International Journal of Computer Sciences and Engineering Systems
Vol. 12 No.2 (December, 2018)

An Overview of OCR Research in Indian Scripts

B. Anuradha Srinivas[1], Arun Agarwal[2] & C. Raghavendra Rao[2]

[1]Department of Computer Science, Vignan Institute of Technology & Science, Deshmukhi, Nalgonda Dist., Andhrapradesh
[2]Department of Computer & Information Science, University of Hyderabad, Andhrapradesh
E-mail: anuradhabs@yahoo.co.in

Manuscript received August 23, 2017, Manuscript revised March 15, 2018

Abstract: *This paper gives an overview of the ongoing research in optical character recognition (OCR) systems for Indian language scripts. This survey paper has been felt necessary when the work on developing OCRs for Indian scripts is very promising, and is still in emerging status. The aim of this paper is to provide a starting point for the researchers entering into this field. Peculiarities in Indian scripts, present status of the OCRs for Indian scripts, techniques used in them, recognition accuracies, and the resources available, are discussed in detail. Examples given in this paper are based on authors' work on developing a character recognition system for Telugu, a south Indian language.*

Keywords: *Optical character recognition; feature extraction; classification; Indian language OCRs*

1. INTRODUCTION

Optical character recognition, usually abbreviated as OCR, is the translation of handwritten or typewritten text into machine-editable form. Some practical application potentials of OCRs are: reading aid for the blind, preserving old/ historical documents in electronic format, desktop publication, library cataloging, ledgering, automatic reading for sorting of postal mail, bank cheques and other documents, etc.

Today, reasonably efficient and inexpensive OCR packages are commercially available to recognize printed texts in widely used languages such as English, Chinese, and Japanese. These systems can process documents that are typewritten, or printed. They can recognize characters with different fonts and sizes as well as different formats including intermixed text and graphics. While a large amount of literature is available for the recognition of Roman, Chinese and Japanese language characters, relatively less work is reported for the recognition of Indian language scripts.

Under the aegis of Technology Development for Indian Languages (TDIL)Programme, Ministry of Communications and Information Technology, Govt. of India, thirteen Resource Centers for Indian Language Technology Solutions (RCILTS) have been established at various educational institutes and research & development organizations covering all Indian Languages (JLT July 2004). Under this program, a number of OCRs, human–machine interface systems and other tools are being developed in different Indian languages. Thus, OCR systems for Indian scripts have just started appearing.

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. Several illustrations from the authors' work on developing a Telugu character recognition system (Anuradha Srinivas et al 2007) are used as examples.

2. INTRODUCTION TO OCR

Optical character recognition is the recognition of printed or written text by a computer. This involves photo scanning of the text, which converts the paper document into an image, and then translation of the text image into character codes such as ASCII.

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, paper quality, resolution of the scanned image, the amount of skew 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 (Anbumani & Subramanian 2000). Typical preprocessing includes the following stages:

- Binarization,
- Noise removing,
- Thinning,
- Skew detection and correction,

- Line segmentation,
- Word segmentation, and
- Character segmentation

Recognition consists of

- Feature extraction,
- Feature selection, and
- Classification.

Figure 1 depicts this sequence and they are described in the following sections.

