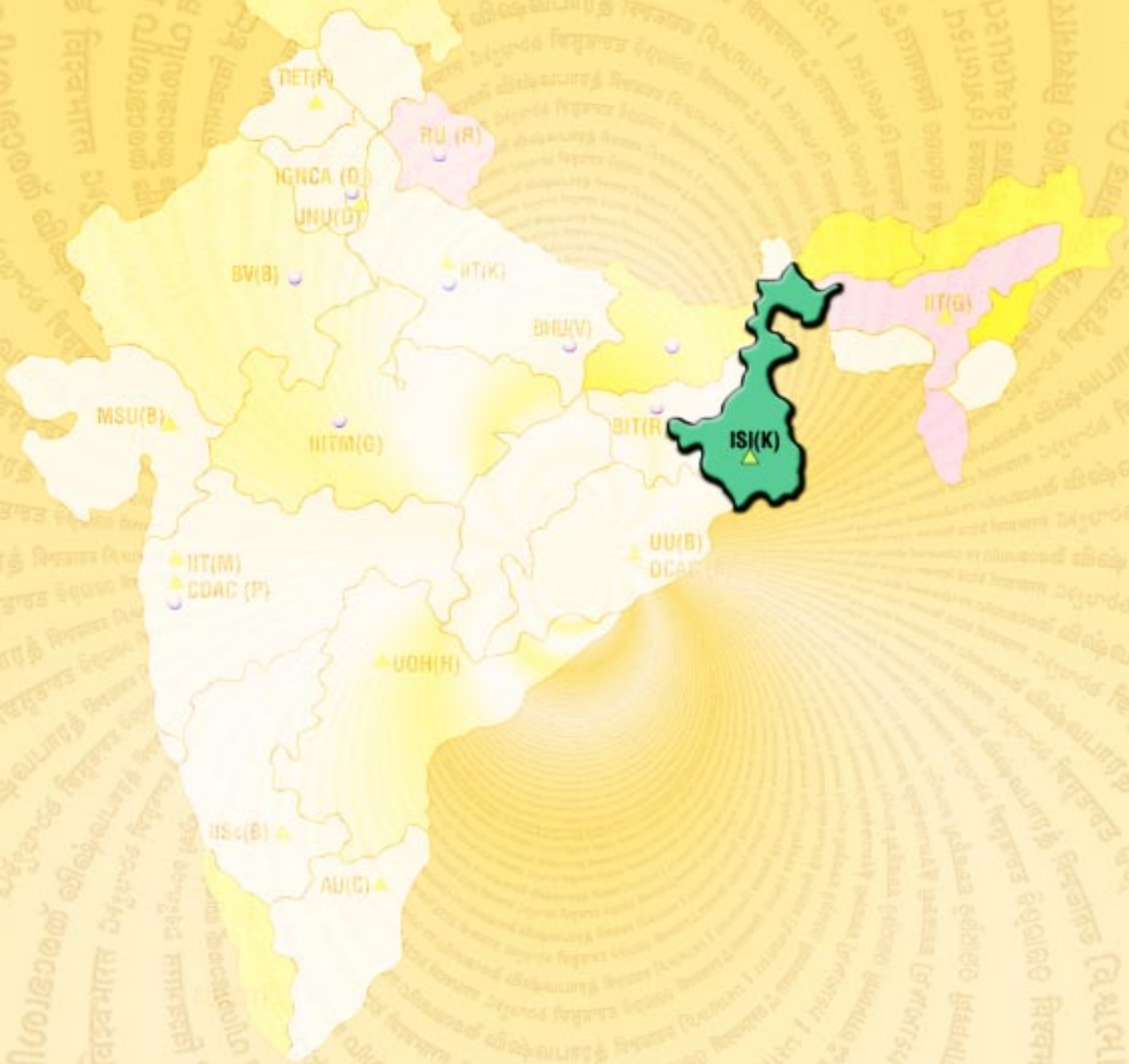
Resource Centre For
Indian Language Technology Solutions – Bengali
Indian Statistical Institute, Kolkata

Achievements



Computer Vision and Pattern Recognition Unit
203, Barrackpore Trunk Road
Indian Statistical Institute, Kolkata-700108
Tel. : 00-91-33-25778086 Extn. 2852
E-mail : bbc@isical.ac.in
Website : http://www.isical.ac.in/~rc_bangla

RCILTS-Bengali
Indian Statistical Institute, Kolkata

Introduction

Main objective of this project was resource and technology development for all aspects of language processing for Eastern Indian languages particularly, Bangla. It included Corpus development, Font generation, Website generation, OCR development, Information retrieval system development etc. as well as training of people on Indian language technology through courses and workshops.

Outcome in Physical Terms

- Corpus of Bangla Document Images, Bangla Text Corpus in Electronic Form and Corpus of Bangla Speech Data
- Web site on Eastern Indian Language Technologies including Bangla Language Technology design guide
- Bangla Font Package
- OCR System for Oriya and Assamese
- Information Retrieval System for Bangla Electronic Documents
- Multi-lingual Script Line Separation system
- Automatic Processing system for Handprinted Table-Form Documents
- Neural Network based tools for printed document (in eastern regional scripts) processing

Production Agencies with which Memorandum of Understanding/link up has been established

MoU has been signed with (i) Webel Mediatronics Ltd, Kolkata (speech synthesis Technology), (ii) Centre for Advanced Computing, Pune (Devanagari and Bangla OCR Technology) (iii) Electronics Research & Development Centre India, Noida (Devanagari OCR Technology), (iv) Orissa Computer Application Centre, Bhubaneswar (Oriya OCR Technology) (v) Indian Institute of Technology, Guwahati (Assamese OCR Technology) Link up for Data source has been established with Bangla Academy.

1. Core Activities

1.1 Web Site Development and The Language Design Guide

- **Creation of a web site on Eastern Indian Languages and Language Technologies for information of people interested in TDIL**

The name of this site is ***Resource Centre for Indian Language Technology Solutions –Bangla.*** The URL of this site is: ***http://www.isical.ac.in/~rc_bangla/.*** At present, this site contains details about this MIT project (Resource Centre for Indian Language Technology Solutions - Bangla), a brief description of the products developed at Indian Statistical Institute, Kolkata. A Bangla design guide is also introduced in the site so that new technology on Bangla could maintain a common framework. It covers details of the origin and development of the script, the alphabets, character statistics, information related to the fonts, presentation and storage consideration, information related to Unicode of Bangla etc. The local language academy (Bangla Academy) has been contacted. A prototype of a Web-based front-end to our spell-checker and phonetic dictionary has been developed using an evaluation copy of CDAC's GIST Software Development Kit (SDK). Initiative has been taken so that this front-end can be put on the Internet for public use with the GIST SDK and iPlugin (or Modular Software's Shree Lipi). Issues related to hosting this web site on another server/Web hosting service are currently being explored.

