OCR in Indian Languages

Abhishek Verma*, Suket Arora** and Preeti Verma***
*B.Tech (CSE), ACET, Amritsar, abhi.asr11@gmail.com
**Assistant Professor, ACET, Amritsar, suket.arora@yahoo.com
***Assistant Professor, ACET, Amritsar, preet_asr156@yahoo.in

Abstract: Optical Character Recognition or OCR is the electronic translation of handwritten, typewritten or printed text into machine translated images. Optical Character Recognition (OCR) is a very important task in Pattern Recognition. Foreign languages, especially English character recognition has been extensively studied by many researches but due to complication of Indian Languages like Hindi ,Punjabi ,teulgu ,malyalam etc. the research work is very limited and constrained. This paper presents the research work related to all Indian languages, various approaches to character recognition along with some applications of character recognition is also discussed in this paper. The aim of this paper is to provide an overview of the research going on in Indian script OCR systems. This survey paper has been felt necessary when the research on OCRs for Indian scripts is still a challenging task. Hence, a brief introduction to the general OCR and typical steps in the development of an OCR are given in this paper, along with a brief description of the different techniques used in them. This paper is prepared to be as self sufficient, and complete as possible, so that it provides a starting point for the researchers entering into this area.

Keywords: preprocessing; noise removal; thinning; binarization; skewing; segmentation; peculiarities of indian scripts.

Introduction

Optical Character Recognition is an active field of research in Pattern Recognition. The problem of character recognition can be classified based on two criteria. One is based on the type of the text which is printed or hand written. The other is based on the acquisition process which can be on-line of off-line .It is generally considered that the on-line method of recognizing handwritten text has achieved better results than its off-line counterpart. This may be attributed to the fact that more information may be captured in the online case such as the direction, speed and the order of strokes of the handwriting. On the other hand machine-printed character recognition can achieve very good results on good quality documents. In case of online character recognition, there is real time recognition of characters. Online systems have better information for doing recognition since they have timing information and can avoid the initial search step of locating the character as in the case of their offline counterpart. Online systems obtain the position of the pen or printed character as a function of time directly from the interface. Offline recognition of characters is known as a challenging problem because of the complex character shapes and great variation of character symbols written or printed in different modes. In case of offline character recognition, the typewritten or handwritten character is typically scanned in the form of a paper document and made available in the form of a binary or gray scale image to the recognition algorithm. Offline character recognition is a more challenging and difficult task as there is no control over the medium and instrument used.

Brief History of Character Recognition

Many methods have been proposed for character recognition. But they are often subjected to substantial constraints due to unexpected difficulties. Historically character recognition system has evolved in three ages, namely the periods cited denoting as:

1900-1980 (early ages)

The history of character recognition can be traced as early as 1900. When the Russian Scientist Tyering attempted to develop an aid for visually handicapped. The first character recognizers appeared in the middle of 1940s with the development of digital computers. The early work on the automatic recognition of characters has been concentrated either upon machine printed text or upon small set of well distinguished hand written text or symbols. The commercial character recognizers were available in 1950s.

1980-1990 (Developments)

The studies until 1980 suffered from the tack of powerful computer hardware and data acquisition derives. However, the character recognition research was focused on basically the shape recognition techniques without using any semantic information.

After 1990 (advancements)
The real progress on character recognition system is achieved during this period, using the new development tools and methodologies, which are empowered by continuously growing information technologies. In the early nineties, Image processing and Pattern recognition techniques are efficiently combined with the Artificial Intelligence methodologies. Nowadays in addition to the more powerful computers and more accurate electronic equipments such as scanners, cameras and electronic tablets, we have efficient, modern use of methodologies such as neural networks, Hidden Markov models; Fuzzy set reasoning and Natural language processing.

Preprocessing in OCR

Any ocr implementation consists of a number of preprocessing steps followed by the actual recognition. The number and types of preprocessing algorithms employed on the scanned image depend on many factors such as age of the document, quality of paper, resolution of the scanned document or image, skewing in the image, the format and layout of the images and text, the kind of script used and also on the type of characters - printed or handwritten. Typical preprocessing includes the following stages:

- Binarization
- Noise removing
- Thinning
- Skew detection and correction
- Line, word and character segmentation
- Feature extraction and feature selection
- Classification.

Binarization
Binarization is a technique by which the gray scale images are converted to binary images. Binarization separates the foreground and background information. Binarization is described below in Fig. 1.

