, Govt. of India, he is primarily attached with the Department of Computer Science and Engineering, Jadavpur University, Kolkata as a Principal Investigator in the Project "Anulikhon: Advanced Optical Character Recognition System for Bangla and Similar Brahmi Based Indian Scripts". His research interest includes Statistical Pattern Recognition in point pattern analysis and consistent estimation, Image Processing and Computer Vision in areas of automatic target recognition, remote sensing, volume visualization and document image analysis.

Sh. Singh Debnath, born in Kolkata (79), received a Bachelor in Science from Kolkata University 1999, and currently doing his Masters in Computer Application from DoE (B-level). He is also currently working as a Project Trainee at the Centre for Micro Processor Application, Department of Computer Science & Engineering, Jadavpur University.

Dr (Mrs.) Nasipuri Mita received her B.E.(Tel.E.), M.E. (Tel.E.) and Ph.D(Engg.) Degrees from Jadavpur University, Kolkata, India, in 1979,1981 and 1990 respectively. She is currently a Professor and Head of the Computer Science and Engg. Department of Jadavpur University. Her current research interest includes Computer Architecture, Image Processing, Pattern Recognition, Multimedia Systems, Bio-medical Signal processing etc. She had a large number of research publications in International/National Journals and International/National Conferences. She is a Senior Member of the IEEE, USA, Fellow, The Institution of Engineers (India), Fellow, West Bengal Academy of Science and Technology.

Dr. Basu Dipak Kumar received his B. (Tel.E.), M.E.(Tel.E.) and Ph.D (Engg.) degrees from Jadavpur University, Kolkata, India, in 1964,1966, 1969 respectively. He joined Electronics & Telecommunication Engg. Department of Jadavpur University as a faculty member in 1968. He is currently a Professor in the Computer Science and Engg. Department of the same University. His field of research interest includes Digital Electronics, Microprocessor Applications, Bio-medical Signal Processing, Knowledge Based Systems, Image Processing, Pattern Recognition, Multimedia Systems etc. He had a large number of research publications in International/National Journals and International/National Conferences. He is a former fellow of the AvH Foundation, Germany. He is a Fellow, the Institution of Engineers (India), Fellow, West Bengal Academy of Science and Technology and Senior Member of the IEEE, USA.