1.2 Training Programmes

- **Education through Training Programmes and Workshops**

Two International workshops under the banner of "International Workshop on Technology Development in Indian Languages (IWTDIL)" were held respectively during March 26-30, 2001 and January 22-24, 2003. Some main topics covered in these workshops were: (a) Machine translation from English to Indian Language (particularly to Bangla),

(b) Text corpus generation, designing and annotation, (c) Speech synthesis, processing and recognition, (d) OCR technology for Indian scripts, (e) Hand written character recognition, (f) Document processing and analysis, (g) Some general features of Indian Languages (e.g, phonology and acoustics of Bangla phonemes and diphthongs), (h) Anaphora resolution ellipsis in Hindi and other Indian Languages. In IWTDIL'01, the first three days had introductory talks and tutorials on various areas of language technology and the last two days featured lectures by international experts on these subjects. A total of twenty-nine speakers presented talks at the workshop and there were seventy-five participants from India and abroad. In IWTDIL'03, four distinguished scientists from abroad and six from India delivered lectures to forty-five participants. A cultural programme consisting of a play by a renowned theatre group of Kolkata was also arranged.

One National workshop entitled "Indian language Spell-checker Design" was organized in July 18-19, 2002. The main theme of the workshop was to present the work on Spell-checker done by various Resource Centres and groups in different Indian languages. The participants demonstrated the Spell-checker software developed by them. It was anticipated that a benchmarking method for spell checker would be evolved out of the workshop. Twenty-five participant attended/presented lectures in the workshop.

2. Services

2.1Corpus Development

- **Printed Bangla Document Images & Ground Truth for OCR and Related Research**

Some famous Bangla novels and books have been selected to prepare a Bangla document image corpus along with ground truth. A brief description of those selected novels and books are given below:

Maitreya Jatak – This is a famous Bangla mythological novel written by Bani Basu, a popular Bangla writer. As far as the linguistic aspect goes, the overall language of the book is old Bangla. Polished and chaste form (Sadhu bhasa) of narration is used through out the book and some archaic forms (terms mostly derived from Sanskrit and Prakrit sources) are also used mainly to create an atmosphere of the period of Goutama Buddha, which the book depicts. Ananda Publishers, one of the largest publishers of Kolkata has published this book. The quality of the printing is excellent. Computer generated font developed by Ananda Publishers has been used in printing.

Pratham Alo – This is also a famous Bangla novel written by famous Bangla writer Sunil Gangopadhyaya. The incidents of the novel are nearly one hundred and fifty years old and the theme is related to Bangla renaissance. The standard Bangla language has been used for this book. The percentage of archaic words is much less than that in Maitreya Jatak. This book is also published by Ananda Publishers, but using offset printing technology. The overall printing quality is good but worse than Maitreya Jatak.

Bhasa Desh Kal – The book is written by Dr. Pabitra Sarkar, a distinguished author of Bangla Linguistics. The language of the book is standard Bangla colloquial. The publisher of the book is Mitra and Ghosh, Kolkata. Printing quality is moderate and worse than Pratham Alo.

Upendra Kishore Rachana Samagra – This book is written by the famous author of juvenile literature Shri Upendra Kishore Roychowdhuri. Old formal and polished Bangla language is used in this book. Here, offset font is used and the book has been published by Ananda publishers of Kolkata.

Amar Jibanananda Abiskar o Anyanya – Sunil Gangopadhyaya is the author of this book. The language of the book is standard Bangla colloquial. Ananda Publisher has published this book. The printing quality is good and offset printing technology is used.

Rajarshi – This is one of the classic novels written by Rabindranath Tagore. Old formal and polished Bangla language is used in this novel. Ashoke Book Agency of Kolkata is the publisher of the book. The printing quality is not as good as that of Ananda Publishers.

Bangla Choto Galpa - The book is a collection of Bangla short stories written by different writers of old Bangla literature and edited by Rabindranath Tagore. Old formal language (Sadhu bhasha) is used in most of the stories. The book is published by Model Publishing House. The printing quality is good.

Punarujjiban – It is the Bangla version of a novel written by famous Russian novelist Lev Tolstoy. The book is published by Raduga Prakashan, Moscow. The printing quality is good and modern colloquial language is used.

Bhraman Amnibus – The book is written by Sri Uma Prasad Mukhopadhyaya. It contains detailed description of different places of Himalayas. Modern Bangla language is used in this book. It is published by Mitra & Ghosh, Kolkata. Offset printing technology is used.

Parakiya – This book is collection of Bangla short stories written by both old and modern writers of Bangla literature and edited by Sunil Gangapadhyaya. The publisher of the book is Punashcha. Both old formal and modern colloquial language are used in different stories of the book. The printing quality is not so good.

Amar Debottar Sampatty – The book is an auto biography of Nirod Chandra Chaudhuri, published by Ananda Publishers. Offset printing technology and old formal language are used.

Prabasi Pakhi - It is a Bangla story book written by Sunil Gangapadhaya and published by Ananda publishers. Offset printing technology and modern colloquial language are used.

Mahabharat Katha – The book is published by Udbodhan publication of Ramkrishna Mission. Modern Bangla language has been used in this book (Table 1 is the list of books scanned).

OCR – A HP Scan Jet flatbed scanner of high resolution (300dpi ordinary and gray level spectral resolution) has been used to get the image documents. The images are then saved into uncompressed TIFF format and are scanned generally using same setting. Minor adjustment of brightness and contrast may be done by using Corel Photopaint software.

Table 1

Name of Books	No. of Pages Scanned	Total No. of Words
Maitreya Jatak	419	2,43,020
Pratham Alo	37	18, 833
Bhasa Desh Kal	15	5,760
Upendra Kishore	330	1,38,270
Amar Jibanananda	21	7,812
Rajarshi	100	39,800
Bangla Chotogalpa	260	1,11,800
Punarujjiban	68	25,704
Bhraman Amnibus	31	14,198
Parakiya	163	85,412
Amar Debottar Sampatty	29	10,730
Prabasi Pakhi	10	4,020
Mahabharat Katha	4	1,304
Total	**1,487**	**7,06,663**

Those page-by-page scanned images are then fed as input of the Bangla module of the existing Bilingual (Hindi and Bangla) Optical Character Recognition (OCR) System developed in our department. Each character of those input images are being recognized by this OCR System and are stored as text documents in 8- bit ISCII format.

The present OCR system gives an accuracy of 96 % to 98 % in character level depending on paper quality and font style of the books. So, the 4% to 2% error is corrected manually to prepare the text documents, which serves as ground truth.

A benchmark software is also developed to give a specified format to all ground truths generated using Indian OCR technologies for automatic evaluation of them (different OCR technologies). The database formed in this way is shown in the tabular form in table1.