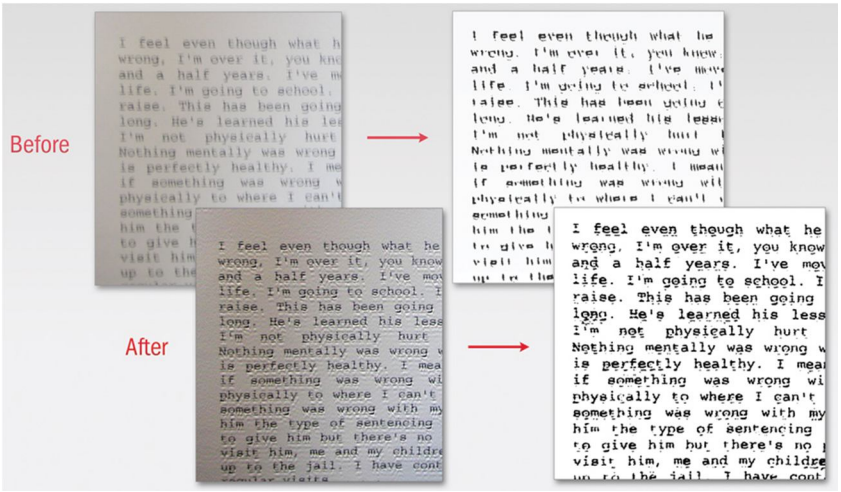

Figure 1. Binarization

Noise removing
Scanned documents often contain noise that arises due to printer, scanner, print quality, age of the document, etc. Therefore, it is necessary to filter this noise before we process the image. For more clarity about noise removing, refer Fig. 2 given below.

Thinning
Thinning is also known as skeletonization. It is a process by which a one-pixel-width representation of an object is obtained, by preserving the connectedness of the object and its end points. The purpose of thinning is to reduce the image components to their necessary information so that further analysis and recognition are facilitated. Thinning of characters is shown in Fig. 3.

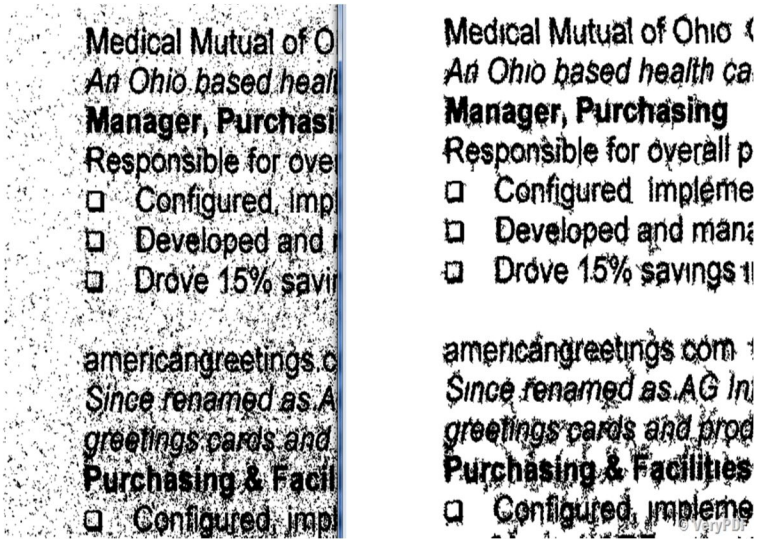

Figure 2. Noise removal

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) are unavoidable. Skew angle is the angle that the lines of text in the digital image make with the horizontal direction. Therefore, it is necessary to reduce or remove the tilt of a scanned image. The skewing of a scanned image can be done as in Fig. 4.

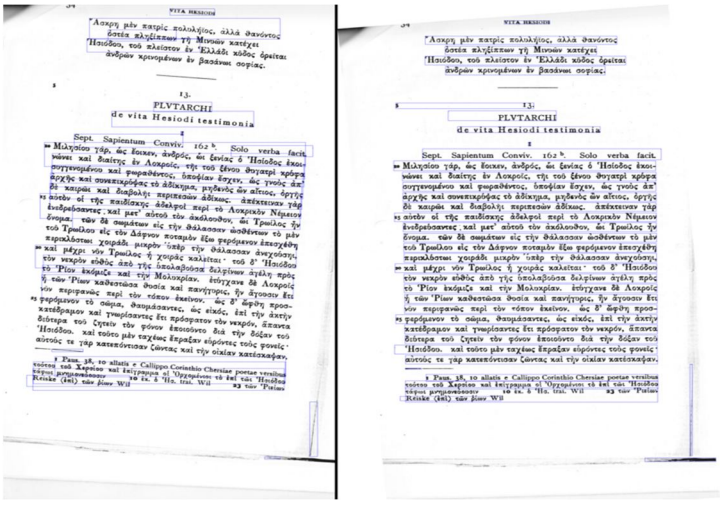

Figure 4. Skewing

Line, Word and Character segmentation

After the tilt is corrected, the text has to be segmented first into lines; each line then into words and finally each word has to be segmented into its constituted characters. The line, word and character segmentation is described in Fig. 5, Fig. 6 and Fig. 7 respectively.