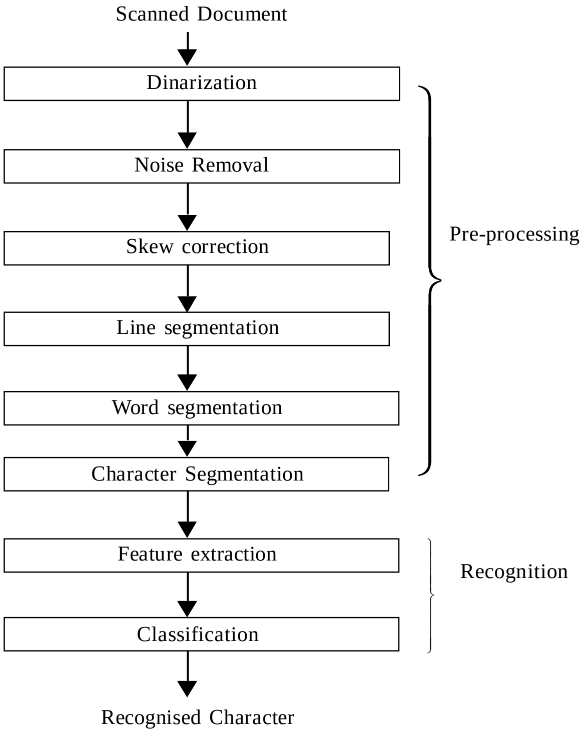
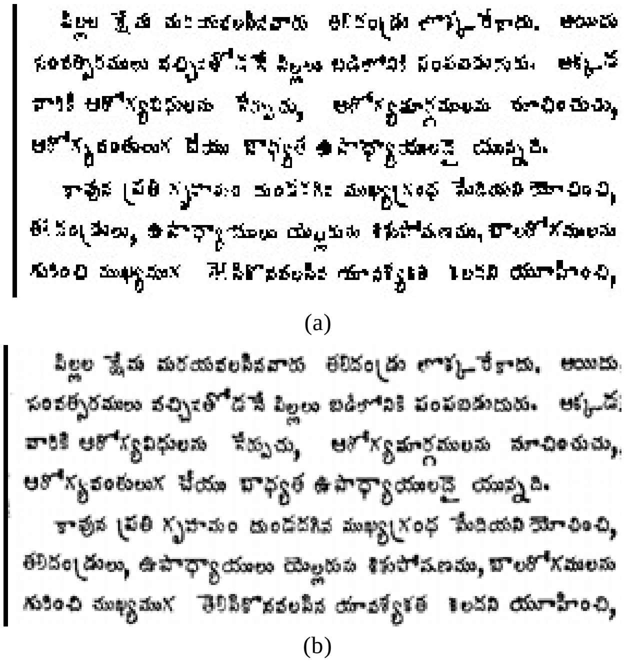


Figure 1: Steps in an OCR

2.1 Binarization

Binarization is a technique by which the gray scale images are converted to binary images. In any image analysis or enhancement problem, it is very essential to identify the objects of interest from the rest. Binarization separates the foreground (text) and background information. The most common method for binarization is to select a proper threshold for the intensity of the image and then convert all the intensity values above the threshold to one intensity value (for example "white"), and all intensity values below the threshold to the other chosen intensity ("black"). Binarization is usually reported to be performed either globally or locally. Global methods apply one intensity value to the entire image. Local or adaptive thresholding methods apply different intensity values to different regions of the image. These threshold values are determined by the neighborhood of the pixel to which the thresholding is being applied. Several binarization techniques are discussed in (Anuradha & Koteswarrao 2006).

Figure 2(a) shows the scanned image of a paper document printed in Telugu, a south Indian language. Figure 2(b) is the same image after binarization in which the text pixels are separated from the background.


(a)

(b)

Figure 2: (a) Original image, (b) Binarized image

2.2 Noise Removal

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. The commonly used approach is to low-pass filter the image and to use it for later processing. The objective in the design of a filter to reduce noise is that it should remove as much of the noise as possible while retaining the entire signal (Rangachar *et al.* 2002).

2.3 Thinning

Thinning, or, skeletonization or is a process by which a one-pixel-width representation (or the skeleton) of an object is obtained, by preserving the connectedness of the object and its end points (Gonzalez &Woods 2002). The purpose of thinning is to reduce the image components to their essential information so that further analysis and recognition are facilitated. This enables easier subsequent detection of pertinent features. Figure 3 shows an image before and after thinning. A number of thinning algorithms have been proposed and are being used. The most common algorithm used is the classical Hilditch algorithm (Rangachar et al 2002) and its variants.

2.4 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

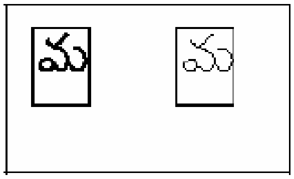


Figure 3: A Character Image (left) before Thinning, and (b) After Thinning.

unavoidable. Skew angle is the angle that the lines of text in the digital image make with the horizontal direction. Figure 4(a) shows an image with skew.