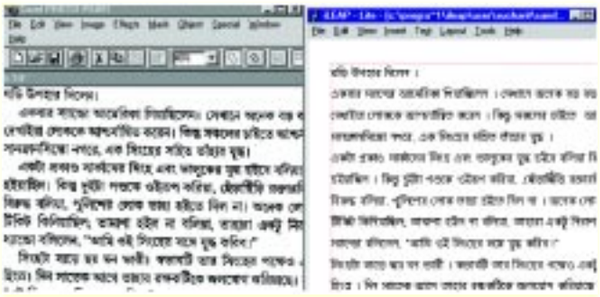


Figure 1: The image is at left hand side and the corresponding ground truth is at right hand side

• Development of Bangla Text Corpus in Electronic Form including a Bangla dictionary, and several Bangla classics

Several novels have been entered into the computer in ISCII format. The description of novels, the corresponding authors and total number of words are given in a tabular form (*Table-2*). More than 34000 words of a bilingual (Bangla-English) dictionary have also been entered. A comprehensive corpus for electronic dictionary for Bangla-Bangla (with 65,000 words) has been constructed and checked. Using the guidelines of the above two dictionaries a trilingual (*Bangla-English-Hindi*) dictionary creation has started. Till today 17,000 words with meaning and parts of speech as well as other information has been entered. However, lack of standard size Bangla-Hindi dictionary in printed form has created some problem (Bangla-Hindi dictionaries available in the market are small, containing about 7,000 words only). We have also designed a prototype of an electronic thesaurus for Bangla (based on WordNet, a well-known electronic resource for the English language).

Papers Published

- Dash, N.S. and Chaudhuri, B.B. (2001) "A corpus based study of the Bangla language". *Indian Journal of Linguistics*. 20: 19-40.
- Dash, N.S. and Chaudhuri, B.B. (2002) "Corpus generation and text processing". *International Journal of Dravidian Linguistics*. 31(1): 25-44.

Table -2

1. Author's Name

Iswarchandra Vidyasagar is known as one of the great social reformers, philanthropist and father of Bengali prose style of Bengal. He played an important role in primary education, widow marriage and women education.

Name of the Classic & Publisher

Vidyasagar Rachanabali (Tuli – Kalam)

Name of Articles (Story, Novel, drama, etc.)

Akhyanmanjari

Betal Panchabingshati

Kathamala

Mahabharat

Niskritilabh Prayash

Provaboti Sambhashan

Sakuntala

Sanskrit Bhasa O Sankriti

Sitar Banobas

Balyabibaher Dosh

Ramer Rajjyabhishek

Bidhababibaha

Jibancharita

Vrantibilas
Banglar Itihas
Charitabali
Bodhadaya

No of Words (per Classic)

Sub. Total : 315783

2. Author's Name

Bankim Ch. Chattopadhyay is considered the first and one of the greatest novelists in Bangla language. He has written 14 novels besides a large number of

essays on various literary and social issues. He wrote the 'Vande Mataram' song, which played a major role in Indian Independence struggle.

Name of the Classic & Publisher

Bankim Rachanabali (Patrajo Publication)

Name of Articles (Story, Novel, drama, etc.)

Anandamath
Bishbriksha
Durgeshnandini
Kamalakanta
Bigyanrahosya
Muchiram Gur
Lokrahasya
Samya
Bibidha Prabandha (Vol. 1)
Bibidha Prabandha (Vol. 2)
Krishnacharita (Vol 1)
Krishnacharita (Vol 2)

No of Words (per Classic)

Sub. Total : 320232

3. Author's Name

Sarat Chandra Chattopadhyay is probably the most popular novelist of Bengal. His novels sympathetically depicted the weaker section of the society including woman.

Name of the Classic & Publisher

Sarat Rachanabali (Sarat Samiti)

Name of Articles (Story, Novel, drama, etc.)
Debdas
Pallisamaj
Srikanta

No of Words (per Classic)

Sub. Total : 246032

4. Author's Name

Michael Madhushudan Dutt was a great poet and play writer of Bengal. He has written a few English sonnets and poems. He introduced the "Amitrakshar" rhyme in Bangali poem. His famous work is the epic named 'Meghnadbadh'.

Name of the Classic & Publisher

Madhushudan Rachanabali (Kallol Prakashani)

Name of Articles (Story, Novel, drama, etc.)

Buroshaliker Ghare Row
Krishnakumerir Natak

Padhyabatir Natak

No of Words (per Classic)

Sub. Total : 54089

5. Author's Name

Computer Dictionary

Name of the Classic & Publisher

B. B. Chaudhuri (Ananda Publishers)

No of Words (per Classic)

Sub. Total : 95435

6. Author's Name

Samsad Dictionary

Name of the Classic & Publisher

S. Biswas et. al. (Sahittya Samsad)

No of Words (per Classic)

Sub. Total : 167941

Grand Total : 883729

• Electronic Corpus Of Speech Data

The composition of existing speech databases for English has been studied. Speech data has been categorized into several classes based on criteria such as sex, age, region of speaker, place of data collection, whether the source of the spoken material is a written script etc. Based on these, the composition of the database has been designed. All India Radio, Calcutta has been designated as a potential source of audio material. However, the speech lab, where in the controlled environment the data could be generated, is yet to be set up. due to delayed sanction of the grant.

2.2 Font Generation and Associated Tools

• Public Domain Bangla Font Generation

Overall font generation and editor development process can be divided into the following modules:

1. Designing the exhaustive glyph set.
2. Converter program from Font file to ISCII file and vice-versa
3. Designing a Bangla text editor
4. Designing a floating Keyboard

Background Of Designing The Exhaustive Glyph Set : Every language has its own character set and many of them can be represented within 256 characters using individual 8-bit character sets including Indic scripts using ISCII code. However for unification of all these languages recently the Unicode consortium proposed 16 bit character encoding scheme. Here characters can be designed at 2^{16} spaces. The Government of India is a member of the Unicode Consortium and has been engaged in a dialogue with the UTC about additional characters in the Indic blocks and improvements to the textual descriptions and annotations. Unicode is designed to be a multilingual encoding that requires no escape sequences or switching between scripts. For any given Indic script, the consonant and vowel letter codes of Unicode are based on ISCII, so they correspond directly. For Bangla, Unicode 3.0 provides places 0980 to 09FF. One Bangla True Type Font (called ISI.ttf) has been developed in this center which should be compliance with ISCII and could be upgraded to support Unicode. The font is designed and generated with the help of ALTSYS Fontographer 3.0. The following Bangla orthographic characters can be written with this font :

1. Vowels (11)
2. Consonants (38)
3. Vowel Matras (10)
4. Halant
5. Conjuncts: Bangla contains numerous conjuncts (250+), which essentially are clusters of up to four consonants without the intervening implicit vowels. The shape of these conjuncts can differ from those of the constituting consonants.
6. Punctuation: (12)
7. Numerals (10)

Converter Program From Font File To ISCII File And Vice-versa : The converter program has two submodules. The first one takes a font encoded string as input and delivers an ISCII encoded string as output. The second one takes an ISCII encoded string as input and gives a font encoded string as output.