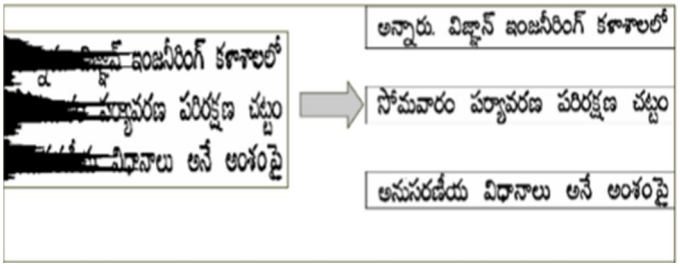

Figure 5. Line segmentation

Figure 6. Word segmentation

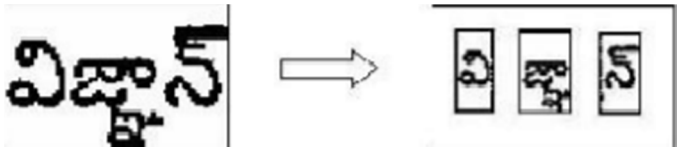

Figure 7. Character recognition

Feature extraction and feature selection

Feature extraction can be considered as finding a set of parameters (features) that define the shape of the underlying character as precisely and uniquely as possible. The features have to be selected in such a way that they help in discriminating between characters.

Classification

The classification stage in an OCR process assigns labels to character images based on the features extracted and the relationships among the features. In simple terms, it is this part of the OCR which finally recognizes individual characters and outputs them in machine editable form.

OCR in different Indian Languages

While a large amount of literature is available for the recognition of English scripts, relatively less work has been reported for the recognition of Indian languages. Not much attempts have been made on the character recognition of Indian character sets. However, some major works are reported on Devanagari. Some attempts are also reported on Tamil, Kannada, Guajarati, Bengali, Malayalam and Telugu. Main reasons for this slow development could be attributed to the complexity of the shape of Indian scripts, and also the large set of different patterns that exist in these languages, as opposed to English.

Peculiarities of Indian script

Indian scripts are different from Roman script in several ways. Indian scripts are two-dimensional compositions of symbols: core characters in the middle strip, optional modifiers above and/or below core characters. Two characters may be in shadow of each other. While line segments (strokes) are the predominant features for English, most of the Indian language scripts are formed by curves, holes, and also strokes. In Indian language scripts, the concept of upper-case, and lower-case characters is absent; however, the alphabet itself contains more number of symbols than that of English.

Techniques used in different Indian script OCRs

Any OCR contains more or less the same steps described above in preprocessing. The exact number and techniques differ slightly from one language to other. We now present the studies in different OCRs, along with the description of the methods used in them.

Bangla OCR

Recognition of isolated and continuous printed multi font Bengali characters is reported in the work by Mahmud et al (2003). This is based on Freemanchaincode features, which are explained as follows. When objects are described by their skeletons or contours, they can be represented by chain coding, where the ON pixels are represented as sequences of connected neighbors along lines and curves. Instead of storing the absolute location of each ON pixel, the direction from its previously coded neighbor is stored. The chain codes from center pixel are 0 for east, 1 for North- East, and so on. Chain code gives the boundary of the character image and slope distribution of chain code implies the curvature properties of the character. In this work, connected components from each character are divided into four regions with the center of mass of as the origin. Slope distribution of chain code, in these four regions is used as local feature. Ray & Chatterjee (1984) presented a recognition system based on a nearest neighbor classifier employing features extracted by using a string connectivity criterion.

A complete OCR for printed Bangla is reported in the work by Chaudhuri & Pal (1998), in which a combination of template and feature-matching approach is used. For a clear document the histogram shows two prominent peaks corresponding to white and black regions. The threshold value is chosen as the midpoint of the two-histogram peaks. Skew angle is determined from the skew of the headline. Text lines are partitioned into three zones and the horizontal and vertical projection profiles are used to segment the text into lines, words, and characters. Primary grouping of characters into the basic, modified and compound characters is made before the actual classification. A few stroke features are used for this purpose along with a tree classifier where the decision at each node of the tree is taken on the basis of presence/absence of a particular feature. The compound character recognition is done in - two stages: In the first stage the characters are grouped into small sub-sets by the above tree classifier. At the second stage, characters in each group are recognized by a run-based template matching approach. Some character level statistics like individual character occurrence frequency, bigram and trigram statistics etc. are utilized to aid the recognition process. For single font, clear documents 99.10% character level recognition accuracy is reported.

Gurmukhi (Punjabi) OCR

G S Lehal and Chandan Singh presented an OCR system for printed Gurumukhi script. The skew angle is determined by calculating horizontal and vertical projections at different angles at fixed interval in the range [0° to 90°]. The angle, at which the difference of the sum of heights of peaks and valleys is maximum, is identified as the skew angle. For line and word segmentation horizontal and vertical projection profiles are respectively used. Each word is segmented into connected components or sub-symbols, where each sub-symbol corresponds to the connected portion of the character lying in one of the three zones. Connected components are formed by grouping together black pixels having 8-connectivity. Primary feature set is made up of features which are expected to be font and size invariant such as number of junctions with the headline equals 1, presence of sidebar, presence of a loop, and loop along the headline. The secondary feature set is a combination of local and global features: number of endpoints and their location, number of junctions and their location, horizontal projection count, right profile depth, left profile depth, right and left profile directions, and aspect ratio. Binary tree classifier is used for primary features, and the nearest neighbor classifier with a variant sized vector was used for the secondary features. This multi-stage classifier is used to classify the sub -symbols and they are then combined using heuristics and finally converted to characters. A recognition rate of 96.6% at a processing speed of 175 characters/second was reported.