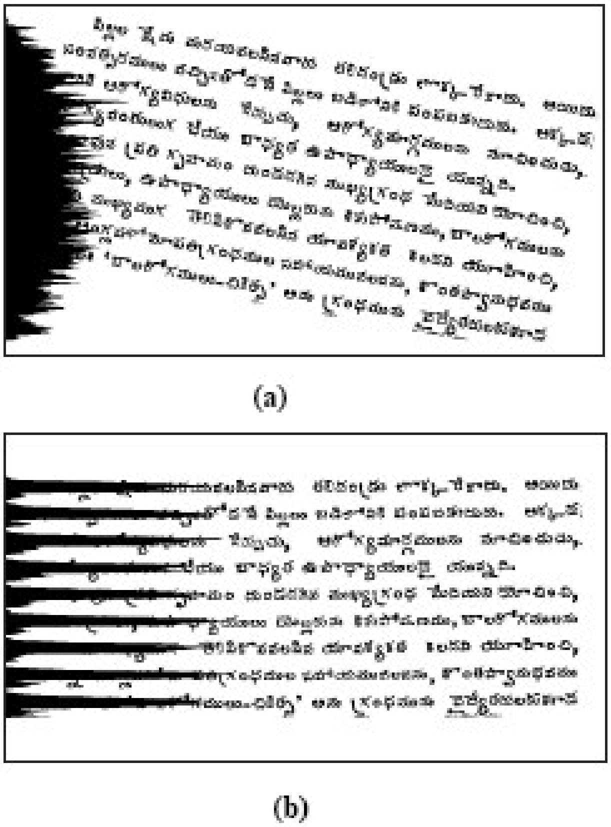


Figure 4: An Image (a) with Skew, (b) without Skew, and its Horizontal Profiles

There exist many techniques for skew estimation. One skew estimation technique is based on the projection profile of the document; another class of approach is based on nearest neighbor clustering of connected components. Techniques based on the Hough transform and Fourier transform are also employed for skew estimation. A survey on different skew correction techniques can be found in Chaudhuri & Pal (1997). A popular method for skew detection employs the projection profile. A horizontal projection profile is a one-dimensional array where each element denotes the number of black pixels along a row in the image. For a document whose text lines span horizontally, the horizontal projection profile has peaks whose widths are equal to the character height and valleys whose widths are equal to the spacing between lines. At the correct skew angle, since scan lines are aligned to text lines, the projection profile has maximum height peaks for text and valleys for line spacing. In the image of figure 4(a), its horizontal projection profile can be seen with no clear valleys due to the presence of skew. Figure 4(b) is an image in which the skew is removed. The peaks and valleys in the projection profile can be clearly seen.

2.5 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. Horizontal projection of a document image is most commonly employed to extract the lines from the document. If the lines are well separated, and are not tilted, the horizontal projection will have separated peaks and valleys, as shown in figure 4(b), which serve as the separators of the text lines. These valleys are easily detected and used to determine the location of boundaries between lines.

Figure 5 shows an image consisting of 3 text lines (left), and the 3 segmented lines (right), using horizontal projection profiles. Similarly a vertical projection profile gives the column sums. One can separate lines by looking for minima in horizontal projection profile of the page and then separate words by looking at minima in vertical projection profile of a single line. Valleys in the vertical projection of a line image can be used in the extraction of words in a line, as well as extracting individual characters from the word.

Figure 6(a) shows a line consisting of 4 words, along with vertical projection profiles, and figure 6(b) shows the 4 words, after segmentation.

In Figure 6(c), a word is shown segmented into its constituting 3 characters. Overlapping, adjacent characters in a word (called kerned characters) cannot be segmented using zero-valued valleys in the vertical projection profile. Special techniques have to be employed to solve this problem.

2.6 Feature Extraction and 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. Thinned data is analyzed to detect features such as straight lines, curves, and significant points along the curves.

Feature selection approaches try to find a subset of the original features. The strategy used for OCR can be broadly classified into three categories:

1. Statistical Approach
2. Syntactic/ structural Approach
3. Hybrid Approach

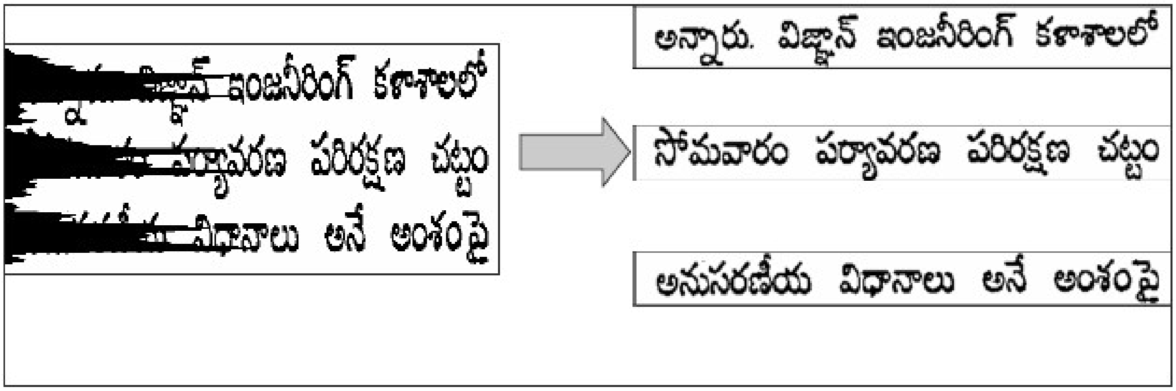

అన్నారు. విజ్ఞాన్ ఇంజనీరింగ్ కళాశాలలో
సోమవారం పర్యావరణ పరిరక్షణ చట్టం
అనుసరణీయ విధానాలు అనే అంశంపై

Figure 5: Line Segmentation

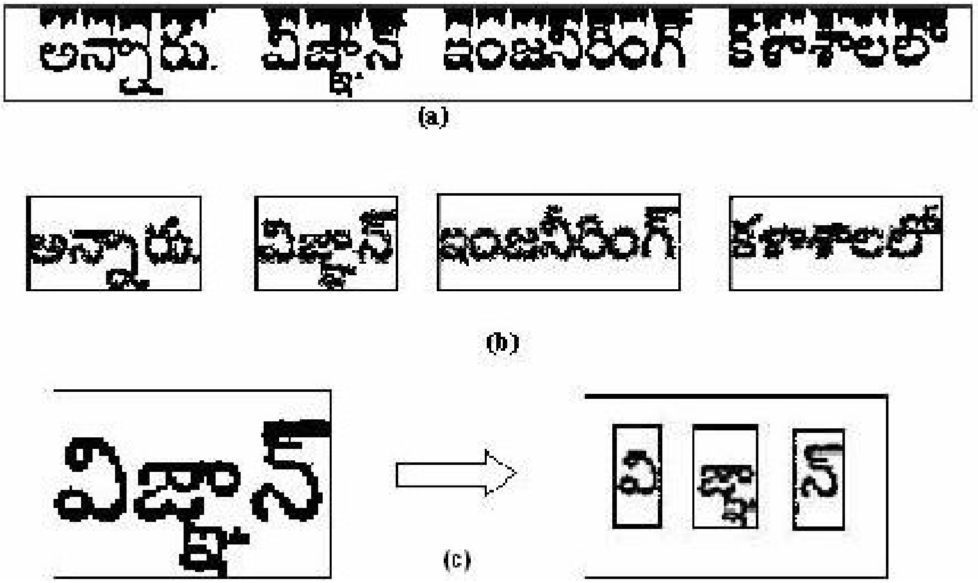


Figure 6: (a) A Line Segment, (b) Word Segmentation, (c) Character Segmentation

In statistical approach, a pattern is represented as a vector: an ordered, fixed length list of numeric features. Many samples of a pattern are used for collecting statistics. This phase is known as the training phase. The objective is to expose the system to natural variants of a character. Recognition process uses this statistics for identifying an unknown character. Features derived from the statistical distribution of points include geometrical moments, and black-to-white crossings.

Structural classification methods utilize structural features and decision rules to classify characters. Structural features may be defined in terms of character strokes, character holes, end points, loops or other character attributes such as concavities. The classifier is expected to recognize the natural variants of a character but discriminate between similar looking characters such as O and Q, c and e, etc.

The statistical approach and structural approach both have their advantages and disadvantages. The statistical features are more tolerant to noise provided the sample space over which training has been performed is representative and realistic than structural descriptions. The variation due to font or writing style can be more easily abstracted in structural descriptions. In hybrid approach, these two approaches are combined at appropriate stages for representation of characters and utilizing them for classification of unknown characters.

2.7 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.