Bangla Text Editor : Along with the font a Bangla editor has been developed. It supports ISI.ttf font. A web version is generated using Bit stream's web-font wizard to display text in Bangla within a web browser. The editor provides standard editing features such as cut, copy, paste, select, select all, file operations, bold, italic, underlined text, superscript and subscript, left, right and center alignments, find and replace specific strings etc. This editor can save the content in plain text, RTF and in ISCII format. A floating keyboard can be invoked on demand for the new user to get the keying information for writing Bangla in this editor.

Designing a Floating Keyboard : The floating Keyboard is illustrated below.

Figure 2:. Standard Bangla DTP typewriter layout

• **Bangla Spell-Checker**

Bangla spell-checker is a tool to detect errors in Bangla words and correcting them by providing a set of correct alternatives which includes the intended word. An errorneous word can belong to one of two distinct categories, namely, non-word error and real-word error. Let a string of characters separated by spaces or punctuation marks be called a candidate string. A candidate string is a valid word if it carries a meaning. A meaningless string is a non-word. A real word error means a valid word but not the intended one in the sentence. It makes the sentence syntactically or semantically ill-formed or incorrect. In both cases, the problem is to detect the erroneous word and either suggest correct alternatives or automatically replace it by the appropriate word. In this spell-checker, only non-word errors are considered.

Word errors can be classified into four major types namely, substitution, deletion, insertion, and transposition error. In Bangla, wrong uses of characters, which are phonetically similar to the correct ones is observed. A great deal of confusion occur in the use of long and short vowels, aspirated and unaspirated consonants, dental and cerebral nasal consonant due to phonetic similarity. Another type of error is typographic error which is caused by the accidental slip of fingers on the keys which are neighbours of the intended key.

In this spell-checker, the main technique of error detection is based on matching the candidate string in the normal as well as in the *reversed dictionary* (following publications may be referred). To make the system more powerful, this approach is combined with a phonetic similarity key based approach where phonetically similar characters are mapped into a single symbol and a nearly-phonetic dictionary of words is formed. Using this dictionary, phonetic errors can be easily detected and corrected. Here a candidate string first passes through the phonetic dictionary. If the word is not found in the dictionary and also failed to give suggestion then it tries to divide the word in root part and suffix part by separately verifying both. If an error is found, the spell-checker attempts providing suggestions. If it fails, it checks whether the string is a conjunct word generated by appending two noun word and suffix. Option for adding new words permanently or temporarily is provided in the spell checker.

Reference

- B. B. Chauhduri and T. Pal, Detection of word error position and correction using reverse word dictionary', *Intl. Conf. On Computational Linguistics, Speech and Document Processing ICCLSDP'98*, February 18-20, 1998, pp, C41-C46.
- B. B. Chauhduri, A Novel Spell-checker for Bangla Text Based on Reversed-Word Dictionary, *Vivek*, Vol. 14(4), pp. 3-12, October 2002.

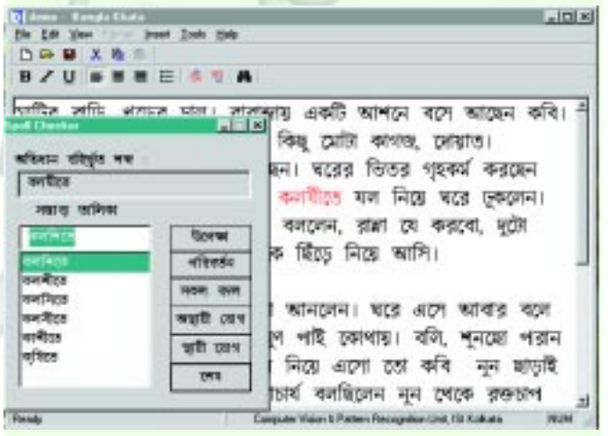


Figure 3: Standard Bangla DTP typewriter layout with a support of a spell checker

For the spell-checker, several files containing root-words and suffix words are maintained. The main dictionary contains about 60,000 root-words and 100,000 inflected words. Noun and verb suffix files are also used. The spell-checker works fast and the non-word error are all correctly detected but it makes about 5% false alarm. This is mainly due to conjunct words formed by euphony and assimilation as well as proper nouns in the corpus.

3. Products

Previously Bangla, Devnagari OCR systems were developed at CVPR Unit of Indian Statistical Institute and these core technologies were transferred to Industry for commercialization. The latest

developments are Oriya OCR and Assamese OCR. These are described below.

3.1 OCR System for Oriya

Purpose

The purpose of this system is to recognize printed Oriya script automatically.

Summary of the System

In this recognition system, the document image is first captured using a flatbed scanner. The image is then passed through different preprocessing modules like skew correction, line segmentation, zone detection, word and character segmentation, etc. Next, individual characters are recognized using a combination of stroke and run-number based features, along with features obtained from the concept of water overflow from a reservoir. These techniques are discussed in a greater detail in the following.

System Description

Text Digitization and Noise Cleaning: Text digitization is done using a flatbed scanner (Model: HP Scanjet 660 C) at a resolution varying from 200 to 300 dots per inch (dpi). The digitized images are in gray tone and a histogram-based thresholding approach is used to convert them into two-tone images. For a clear document, the histogram shows two reasonably prominent peaks corresponding to white and black regions. The threshold value is chosen as the midpoint between the two peaks of the histogram. The two-tone image is converted into 0-1 labels where 1 and 0 represent object and background, respectively. The digitized image shows protrusions and dents in the characters, as well as isolated black pixels over the background, cleaned by a morphological smoothing approach.