Lehal & Singh (2002) also developed a post processor for Gurmukhi. In this, Statistical information of Punjabi language such as word length, shape of the words, and frequency of occurrence of different characters at specific positions in a word, information about visually similarlooking words, grammar rules of Punjabi language, and heuristics are utilized. RCILTS for Punjabi is Thapar Institute of Engineering & Technology, Patiala.

Devanagari OCR

OCR work on printed Devnagari script started in early 1970s. Sinha & Mahabala (1979) presented a syntactic pattern analysis system with an embedded picture language for the recognition of handwritten and machine printed Devnagari characters. For each symbol of the Devnagari script, the system stores structural description in terms of primitives and their relationships. Problems that arise in developing OCR systems for noisy images are addressed in the work by Parvati Iyer et al (2005). Lines are segmented into word-like units, based on the dips in the vertical projection profile of the line. Some statistical data such as minimum and average widths, height, etc, are computed. Basic geometrical shapes such as full vertical bar, a horizontal line, diagonal lines in both the orientations, circles and semicircles of varying radii, and orientations are used to form the feature vector. Characters are classified using a rule-based approach. Hamming distance metric is employed. Character recognition rate of only 55% is reported. The authors also trained a feed- forward back propagation neural network, with a single hidden layer. Character recognition rate of 76% is reported with this neural network approach.

Veena (1999) described Devnagari OCR in her doctoral thesis. Here, segmentation is done using a two-stage, hybrid approach. The initial segmentation extracts the header line, and delineates the upper strip from the rest. This yields vertically separated character boxes that could be conjuncts, touching characters, shadow characters, lower modifiers or a combination of these. Segmentation is done based on structural information obtained from boundary traversal in the second stage.An error

detection and correction phase is also included as post processing. Performance of 93% accuracy at character level is reported. Pal & Chaudhuri (1997) reported a complete OCR system for printed Devnagari. In this, headline deletion is used to segment the characters from the word. Also, a text line is divided into three horizontal zones for easier recognition procedure. After preprocessing, and segmentation using zonal information and shape characteristics; the basic, modified and compound characters are separated. A structural feature-based tree classifier recognizes modified and basic characters, while compound characters are recognized by hybrid approach combined with structural and run based template features. The method reports about 96% accuracy.

Telugu OCR

The first reported work on OCR of Telugu Character is by Rajasekaran & Deekshatulu(1977). It identifies 50 primitive features and proposes a two-stage syntax-aided character recognition system. In the first stage a knowledge-based search is used to recognize and remove the primitive shapes. In the second stage, the pattern obtained after the removal of primitives is coded by tracing along points on it. Classification is done by a decision tree. Primitives are joined and superimposed appropriately to define individual characters.

The concept of Telugu characters as composed of circular segments of different radii is made use of in the work by Rao & Ajitha (1995). Recognition consists in segmenting the characters into the constituent components and identifying them. Feature set is chosen as the circular segments, which preserve the canonical shapes of Telugu characters. The recognition scores are reported as ranging from 78 to 90% across different subjects, and from 91 to 95% when the reference and test sets were from the same subject. Sukhaswami et al (1995) proposed a neural network based system. Hopfield model of neural network working as an associative memory is chosen for recognition purposes initially. Due to the limitation in the storage capacity of the Hopfield neural network, they later proposed a multiple neural network associative memory (MNNAM). These networks work on mutually disjoint sets of training patterns. They demonstrated that storage shortage could be overcome by this scheme.

Pujari et al (2002) proposed a recognizer that relies on wavelet multi-resolution analysis for capturing the distinctive characteristics of Telugu script . Gray level input text images are line segmented using horizontal projections; and vertical projections are used for the word segmentation. Character images of size 8x8 are converted to binary images using the mean value of the grey level as the threshold. The resulting bit string of 64 bits is used as the signature of the input symbol. A Hopfield-based Dynamic Neural Network is designed for the recognition purpose. The performance across fonts and sizes is reported as varying from 93% to 95%. The authors reported that the same system, when applied to recognize English characters, resulted in very low recognition rate since the directional features that are prevalent in Latin scripts are not preserved during signature computation with wavelet transformation.

An OCR for Telugu is reported by Negi, et al (2001). Instead of segmenting the words into characters as usually done, words are split into connected components. Run Length Smearing Algorithm (RLSA) and Recursive XY Cuts are used to segment the input document image into words. About 370 connected components are identified as sufficient to compose all the characters including punctuation marks and numerals. Template matching based on the fringe distance is used to measure the similarity or distance between the input and each template. The template with the minimum fringe distance is marked as the recognized character. The template code of the recognized character is converted into ISCII, the Indian Standard Code for Information Interchange. Raw OCR accuracy with no post processing is reported as 92%. Performance across fonts varied from 97.3% for Hemalatha font to 70.1% for the newspaper font.