Template matching is one of the most common and oldest classification methods. In template matching, individual image pixels are used as features. Classification is performed by comparing an input character image with a set of templates (or prototypes) from each character class.

The template, which matches most closely with the unknown, provides recognition.

Classification strategies following feature extraction are mostly based on identification of a neighbor pixel with the nearest distance. A distance measure between the vectors is used as the similarity between the images. Binary- tree classifiers and the nearest-neighbor classifiers are the two most commonly used classifiers. The frequently used distance calculation is the Euclidean distance measure.

3. INDIAN LANGUAGE OCRS

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. 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. Some of the peculiarities in Indian scripts are explored in section 3.1.

3.1 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.

Most Indian languages have around fourteen vowels and thirty-six consonants resulting in a total of 50 or even more basic characters. Vowels occur either in isolation or in combination with consonants. Apart from vowel and consonant characters called basic characters, there are compound characters in most Indian script alphabet systems (except Tamil and Gurumukhi scripts) which are formed by combining two or more basic characters (Pal & Chaudhuri 2004). The shape of a compound character is usually more complex than the constituent basic characters. Coupled to this, in some languages there is a practice of having more than twelve forms each for the thirty six consonants, giving rise to modified shapes which, depending on whether the vowel is placed to the left, right, top or bottom of the consonant. They are called modified characters. The net result is that there are several thousand different shapes or patterns, which may, in addition, be connected with each other without any visible separation. This makes the Indian OCRs more difficult to develop. Nevertheless, under the aegis of TDIL Programme, Resource Centers for Indian Language Technology Solutions (RCILTS) developed OCRs in different Indian languages. A brief summary of the techniques used in these OCRs as given in JLT (October 2003) along with other reported works are described in section 3.2.A detailed performance reports as given in Language Technology Products Testing Reports from July 2004 news letter of TDIL (JLT July 2004) are presented in section 4.

3.2 Techniques used in Different Indian Script OCRs

Any OCR contains more or less the same steps described in section 2.The exact number and techniques differ slightly from one language to other. We now present the studies in different OCRs, along with a detailed 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 Freeman-chaincode 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. This is represented pictorially in figure 7(a) and (b). Chain code gives the boundary of the character image; 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. Using chain code representation, classification is done by a feed forward neural network. Testing on three types of fonts with accuracy of approximately 98% for isolated characters and 96% for continuous characters is reported.

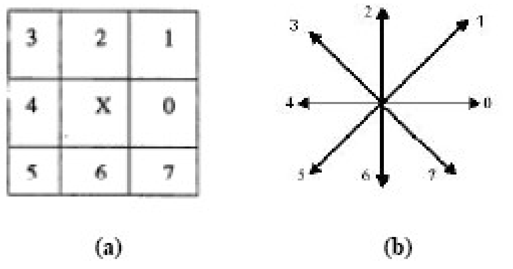


Figure 7: Chain Code and Graphical Representations

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. A histogram-based thresholding approach is used to convert the image into binary images. 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: Lehal and Singh presented an OCR system for printed Gurumukhi script. (Lehal &Singh 2000). 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 similar-looking 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. The rule base consists of more than one rule for a given character to account for different font-specific representations of the same character. 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. Observing the coverage of core strip, vertical bar features, horizontal zero crossings, number and position of vertex points, and moments, etc does the classification. 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. Images are uniformly scaled to 32x32 using zero-padding techniques. Wavelet representation with three levels of down sampling reduces a 32x32 image into a set of four 8x8 images, of which only an average image is considered for further processing. 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 (glyphs). Run Length Smearing Algorithm (RLSA) (Wong *et al.* 1982) and Recursive XY Cuts (Nagy *et al.* 1992) methods are used to segment the input document image into words. About 370 connected components (depending on the font) are identified as sufficient to compose all the characters including punctuation marks and numerals. Template matching based on the fringe distance (Brown 1994) 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 in the glyphs. 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). Some of them are: identification of glyph position information, and recognizing punctuation marks from the width, and height information, etc. The authors observed that handling of confusion pairs of glyphs, and touching characters would improve the OCR recognition accuracy further. Overall, 25% improvement is expected from considering the above factors.

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 and pepper 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.