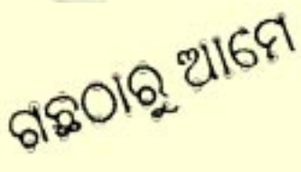


Figure 4: Uppermost and lowermost point of components in a skewed text line

Skew Detection and Correction: When a document is fed to the scanner either mechanically or by a human operator, a few degrees of skew (tilt) is unavoidable. The *skew angle* is the angle that the text lines in the digital image make with the horizontal direction. Skew detection and correction are important preprocessing steps of document layout analysis and OCR approaches. Skew correction can be achieved in two steps, namely (i) estimation of skew angle, and (ii) rotation of the image by the skew angle in the opposite direction. Here a Hough transform based technique is used for estimating the skew angle of Oriya documents. It is observed that the uppermost and lowermost points of most of the characters in an Oriya text line lie on the mean line and base line, respectively. The lowermost and uppermost points of characters in a skewed Oriya text are shown in *Figure 4*. To reduce the amount of data to be processed by the Hough transform, only the uppermost and lowermost pixels of each component are considered. First, the connected components in a given image are identified. For each component, its bounding box (minimum upright rectangle containing the component) is defined. The mean width of the bounding box *bm* is also computed. Next, components having bounding box-width greater than or equal to *bm* are retained. By thresholding at *bm*, small components like dots, punctuation marks, small modified characters, etc., are mostly filtered out. Because of this filtering process, the irrelevant components cannot create errors in skew estimation. Now, the usual Hough transform technique is used on these points to get the skew angle of the document. The image is then rotated according to the detected skew angle. Font style and size variation do not affect the proposed skew estimation method. Also, the approach is not limited to any range of skew angles.

Line, Word and Character Segmentation: For convenience of recognition, the OCR system should automatically detect individual text lines, segment the words from the line, and then segment the characters in each word accurately. Since Oriya text lines can be partitioned into three zones (*Figure 5*), it is convenient to distinguish these zones. Character recognition becomes easier if the zones are

distinguished because the lower zone contains only modifiers and the *halant* marker, while the upper zone contains modifiers and portions of some basic characters.

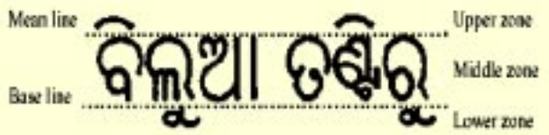


Figure 5: Zones in an Oriya text line

Text Line Detection and Zone Separation: The lines of a text block are segmented by finding the valleys of the projection profile computed by counting the number of black pixels in each row. The trough between two consecutive peaks in this profile denotes the boundary between two text lines. A text line can be found between two consecutive boundary lines. After line segmentation, the zones in each line are detected. From *Figure 6* it can be seen that the upper zone is separated from the middle zone of a text line by the mean line, and that the middle zone is separated from the lower zone by the base line. The uppermost and lowermost points of the connected components in a text line are used to detect the mean line and base line, respectively. A set of horizontal lines passing through the uppermost and lowermost points of the components are considered. The horizontal line that passes through the maximum number of uppermost points (lower most points) is the mean line (base line). It should be noted that the uppermost and lowermost points of the components are previously detected during skew detection, so these points do not have to be recalculated during zone detection.

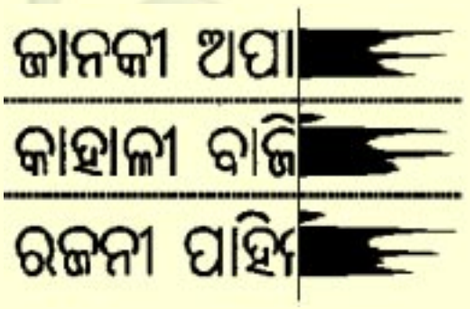


Figure 6 : Projection profile of rows in Oriya text lines (dotted lines show line boundaries)

Word and Character Segmentation: After a text line is segmented, it is scanned vertically, column-by-column. If a column contains two or fewer black pixels, then the scan is denoted by 0, else the scan is denoted by the number of black pixels in that column. In this way, a vertical projection profile is constructed. Now, if in the profile there exists a run of at least k consecutive 0s then the midpoint of that run is considered as the boundary between two words. The value of k is taken as 2/3 of the text line height (text line height is the normal distance between the mean line and the base line). To segment each word into individual characters, only the middle zone of the word is considered. To find the boundary between characters, the image is scanned in the vertical direction starting from the mean line of the word. If during a scan, the base line without encountering any black pixel is reached, then this scan marks the boundary between two characters. However, the gray-tone to two-tone conversion of the image gives rise to some touching characters, which cannot be segmented using this method. To segment these touching characters, the principle of water overflow from a reservoir is used, which is as follows. If water is poured on top of the character, the positions where water will accumulate are considered as reservoirs. *Figure 7* shows the location of reservoirs in a single character as well as in a pair of touching characters. The height of the water level in the reservoir, the direction of water overflow from the reservoir, position of the reservoir with respect to the character bounding box, etc are noted. A reservoir whose height is small and which lies in the upper part of the middle zone of a line is considered as a candidate reservoir for touching character segmentation. The cusp (lowermost point) of the candidate reservoir is considered as the separation point of the touching characters. In *Figure 7*, this position is marked by a vertical line. Because of the round shape of most of the Oriya characters, it is observed that such a reservoir is formed in most of the cases when two characters touch each other. Sometimes, two or more reservoirs may be formed. In such cases, the reservoir close to the middle of the bounding box for segmentation is selected.

Figure 7: Water reservoirs in a single and touching Oriya characters

Feature Selection and Detection: Topological features, stroke-based features as well as features obtained from the concept of water overflow for character recognition are considered. These features are known as principal features. The features are chosen with the following considerations: (a) Robustness, accuracy and simplicity of detection (b) Speed of computation (c) Independence of size and fonts, and (d) Tree classifier design need. Stroke-based and topological features are considered for the initial classification of characters. These features are used to design a tree classifier where the decision at each node of the tree is taken on the basis of the presence/ absence of a particular feature (see *Figure 8*). Stroke-based features include the number and position of vertical lines. The topological features used include existence of holes and their number, position of holes with respect to the character bounding box, ratio of hole height to character height, etc. In addition, the concept of water overflow from a reservoir is also used. The reservoirs in a character are identified, and the position of the reservoirs with respect to the character bounding box, the height of each reservoir, the direction of water overflow, etc. are used as features in the recognition scheme.

Platform

This system is developed in C language in UNIX platform. A WINDOWS based version is also designed.

Portability

The system can run in any UNIX and WINDOWS platform.

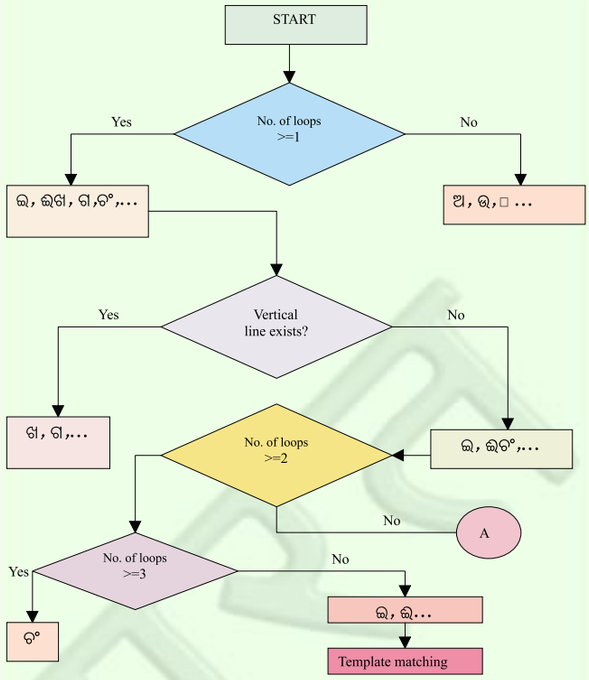


Figure 8: Portion of the tree classifier for Oriya characters

Performance

Line Segmentation: Our system identifies individual text lines with an accuracy of 97.5%.