Non-linear normalization to improve performance was used by Negi et al, (2002) by selectively scaling regions of low curvature. This is based on a dot density feature normalization method. The authors observed distortions in the shapes, but reported improvement in the OCR recognition accuracy. Performance across different fonts is not investigated. Negi and Nikhil (2003) attempted Layout analysis to locate, and extract Telugu text regions from document images. The gradient magnitude of the image is computed to obtain contrasting regions in the image. After binarization, and noise removing, Hough Transform for circles is applied on the gradient magnitude of the image to obtain the circular gradient which is a prominent feature of Telugu text. Each detected circle is filled to obtain the regions of interest. Recursive XY cuts and projection profiles are used to segment the document image into paragraphs, lines, and words. Factors that can improve the OCR performance are discussed by Bhagvati et al (2003).

Lakshmi & Patvardhan (2003) presented recognition of basic Telugu symbols .After obtaining the minimum bounding rectangle, each character (basic symbol) is resized to 36 columns, while maintaining the original aspect ratio. A preliminary classification is done by grouping all the symbols with approximately same height (rows). Feature vector is computed out of a set of seven invariant moments from the second and third order moments. Recognition is done using k-nearest neighbor algorithm on these feature vectors. A single font type is used for both training and test data. Testing is done on noisy character images with Gaussian noise, salt andpepper noise and speckle noise added. Preprocessing such as line, word, and character segmentation is not addressed in this work. The authors extended the work to multi font OCR (Lakshmi & Patvardhan 2002). Preprocessing stages such as binarization, noise removal, skew correction using Hough transform method, Lines and words segmentation using horizontal and vertical projections are included in this work. Basic symbols from each

word are obtained using connected components approach. After preliminary classification as in the previous work, pixel gradient directions are chosen as the features. Recognition is done again using the k-nearest neighbor algorithm on these feature vectors. The training vectors are created with three different fonts and three different sizes: 25, 30 and 35. Testing is done on characters with different sizes, and also with some different fonts. Recognition accuracy of more than 92% for most of the images is claimed.

In a more recent work by the same authors (Lakshmi & Patvardhan 2003) , neural network classifiers and some additional logic are introduced. The feature vectors obtained from pixel gradient directions are used to train separate neural networks for each of the sets identified by the preliminary classification scheme. Testing is done on the same 3 fonts used for training, but, with different sizes. A high recognition accuracy of 99% in most cases for laser and desk jet quality prints is reported.

DRISHTI is a complete Optical Character Recognition system for Telugu language developed by the Resource Center for Indian Language Technology Solutions (RCILTS), at the University of Hyderabad (JLT, July 2003 pg110-113).The techniques used in Drishti are as follows: For binarization three options are provided: global, percentile based and iterative method. Skew Detection and Correction are done by maximizing the variance in horizontal projection profile. Text and Graphics Separation is done by horizontal projection profile. Multi-column Text Detection is done using Recursive X-Y Cuts technique. It is based on recursively splitting a document into rectangular regions using vertical and horizontal projection profiles alternately. Word segmentation is done using a combination of Run-Length Smearing Algorithm (RLSA) and connected-component labeling. Words are decomposed into glyphs by running the connected component labeling algorithm again. Recognition is based on template matching using fringe distance maps.

Anuradha Srinivas, et al (2007) developed a Telugu optical character recognition system for a single font. Sauvolas algorithm is used for binarization; skew detection and correction are done by maximizing the variance in horizontal projection profile. For decomposing the text document into lines, words and characters, horizontal and vertical projection profiles are used. Zerocrossing features are computed, and Telugu characters are grouped into 11 groups based on these crossing features. A 2-stage classifier with first stage identifies the group number of the test character, and a minimum-distance classifier at the second stage identifies the character. Recognition accuracy of 93.2% is reported.

Gujarati OCR

Antani and Agnihotri (1999) described recognition of Gujarati characters. Subsets of similarlooking Gujarati characters were classified by different classifiers: Euclidean Minimum Distance classifier, Nearest Neighbor classifier were used with regular and invariant moments, and the Hamming Distance classifier was also used in the binary feature space. However, a low recognition rate of 67% is reported.

A working prototype of Gujarati OCR is developed by Maharaja Sayajirao University, Baroda. It employs the template matching technique for recognition and nearest neighbor for classification. The input image is assumed to be skew-corrected, with one column of text only. Output is in Unicode format as plain text file. Recognition accuracy of initial results is reported to be good, but not specified.