Jawahar *et al.* (2003) proposed a Bilingual OCR for Hindi-Telugu documents. It is based on Principal Component Analysis followed by support vector classification. An overall accuracy of approximately 96.7% 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 pg 110-113). The techniques used in Drishti are as follows:

For binarization three options are provided: global (the default), 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. The template with the best matching score is output as the recognized glyph.

Anuradha Srinivas, *et al.* (2007) developed a Telugu optical character recognition system for a single font. Sauvola’s 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. Zero-crossing 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 similar-looking 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 (JLT July 2003. pg 28). 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) (JLT July2003 pg 59), 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%.

Adaptation of Bangla OCR to Assamese is also reported in JTL (2004).Since Assamese and Bangla share the same script, the Bangla OCR system is tried for Assamese documents after some modifications in the post processing stage where language specific OCR error correction is needed. Character-level accuracy of about 95% is reported.

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 as given below.

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: The features of Assamese OCR are based on the Bangla OCR developed at Indian Statistical Institute, Kolkata (JLT October 2003) , and are as given below.

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 (a black run is a set of black pixels with white pixel at either end) 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 (JLT July 2003 pg. 104).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 sub-character 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 line-like 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. Ostu's 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), (JLT July 2003 pg 103): 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 (JLT July 2003 pg.137).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.

4. TESTING

In order to address the issues of quality in the developed language technology products, Department of Information Technology (DIT) initiated a project entitled- Software Quality Engineering in Indian Language Products (JLT July 2004 pg50). The OCRs developed at the RCILTs are tested for functionality, usability and portability. In all cases of testing, text from newspapers, books and laser printed output were used as test inputs to be scanned and then presented to the OCR for converting to computer editable text. Standardization Testing and Quality Certification (STQC) division of DIT was designated as the third party for evaluation of the language technology tools and products developed at TDIL programme. The performance results (JLT July 2004 pgs.53,54) of OCRs tested by STQC are now tabulated in Table 1.

5. CONCLUSIONS

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] Aparna K. G., Ramakrishnan A. G., 2002, A Complete Tamil Optical Character Recognition System. 5th International Workshop on Document Analysis Systems DAS 2002, Princeton, NJ, USA, pp. 53-57.

[2] Aparna K. G., Ramakrishnan A. G., 2001, Tamil Gnani – an OCR on Windows, Proc. Tamil Internet 2001, Kuala Lumper, pp. 60-63.

[3] Anbumani, Subramanian 2000, Optical Character Recognition of Printed Tamil Characters. Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg.

[4] Antani Sameer, Lalitha Agnihotri, 1999, Gujarati Character Recognition. Fifth International Conference on Document Analysis and Recognition (ICDAR'99), p. 418.

[5] Anuradha B., Koteswarrao B. 2006, An Efficient Binarization Technique for Old Documents. Proc. of International Conference on Systemics,Cybernetics, and Inforrmatics (ICSCI2006), Hyderabad, pp. 771-775.

[6] AnuradhaSrinivas, Arun Agarwal, C. R. Rao 2007, Telugu Character Recognition. Proc. of International Conference on Systemics, Cybernetics, and Informatics, Hyderabad, pgs. 654-659.

[7] Ashwin T. V. and P. S. Sastry 2002, A Font and Size-Independent OCR System for Printed Kannada Documents using Support Vector Machines. Saadhanaa, **27**(1), 2002, 35–58.

[8] Brown R. L. The Fringe Distance Measure: An Easily Calculated Image Distance Measure with Recognition