Word Segmentation: The overall word segmentation accuracy of the system is 97.7%. The error rate for the inferior documents is 4.2%, whereas for the good-quality documents, it is only 1.2% (these figures were calculated based on correctly segmented text lines only).

Character Segmentation : The character segmentation accuracy of the system is 97.2%. The proposed method for separating touching characters based on the water reservoir concept is generally successful.

Character Recognition: On average, the system recognizes characters with an accuracy of about 96.3%, i.e. the overall error rate is 3.7%.

3.2 Adaptation of Bangla OCR to Assamese

Background

We have already developed an efficient OCR system for printed documents in Bangla. Since

Assamese and Bangla share the same script, this OCR system can be successfully used for Assamese documents after some modifications. The modification is needed mainly in the post-processing stage when language specific OCR error correction is needed.

Summary of the System

The segmentation of a document image into lines, words, and characters, and the recognition of segmented characters is dependent on the script only. Thus, the modules of the existing OCR system for Bangla can be used for Assamese OCR. However, certain post-processing steps after basic recognition in order to improve OCR accuracy are required. For example, the words in the output of the OCR system may be looked up in a lexicon; a correctly recognized word will be present in the lexicon, whereas an incorrectly recognized word will usually not be found. The incorrect word can then be replaced by the lexicon-word nearest to it. This post-processing is obviously language-dependent. In this project, we have made the necessary modifications in our OCR system so that it can be used on Assamese documents.

System Description

The major components of the system are same as those of Bangla. Details of these components can be found in the following references. Modifications are done in the following modules :

1. Update of Symbol-list: The symbol-list for Bangla script is updated to contain Assamese "ra" and "wa" and all the conjuncts involving these two characters. Other characters remain unaltered.

2. Formation of the Prototype Library: The prototype library used in Bangla OCR is modified by adding new character shapes found in Assamese script. Character shapes not appearing in Assamese script are deleted from the library.

Figure 9. A sample output from the Assamese OCR System

Performance

Test pages are selected from three Assamese books. Pages are scanned at 300 dpi. In total, 50 pages are used in testing phase. Analysis of test results shows a character-level accuracy of about 95%. Since the font used for printing Assamese materials are somewhat different from the fonts used in Bangla, generation of a new prototype library considering major Assamese fonts would improve the overall accuracy of the system.

Design of post-processing module: Post-processing in Bangla OCR is done using a lexicon of Bangla language. The same module can be used to do post-processing for Assamese also. However, a lexicon of Assamese language is needed for this purpose. This activity has been taken up by IIT, Guwahati.

Technology Transfer

The source code of the system along with the technical details has transferred to IIT, Guwahati.

Technical Reports/Correspondence

An MCA student, Anirban Mukherjee did work on this project for his MCA dissertation submitted to Indira Gandhi National Open University (IGNOU), New Delhi. Title of his work is "Development of an Optical Character Recognition System for Assamese Script".

References

1. B.B. Chaudhuri and U. Pal, "A complete printed Bangla OCR system", *Pattern Recognition*, vol. 31, pp. 531-549, 1998.

2. U. Garain and B. B. Chaudhuri, "Segmentation of Touching Characters in Printed Devnagari and Bangla Scripts using Fuzzy Multifactorial Analysis", *IEEE Transactions on Systems, Man and Cybernetics, Part C*, Vol. 32, No. 4, pp. 449-459, 2002.

3. A. Ray Chaudhuri, A. Mandal, and B. B. Chaudhuri, *Page Layout Analyzer for Mulitingual Indian Documents,* Proc. Language Engineering Conference, IEEE CS Press, 2002.

3.3 Information Retrieval system for Bangla documents

Background

Digital information is available in various forms like text, image and speech data or multimedia content. Among them the text information is considerably abundant and can be easily created. Passage retrieval from text documents is gaining momentum over document retrieval for the last several years. Document ranker returns documents, which is often infeasible to search and extract the necessary information from the entire document. Passage retrieval on the other hand returns fixed or variable sized text chunks from document(s) where the information is likely to reside. This saves both time and effort for searching from a huge text document corpus.

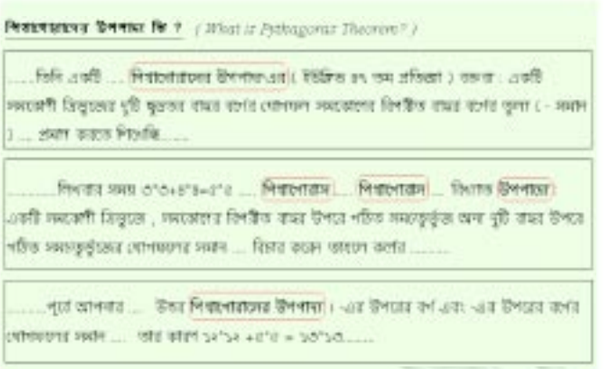


Figure 10. A sample output from the prototype developed. The question "what is Pythagoras theorem" was submitted in Bangla and the system retrieves passages and ranked as output.

Summary of the System

A prototype *n*-gram based language identifier for identifying Indian languages has been developed [1]. A prototype of natural language text indexer for passage retrieval has been developed for Bangla.

System Description

Note that Indian languages can be grouped into five categories based on their origins: Indo-European (Hindi, Bangla, Marathi, etc.), Dravidian (Tamil, Telugu, etc.),Tibeto-Burmese (e.g., Khasi), Astro-Asiatic (Santhali, Mundari, etc.) and Sino-Tibetan (e.g., Bhutanese). Languages within a group share a number of common elements. For instance, there is a significant overlap in the vocabulary of Bangla and other Indo-European languages and are mutually closer than the profiles for a pair of languages from two different groups. We have tested the character level *n*-gram algorithms for language identification from a multilingual collection of Indian language documents. Also, a prototype of a "English to Bangla" phonetic transliteration scheme is designed and implemented for cross-lingual information.

Part of the *n*-gram distance matrix between every pair of Indian languages is shown in *Table* 3.