OCR for Oriya

The features of Oriya OCR developed at the Indian Statistical Institute, kolkata are similar to the Bangla OCR developed by the same team Chaudhuri, et al (2002) and are as follows:

Scanning resolution is at 200 to 300 dots per inch (dpi). Histogram-based thresholding approach is used to convert the images into two-tone images. The threshold value is chosen as the midpoint between the two peaks of the histogram. Hough transform based technique is used for estimating the skew angle using only the uppermost and lowermost pixels of each component. The lines of a text block are segmented by finding the valleys of the horizontal projection profile. Oriya text lines are partitioned into three zones: lower zone contains only modifiers and the halant marker, while the upper zone contains modifiers and portions of some basic characters. After line segmentation, the zones in each line are detected. Vertical projection profile is used for word segmentation. To segment each word into individual 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, and then this scan marks the boundary between two characters. To segment the touching characters, principle of water overflow from a reservoir (Garain & Chaudhuri 2002) is used. After preprocessing, 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. Topological features, stroke-based features as well as features obtained from the concept of water overflow are considered for character recognition. Stroke-based features like the number and position of vertical lines are used for the initial classification of characters using a tree classifier. The topological features used include existence of holes and their number, position of holes with respect to the character bounding box, and ratio of hole- height to character height, etc. Water reservoir features include position of the reservoirs with respect to the character bounding box, the height of each reservoir, the direction of water overflow, etc. On average, the system reports an accuracy of about 96.3%.

Mohanty & Behera (2004) described a complete OCR development system for Oriya script. For skew detection and correction, angular projection profiles are prepared for –8 0 to +8 0 with an interval of 0.050 and a strip-wise histogram is

constructed using these projection profiles. A global maximum at a particular angular direction gives the global skew angle; the whole document is then rotated to remove the skew. Structural features such as upper-part circular, a vertical line on the right most part, holes, horizontal run code, vertical run code, number and position of holes, are extracted from the 16x16 pixel matrix. A tree-based classifier is used in which each node denotes a particular feature. All leaf nodes contain individual characters, modifiers (matras), digits and composite characters. The recognition phase has two parts: In the first phase individual characters, and left and right modifiers are recognized based on structural features; whereas in the second stage upper and lower modifiers are recognized based on run length code. Recognition accuracy is not mentioned.

The Resource Center for Indian Language Technology Solutions (RCILTS) for Oriya is established at Utkal University, Bhubaneswar (JLT October 2003).The features of the OCR, DIVYADRUSTI developed at Utkal University are: Binarization is done using dynamic thresholding technique. Repeated angular projection profiles are used for skew detection. Lines are extracted through strip-wise vertical histogram analysis. Character segmentation is achieved using region growing and labeling, and matra extraction is done using region analysis. Connected components are handled by forward and backward chaining of appropriate mask.

Assamese OCR

Assamese OCR based on the Bangla OCR developed at Indian Statistical Institute, Kolkata (JLT October 2003) has many features. Histogram based global threshold approach is used for binarization. Skew detection and correction is done using the method proposed by Chaudhary & Pal (1998) by finding the headline of the Assamese script in the document image. Words are segmented using the connected component analysis. Segmenting individual characters from a word is by deleting the headline from the word. Simple stroke features like vertical and horizontal lines, horizontal and vertical black runs are used. Two types of classifiers are used. Classifier-1 detects simple stroke features like vertical and horizontal lines. The classifier one is designed to separate the basic characters, modifiers and compound characters. . Characters like punctuation marks, special marks likes quotes are also recognized by this classifier. Classifier-2 extracts features from individual characters by counting the horizontal and vertical black runs. Recognition is done by calculating distance (dissimilarity) measure between character and stored prototypes. In the post processing stage dictionary match and morphological analyzer are used to select the correct word from the set of alternatives.

Kannada OCR

Ashwin & Sastry (2002) developed a font and size-independent OCR for Kannada. Text page is binarized using a global threshold computed automatically. Skew correction is done by a windowed Hough transform technique. Line and word segmentation are done by projection profile based methods. For segmentation, the words are first split into three vertical zones based on the horizontal projection for the word. The three zones are then horizontally segmented using their vertical projections. A character is segmented into its constituents, i.e. the base consonant, the vowel modifier and the consonant conjunct.

The features of Kannada OCR developed at RCILTS, Indian Institute of Science, Bangalore are as follows .Input image scanning is done at 300 dpi. Binarization is done using global threshold. Skew detection and correction are done using Hough transform technique. Line and Word Segmentation are based on projection profiles. Words are segmented into subcharacter level so that each akshara may be composed of many segments. Distribution of the ON pixels in the radial and the angular directions are extracted to capture the rounded shape of the Kannada characters. Classification based on the Support Vector Machines is adopted. Recognition accuracy is reported as 85% for the aksharas(Kannada characters).

OCR for Tamil

Siromony et al (1978) described a method for recognition of machine printed letters of the Tamil alphabet using an encoded character string dictionary. The scheme employs string features extracted by row- and column-wise scanning of character matrix. The features in each row /column are encoded suitably depending upon the complexity of the script to be recognized.