Table 1
Test Results from STQC

OCR Name	*Developed by*	*Platform*	*Font Size*	*Accuracy*	*Speed (CPS)*	*Skew angle*	*Input*	*Output file*	*Supports*	*Resolution*
Bangla	ISI, Kolkata	DOS	14-36 pts.	30-96%*	13 to 47.	+/–5°.	TIFF	PC	Single column, bold character, but not symbols	300dpi
Gurmukhi	TIET, Patiala	WIN 9X	12-20 pts.	93-99%*	12 to 59	+/–5°.	BMP	Punjabi font in Word	Single column bold character, but not symbols	300 dpi
Tamil	IISC, Bangalore	WIN 95/98	12-18 pts.	62-98%*	1 to 19	+/–10°.	BMP. PGM	TAB, RTF	Single column, bold character & special symbols	300 dpi
Devnagari	ISI Kolkata	WIN95 (Min)/Linux	10-24 pts	48-97%*	20-44	+/–5°.	TIFF	PC	Single column, bold character, but not special symbols	300 dpi
Hindi/ Marathi	CDAC, Pune	WIN 95X/NT/ 2000	10-36 pts	85-98%*	93-368	+/–5°	BMP, JPG TIFF	RTF UNICODE/ ISCII	Multi column, bold characters, but not special symbols	270-320
Telugu	DCIS, University of Hyderabad	Linux	12-20 pts	84-87%*	23-29	NM*	PGM	ACI	Single column	300 dpi
Hindi	CDAC, Noida	WIN 98/2000	12-36 pts.	5-96%*	9-188	+/–5°	TIFF,BMP	RTF	single column, bold characters, but not special symbols	280-320 dpi
Oriya	Utkal University Bhubaneswar	Windows	NM*	74-86%*	NM*	NM*	BMP	TXT	single column text	300 dpi
Malayalam	CDAC Thiruvanthapuram	windows	NM*	93-97%	NM*	+/–5° JPEG and	BMP, TIFF HTML GIF	RTF, formats multicolumn ACI, TXT	multiple font sizes, layouts containing both text & images	300 dpi

* Depending on type of inputs.

*NM=Not mentioned.

Results Comparable to Gaussian Blurring. IEEE Trans.System Man and Cybernetics, **24**(1), 111-116, 1994.

[9] Chakravarthy Bhagvati, T. Ravi, S. M. Kumar, and Atul Negi. 2003 On Developing High Accuracy OCR Systems for Telugu and other Indian Scripts. Proc. of Language Engineering Conference, Pp 18-23, Hyderabad, IEEE Computer Society Press.

[10] Chaudhuri B. B. and U. Pal 1997, Skew Angle Detection of Digitized Indian Script Documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**(2), 1997.

[11] Chaudhuri, B. B. and Pal, U. 1998, A Complete Printed Bangla OCR System. Pattern Recognition, **31**, 1998, 531-549.

[12] Chinnuswamy P., S. G. Krishnamoorty, 1980, Recognition of Hand-printed Tamil Characters. Pattern Recognition, **12**(3), 141–152.

[13] Garain U. and B. B. Chaudhuri, 2002, Segmentation of Touching Characters in Printed Devnagari and Bangla Scripts using Fuzzy Multifactorial Analysis. *IEEE Transactions on Systems, Man and Cybernetics, Part C*, **32**(4), 449-459.

[14] Gonzalez R. C., and R. E. Woods 2002, Digital Image Processing. (New Jersey: Prentice-Hall).

[15] Jalal Uddin Mahmud, Mohammed Feroz Raihan and Chowdhury Mofizur Rahman 2003, A Complete OCR System for Continuous Bengali Characters. Conference on Convergent Technologies for Asia-Pacific Region (TENCON) **4**(15-17), 1372–1376.

[16] Jawahar C. V., M. N. S. S. K. Pavan Kumar, S. S. Ravi Kiran 2003. A Bilingual OCR for Hindi-Telugu Documents and its Applications. International Conference on Document Analysis and Recognition,

Journal of Language Technology, Vishwabharat@tdil, July 2003,

Journal of Language Technology, Vishwabharat@tdil, October, 2003.

Journal of Language Technology, Vishwabharat@tdil, July,2004.http://www.tdil.mit.gov.in/July05/extract/indian%20language%20technology%20resource%20center.pdf

Journal of Language Technology, Vishwabharat@tdil. July,2004,pgs53-54 http://www.tdil.mit.gov.in/July05/extract/indian%20language%20technology%20testing%20reports.pdf

[17] Lakshmi C. V., C. Patvardhan, 2003, Optical Character Recognition of Basic Symbols in Printed Telugu Text. IE(I)Journal-CP **84**, 66-71.

[18] Lakshmi C. V., C. Patvardhan, 2002, A Multi-font OCR System for Printed Telugu Text. Proc. of Language Engineering Conference LEC, Hyderabad. pgs. 7-17.

[19] Lakshmi C. V., C. Patvardhan 2003 A High Accuracy OCR for Printed Telugu Text. Conference on Convergent Technologies for Asia-Pacific Region (TENCON) **2**(15-17), 725-729.