Table 3. The n-gram distance between some major Indian language

Profile	Bangla	Hindi	Kannada	Kashmiri	Malayalam	Telugu	Urdu
Bangla	0	16.54	19.42	23.66	20.27	19.08	24.01
Hindi	16.54	0	18.40	23.65	19.74	18.58	24.12
Kannada	19.42	18.40	0	23.80	18.11	16.65	24.09
Kashmiri	23.66	23.65	23.80	0	24.02	23.88	19.54
Malayalam	20.27	19.74	18.11	24.02	0	18.07	24.29
Telugu	19.08	18.58	16.65	23.88	18.07	0	24.15
Urdu	24.01	24.12	24.09	19.54	24.29	24.15	0

A passage detection and ranking algorithm for Bangla text has been designed and implemented. *Stop-word-list* are common words ignored by search engines at the time of searching and these words generally do not contain any information. For constructing Bangla search engines by combining statistical and manual methods, about 500 stop words are identified. The DoE Bangla corpus is used for this purpose.

The indexer generates an indexed file, which keeps the record in this fashion:

Document No., Term, Term_weight (f_{tD}/N_D), Frequency of the term in whole corpus f_t, Occurence position of the term

Document No: The identity number of the document in the whole corpus.

Term weight: The ratio of the term frequency for a particular term in a document (f_{tD}) and the document length in bytes (N_D).

The occurrence position of the term: The numerical value of the position of the term, counted from the beginning of the document. The position of first term of the document is one. So for each term f_{tD} occurrence positions are listed.

A prototype of passage detection algorithm is developed and a few standard passage-ranking algorithms have also been tested.

Technical Reports/Correspondence

1. "N-gram: a language independent approach to IR and NLP", P. Majumder, M. Mitra, B.B. Chaudhuri, Proc. *International Conference on UniversalKnowledge and Language (ICUKL-2002), Goa, India, 2002.*

3.4 Script Identification and Separation From Indian Multi-Script Documents

Background

India is a multi-lingual multi-script country, where a single document page (e.g., passport application form, examination question paper, money order form, bank account opening application form) may contain lines in two or more language scripts. For this type of document page there is a need for separating different scripts before feeding them to respective OCR systems. The purpose of this system is to identify different script regions of the document and hence separate them.

Summary of the System

The system works in two stages. In the first stage, it separates each line in the scanned document page. Line segmentation is based on horizontal profile based technique. Secondly, based on the distinguishing features between different Indian scripts, it identifies the script for each line in which it is actually written. The identification of a particular script from other scripts is mainly based on water reservoir principle based features, features based on contour tracing, profile features etc. These features are elaborated below. At present, the system has an overall accuracy of about 97.52%.

System Description

Some of the major distinguishing features used to separate different Indian scripts in a document page from each other, are discussed.

Horizontal projection profile: From the following *Figure 11*, it is apparent that there is a distinct difference among some of the scripts in terms of horizontal projection profile.

Figure 11: Different Indian script lines (from top to bottom: Devnagari, Bangla, Gurmukhi,

Malayalam, Kannada, English, Tamil, Telugu, Urdu, Kashmiri, Gujarati, Oriya) with their row-wise maximum run (left side) and horizontal profile (right side).

Water reservoir principle based feature: Top (bottom) reservoir is defined as the reservoir obtained when water is poured from top (bottom) of the component. (A bottom reservoir of a component is visualized as top reservoir when water will be poured from top after rotating the component by 180°). Similarly, if water is poured from left (right) side of the component, the cavity regions of the components where water will be stored are considered as left right) reservoirs. This is shown in *Figure 12*. Here top, bottom, left and right reservoirs are shown for the English character X. The water flow levels of these reservoirs are also shown in this figure.

Figure 12: Top, bottom, left and right reservoirs are shown for the character X. Water flow level of reservoir is shown by dotted arrow.

In some scripts, many reservoirs may be obtained from a particular side of the characters whereas in some other scripts many reservoirs may not be obtained from that side. Thus, this feature is useful in distinguishing between scripts.

Left and right profile: In this feature each character is located within a rectangular boundary, a frame. The horizontal or vertical distances from any one side of the frame to the character edge are a group of parallel lines, known as the *profile* (*Figure 12*). If left, right and top profile of the characters in a text lines are computed, it is observed that there are some distinct difference in some of the scripts according to these profiles. Left and right profile of a Malayalam character is shown in *Figure 13*.

Figure 13: Left and right profile of a character is shown.

Head-line feature: If the longest horizontal run of black pixels on the rows of a text line is taken then such run length (known as Head-line) can be used to distinguish between scripts with this feature (like Bangla) and those without Head-line (like English).

Feature based on jump discontinuity: Here jump discontinuity (a relatively small white run between two black runs) of a component from a particular side is considered. Some script's characters (like in Telugu, Kannada etc.,) have prominent jump discontinuties and the occurrence frequency of this particular feature is successfully used for script identification. The jump discontinuity for a Gujrati character is shown in *Figure 14*.

Figure 14: Example of jump discontinuity feature.

Based on these above major features the system identifies the script of a particular line.

Script Identification Technique:

- Scanned gray tone document is converted to two tone image using histogram based automatic thresholding approach.
- Noise Removal and Skew correction are performed on this two tone image.
- Line segmentation is performed to the refined image.
- For each line of the image the particular script in which the line is actually written is identified based on a Binary Tree classifier.
- Once the script of the line is identified its region is marked by the name of the particular script.
- The marked lines are shown in the output image.

A part of the Binary Tree classifier is shown in *Figure 15*.

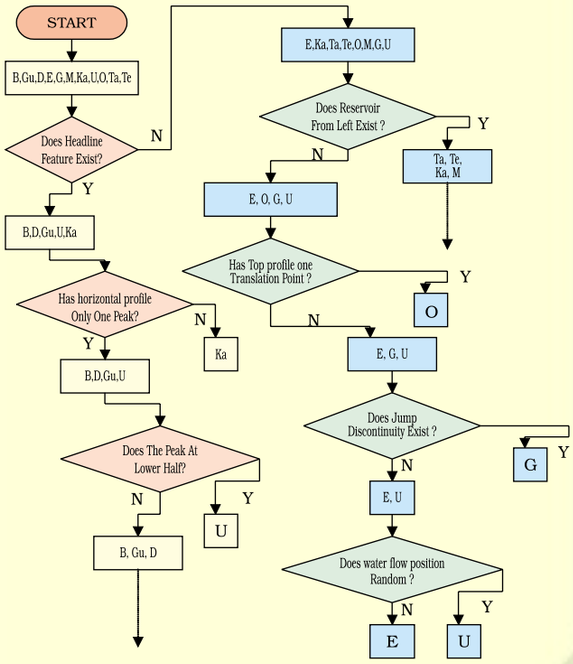


Figure 15: Flow diagram of the script identification scheme.
(Here B =Bangla, D=Devnagari, E=English, Gu=Gurumukhi, M=Malayalam, Ta=Tamil, Te=Telugu, G=Gujarati, Ka=Kannada, U=Urdu and O=Oriya).