Chinnuswamy & Krishnamoorthy (1980) proposed an approach for hand-printed Tamil character recognition. Here, the characters are assumed to be composed of linelike elements called primitives, satisfying certain relational constraints. Labelled graphs are used to describe the structural composition of characters in terms of the primitives and the relational constraints satisfied by them. The recognition procedure consists of converting the input image into a labelled graph representing the input character and computing correlation coefficients with the labelled graphs stored for a set of basic symbols. The algorithm uses topological matching procedure to compute the correlation coefficients and then maximizes the correlation coefficient.

Seethalakshmi et al (2005) described a Tamil OCR in Unicode. After preprocessing, the individual character glyphs are segmented into 32×32 size. Features such as character height, character width, number of horizontal lines (long and short), number of vertical lines (long and short), horizontally oriented curves, the vertically oriented curves, number of circles, number of slope lines, image centroid, and special dots are computed. The extracted features are passed to a Support Vector Machine where the characters are classified by supervised learning algorithm. These classes are mapped onto Unicode for

recognition. Then the text is reconstructed using Unicode fonts. Performance comparison of three types of classifiers, viz, rule based classifier, back propagation based artificial neural network classifier and support vector machine based classifiers are studied.

Aparna & Chakravarthy (2002) detailed a complete OCR for Tamil magazine documents. Radial basis function neural network is used for separating text and graphics. For skew correction, cumulative scalar products (CSP) of windows of the text boxes at different orientations with the Gabor filters are computed. Orientation with the maximum CSP gives the skew. Ostus method is used for binarization. Line segmentation is done using horizontal projection. Inclined projections are used for segmenting lines into words and characters. A Radial basis function neural network is trained for character recognition. Response of 40 Gabor filters with 10 filters in each of the 4 directions is computed. The recognition accuracy is reported to be varying from 90-97 %.

The Tamil OCR developed at Indian Institute of Science, Bangalore has the following features (Aparna & Ramakrishnan 2001), Input image scanning is done at 300 dpi into a binary image. Skew detection and correction are achieved through Hough transform and Principal Component Analysis. Horizontal and vertical projection profiles are employed for line and word detection, respectively. Connected component analysis is performed to extract the individual characters. The segmented characters are normalized to predefined size and thinned before recognition phase. Depending on the spatial spread of the characters in the vertical direction, they are grouped into 4 classes. These classes are further divided into groups based on the type of ascenders and descenders in the characters. Second order moments are employed as features to perform this grouping. Truncated Discrete Cosine transform (DCT) based features are used for the final classification with a nearest neighbor classifier. Recognition accuracy of 98% on a sample size of 100 is reported.

Malayalam OCR

NAYANA is the Malayalam OCR developed at C-DAC, Thiruvananthapuram. Binarization is done using histogram based thresholding approach (Otsu ' s algorithm).Skew detection is done using the projection profile based technique. Linguistic rules are applied to the recognized text in the post processing module to correct classification errors. Recognition speed of 50 characters per second and accuracy of 97% for good quality printed documents are reported.

Applications of Character Recognition System

Optical Character Recognition has a wide range of applications in various areas. It can be used as a telecommunication aid for postal address reading for the deaf, processing of documents, in recognition of foreign language and also for language translation. In bill processing systems it is used to read payment slips like electricity bills, telephone / water bills. It will read and recognize the amount to be paid and also recognize the account number. The character recognition system can also be used for reading the address, assigning Zip codes to letters, application forms, voter ID cards (refer Fig.8), and identification of bank cheques by recognizing the account number and the amount written on the cheque (refer Fig.9). These systems can also be used in automatic processing of issuing tickets to air line passengers, validation of passports and visa cards etc. Address readers in postal departments locates the address on letters and sorts them according to their location using the zip code. The multiline optical character reader (MLOCR) by United States Postal Services (USPS) locates the address block on a mail piece, reads the address, identifies ZIP and generates a 9-digit bar code and sorts the mail to the correct stacker. This classifier recognizes up to 400 fonts and the system can process up to 45,000 mail pieces per hour. OCR helps in the recognition of hand written text and converts it into a scanned image (refer Fig. 10). CAPTCHA generated on different sites for security purposes is a vital application of OCR system (refer Fig. 11). Moreover, Automatic number plate recognition is used as a mass surveillance technique making use of optical character recognition on images to identify vehicle registration plates (refer Fig. 12). Some of the applications of an OCR are shown in the figures given below.