[20] Lehal G. S. and Chandan Singh 2000, A Gurmukhi Script Recognition System 15th International Conference on Pattern Recognition (ICPR'00), **2,** 2557.

[21] Lehal G. S. and Chandan Singh, 2002, A Post-processor for Gurmukhi OCR Saadhana **27**, Part 1, February, pp. 99–111.

[22] Mohanty S., H. K. Behera, 2004, A Complete OCR Development System for Oriya Script. Proceeding of Symposium on Indian Morphology, Phonology and Language Engineering, IIT Kharagpur.

[23] Nagy G. , S. Seth, and M. Vishwanathan 1992, A Prototype Document Image Analysis System for Technical Journals. Computer, 25(7).

[24] Negi Atul, Chakravarthy Bhagvati and Krishna B. 2001, An OCR System for Telugu. Proc. of 6th Int. Conf. on Document Analysis and Recognition IEEE Comp. Soc. Press, USA,. Pgs. 1110-1114.

[25] Negi Atul, Chakravarthy Bhagvati, and V.V.Suresh Kumar. 2002 Non-linear Normalization to Improve Telugu OCR Proc. of Indo-European Conf. on Multilingual Communication Technologies, pgs 45-57, Tata McGraw Hill Book Co., New Delhi,

[26] Negi Atul, Nikhil Kasinadhuni 2003, Localization and Extraction of Text in Telugu Document Images Conference on Convergent Technologies for Asia-Pacific Region (TENCON) pgs. 749-752.

[27] Pal U., B. B. Chaudhuri 2004, Indian Script Character Recognition: A Survey. Pattern Recognition 37 pgs.1887 –1899.

[28] Pal U., B. B. Chaudhuri 1997, Printed Devnagari Script OCR System. Vivek, **10**, 12-24.

[29] Parvati Iyer, Abhipsita Singh, S. Sanyal 2005, Optical Character Recognition System for Noisy Images in Devnagari Script. UDL Workshop on Optical Character Recognition with Workflow and Document Summarization.

[30] Pujari Arun K., C. Dhanunjaya Naidu & B. C. Jinaga 2002, An Adaptive Character Recognizer for Telugu Scripts using Multiresolution Analysis and Associative Memory. ICVGIP, Ahmedabad.

[31] Rajasekaran S. N. S. Deekshatulu B. L. 1977, Recognition of Printed Telugu Characters. Comput. Graphics Image Processing, 6 pgs. 335–360.

[32] Rangachar Kasturi, Lawrence O'Gorman and Venu Govindaraju 2002, Document Image Analysis: A Primer. Saadhanaa **27**, Part 1, pp. 3–22.

[33] Rao P. V. S. & T. M. Ajitha 1995, Telugu Script Recognition-a Feature Based Approach. Proce. of ICDAR, IEEE pgs. 323-326,.

[34] Ray K., and Chatterjee B. 1984, Design of a Nearest Neighbor Classifier System for Bengali Character Recognition. *Journal of . Inst. Electronics. Telecom. Eng.* 30 pgs.226–229.

[35] Seethalakshmi R., Sreeranjani T. R., Balachandar T., Abnikant Singh, Markandey Singh, Ritwaj Ratan, Sarvesh Kumar 2005 Optical Character Recognition for printed Tamil text using Unicode. *Journal of Zhejiang University* SCI 6A(11) pgs. 1297-1305.

[36] Sinha R. K, Mahabala 1979, Machine Recognition of Devnagari Script. *IEEE Trans. Systems Man Cybern.* Pgs. 435–441.

[37] Siromony, G. Chandrasekaran R., Chandrasekaran M. 1978 Computer Recognition of Printed Tamil Characters. Pattern Recognition **10,** 243–247.

[38] Sukhaswami R., Seetharamulu P., Pujari A. K. 1995, Recognition of Telugu Characters using Neural Networks, *Int. J. Neural Syst.* **6,** 317–357.

[39] Veena Bansal 1999, Integrating Knowledge Sources in Devnagari Text Recognition. Ph.D. Thesis, IIT Kanpur.

[40] Wong K., Casey R., and.Wahl F. 1982, Document Analysis System. *IBM Journal of Research and Development*, **26** (6).