Platform

This system is developed in C language in UNIX platform. An user interface of this system is built in VC++ 6.0 in WINDOWS2000 platform.

Portability

Both the C version and the VC++6.0 version of this system can run in any UNIX and WINDOWS platform respectively.

Performance

The system is tested based on data taken from various sources like Journals, Newspaper, Synthetic Document etc. Currently the system has an overall accuracy of 97.52%. The output generated by the system when run on a document page is shown in *Figure 16*.

Figure 16: The output generated by the system

Papers Published

- U. Pal and B. B. Chaudhuri, "Identification of different script lines from multi-script documents", Image and Vision computing, vol. 20, no.13-14 pp. 945-954 2002.
- U. Pal and B. B. Chaudhuri, "Script line seperation from Indian multi-script documents" IETE Journal of Research, vol. 49. no.1. 2003.
- U. Pal, S. Sinha, B. B. Chaudhuri, "Multiscript Line Identification From Indian Documents", 7th International Conference on Document Analysis & Recognition, ICDAR 2003 (In press).

4. Research & Development

4.1 Automatic Processing of Hand-printed Table-Form Documents

Background

In office environment, thousands of documents containing tables may be handled while processing application forms. In Indian context, most of the cases, these table-form documents contain hand-printed text (like customer's name, date, item details/quantity/price etc.) mixed with printed text (invoice no., challan no., etc.). Once these forms are collected from the stockists, customers, or other business centers, the computer operators

manually enter the data into the computer to maintain an electronic version of the same. This manual approach makes this processing time consuming, tedious and inefficient. Hence, an automatic approach is called for. Moreover, the work may be the basis of handwriting recognition of Indian languages.

Summary of the Project

This project deals with the automatic processing of hand printed table-form documents. It extracts different blocks of handwritten information from a filled-in form. Each such block is segmented into lines, words and characters. Identification of each block is followed by tagging (numbering) them according to the order of their physical placements. In the final stage of this information extraction procedure, images of individual hand printed characters are obtained which should be passed as inputs to a hand printed character recognition system.

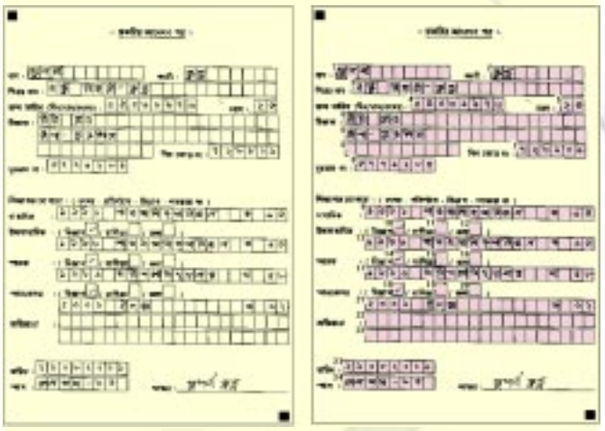


Figure 17: The binarized image(left) and the result after the form segmentation (right).

Description of the Work Done

In our approach, extraction of handwritten information from a filled-in form, does not take care of the grey values of the pixels but it works on the binarized image of the input form. In the first step, horizontal profiles of object pixels help to segment different horizontal lines of information. In each such line, one or more blocks are identified by exploiting the vertical profiles within each horizontal strip. Within a block, words are identified by considering the gap between two consecutive words. Characters in a word are segmented again by considering the vertical profiles restricted to the image of the word. This information in a block-word-character hierarchy is stored in a three dimensional list.

Testing of the System

We collected several hand-filled-in application form seeking a job. In Figure 17, result of our form document processing system is shown.

Technical Reports/Correspondence

An MCA student, Mr. Saikat Das did the project work for his MCA dissertation submitted to Indira Gandhi National Open University (IGNOU), New Delhi. Title of his work is "Recognition of Hand-written (touching) Bangla characters from Form Type Documents".

Another student Prasenjit Mitra to complete DOE "B" level has started his project work to develop a system which can automatically process Indian Money Order Forms. He will take approximately another 4 months to complete the assignment.

4.2 Research and Development of Neural Network based tools for printed document (in eastern regional scripts) processing

Background

Artificial neural network (ANN) based methods in solving various pattern recognition problems have several advantages over the conventional/ classical approaches. Recently ANN based classification approaches have gained tremendous popularity. They are commonly used in the high accuracy systems because they perform satisfactorily in the presence of incomplete or noisy data and also they can learn from examples. Another advantage is the parallel nature of ANN algorithms. On the other hand, we have already

developed an efficient OCR system for printed documents in Bangla. There are obvious justifications to explore suitable supervised/ unsupervised neural network/hybrid models for developing software tools for shape extraction of individual printed characters with a view to character classification and this may in turn make the already developed OCR system more efficient.

Summary of the Project

For extraction of shapes of individual printed characters we may consider Self-Organizing neural network or Vector Quantization techniques. Instead of using the input character image for classification purpose we may obtain its graph representation consisting of a few nodes and links between them. Useful topological and geometrical features may be easily obtained from such a representation and those in turn should result in better classification accuracy.

Description of the Work Done

We used Topology Adaptive Self-Organizing Neural Network (TASONN) model to obtain the graph representation of the input character. We considered a few structural features of this graph describing the topology of the character along with a hierarchical tree classifier to classify printed Bangla characters into a few subclasses. To recognize different characters in each of these resulting subclasses we considered several geometrical features. Final recognition is performed using a look-up table of these feature values.

Testing of the System

Test pages are selected from different Bangla books. Pages are scanned at 300 dpi. In total, 100 pages are used for training purpose and another 50 pages are used in the testing phase. Analysis of test results shows a character-level accuracy of about 98%. Error analysis indicates that since the font used for printing of different books are somewhat different from each other, generation of a larger training set would improve the overall accuracy of the system.

5. The Team Members

S. K. Parui	swapan@isical.ac.in
A. K. Datta, M. Mitra	mandar@isical.ac.in
U. Pal	umapada@isical.ac.in
S. Palit	sarbani@isical.ac.in
U. Bhattacharyya	ujjwal@isical.ac.in
U. Garain	utpal@isical.ac.in
T. Pal	tama@isical.ac.in
N. S. Dash	niladri@isical.ac.in
A. Datta and D. Sengupta	

Courtesy: Prof. B.B Chaudhary
Indian Statistical Institute
Computer Vision and Pattern Recognition Unit
203, Barrackpore Trunk Road
Kolkata –700035
(RCILTS for Bengali)
Tel: 00-91-33-25778086
Extn. 2852, 25781832, 25311928
E-mail: bbc@isical.ac.in