Figure 8. Reading of voter ID card by an OCR

Figure 9. Reading of bank cheques by an OCR

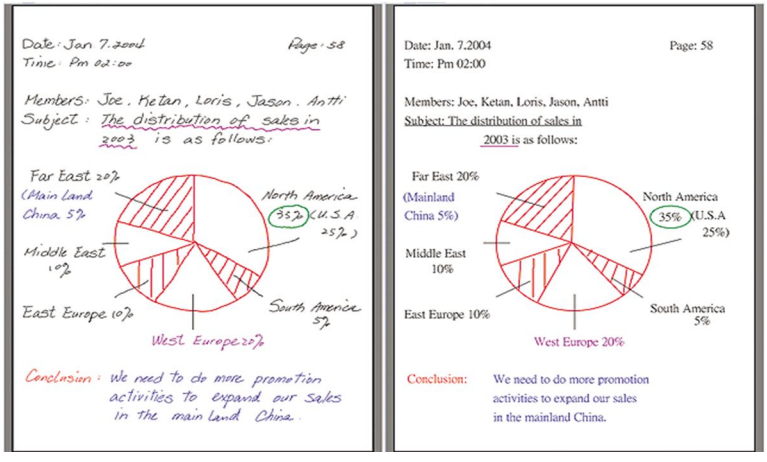

Figure 10. Handwriting recognition by an OCR

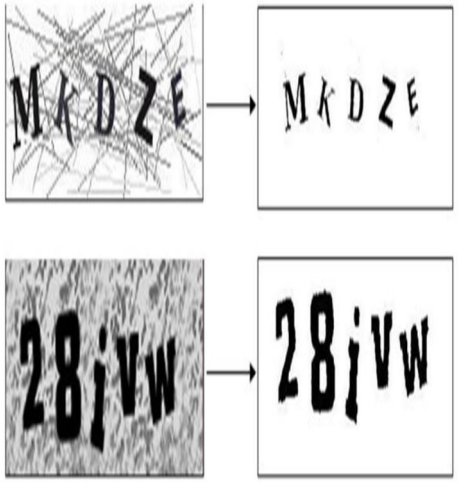

Figure 11. Segmentation of captcha by an OCR

Figure 12. Number plate recognition recognition by an OCR

Conclusion

As discussed earlier in this paper that there are different scripts in India, so there are lots of research work further to do in the field of Indian languages, the work of pattern recognition in Indian languages started very late than European or any other foreign language but still lots of work have been done for printed Indian scripts with good accuracy at character level, but work for hand written material is still in the way and less work done at word level for printed text also so this field have wire area to research.

There are many factors such as noise, various font sizes, broken lines or characters, quality of the image, problems in segmentation that influence recognition process. India is a multi lingual country; so many more efficient and real-time text recognizers are required. A good text recognizer has many commercial and practical applications. Hence there is a need to develop a very good character recognition system which must achieve highest accuracy. A study is made on different optical character recognition systems developed for Indian scripts. The technologies of these OCRs are discussed at length in this paper, which can be used as a starting step for the researchers entering into this area.

References

[1] B.Anuradha srinivas, Arun agarwal, and C.Raghavendra rao, “An Overview of OCR Research in Indian Scripts”, IJCSES International Journal of Computer Sciences and Engineering Systems, Vol.2, No.2, April 2008.

[2] M. Antony Robert Raj, Dr.S.Abirami “A Survey on Tamil Handwritten Character Recognition using OCR Techniques”, The Second International Conference on Computer Science, Engineering and Applications.

[3] Ms.M.Shalini , Dr.B.Indira, “Automatic Character Recognition of Indian Languages – A brief Survey”, IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 2, April 2014.

[4] Amandeep Kaur, Er. Manish Mittal, “Survey Paper on Hindi Digit Recognition”, IJCST Vol. 7, Issue 2, April - June 2016.

[5] G. S. Lehal and Chandan Singh, “Feature Extraction and Classification for OCR of Gurumukhi Script” ,2002.

[6] https://www.google.co.in/search?q=noise+removal+in+ocr&biw=1517&bih=708&source=lnms&tbm=isch&sa=X&ved=0ahUKEwjCjp3Y48PRAhWKNI8KHUNpCUAQ_AUIBygC#imgrc=w0y8tU8RQWNwiM%3A

[7] https://www.google.co.in/search?q=binarization+in+ocr&biw=1517&bih=708&source=lnms&tbm=isch&sa=X&ved=0ahUKEwjRvoG_5MPRAhXMRI8KHYdJAiYQ_AUIBigB#imgrc=l1IamWl6guhnWM%3A

[8] https://www.google.co.in/search?q=skewing+in+ocr&biw=1517&bih=708&source=lnms&tbm=isch&sa=X&ved=0ahUKEwif5tTS5MPRAhVHuI8KHSZ3BtMQ_AUIBigB#imgrc=_p9M9QOltwcHyM%3A

[9] https://www.google.co.in/search?q=ocr+for+bills&biw=1242&bih=557&source=lnms&tbm=isch&sa=X&ved=0ahUKEwj6lcHB05fSAhWDr48KHeNwCa8Q_AUIBygC#tbm=isch&q=reading+of+voter+id+cards+by+ocr&imgrc=kWkcRBpLT2TZAM:

[10] https://newram-india.blogspot.in/2011/10/ocr-system-for-telugu.html

[11] D Jayaram, Kamakshi Prasad, CRK Reddy, M Swamy Das, “An Overview of Optical Character Recognition Systems Research on Telugu Language”, IJSAT - International Journal of Science and Advanced Technology, Volume 2 No 9 September 2012.