International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-2, Issue-2, January 2013

Algorithm to Identify Kannada Vowels using Minimum Features Extraction method

K.S. Prasanna Kumar

Abstract— This paper introduces a novel way of feature extraction for Optical Character Recognition (OCR) customized for Kannada characters. The algorithm described here relies on breaking the character into four equal parts and using one of the quarters for extraction. The algorithm is deliberately kept away from all the complexities and the number of features to be extracted is also minimized so as to increase the efficiency and speed of recognition. The algorithm also describes a conflict resolution technique helpful in effectively utilizing the algorithm.

Keywords— Kannada OCR, Minimal Feature Extraction, Character recognition Algorithm, Conflict resolution.

I. INTRODUCTION

Kannada script has evolved from the Kadamba script in 5th century. It has a attested epigraphically for about one and half millennia. The earliest work extract is a treatise or poetics called by the King Kavirajamarga. The King even makes reference to writings of his earlier periods, which indicates the existence of language even before his time [26]. Recent references found in the 'Charition mime' of 1st or 2nd CE in Egypt indicates Kannada to have rich literary heritage. The numerical system and the Kannada script has undergone radical changes over a period of time and various kings and countrymen using their own language have contributed enormously to evolve the language in its present form. The script has also undergone number of transformations to reach the present stage/shapes. Unlike other languages, the different fonts make little effect on the overall shapes of the numbers. Printed numeral and character recognition is an integral part of the OCR and this paper focuses on recognizing vowels of the language. There are a number of authors who have developed algorithms for OCRs to recognize based on feature extraction. This paper proposes to extract minimum number of features and achieve better results.

II. KANNADA OCR

A. Kannada writing system

Kannada has forty nine phonemic letters. These are divided into three groups as swaragalu (vowels – thirteen letters), vyanjanagalu (Consonants – thirty four letters) and yogavaahakagalu (neither vowels nor consonants two letters). The language is more complicated due to various combinations of glyphs. Script has evolved in the stages of ProtoKannada to PreOldKannada to OldKannada and to ModernKannada. The ProtoKannada evolved from Brahmi in around c.3rd century CE. The PreOldKannada evolved around c.4th century CE. OldKannada script evolved in 10th Century CE and ModernKannada script around 17th Century CE. [27].

Manuscript received on January, 2013.
Prof. K.S. Prasanna Kumar, PG Department of Computer Science, Acharya Institute of Technology, Bangalore, India.

B. Major OCR components

The following are the main components of information extraction and processing in OCR:

Image acquisition: Acquire the candidate image's color, or gray levels from all regions of interest.
Form identification: Input image is to be stored in most suitable form for pre and post processing in a database.
Layout analysis: Understand semantic meaning of the image in its best forms.
Noise removal: Unwanted/unexpected extra acquisitions in the first stage. Various techniques of cleaning, smoothing, enhancing, pre-segmentation, normalization, etc are used to remove noise.
Data extraction: extract important features from candidate image after preprocess the data.
Character recognition: convert the gray or binary images that contain textual information to electronic representation of characters that facilitate post-processing.

C. Preprocessing

Pre-processing is the preliminary step which transforms the data into a form that is more easily and effectively processed. The main task of preprocessing is to process the captured data and decrease the noise that causes a reduction in the recognition rate and increases the complexities. Hence, preprocessing is an essential stage prior to feature extraction, as it controls the suitability of the results for the successive of the algorithm.

Global image thresholding is the process of distinguishing the image from its background, so thresholding is applied to grey-level candidate scanned image. Thresholding is categorized into two main categories i.e., global and local threshold. Global thresholding methods choose one threshold value for the entire document image, which is often based on the estimation of the background level from the intensity histogram of the image; hence, it is considered a point processing operation and local adaptive thresholding uses different values for each pixel according to the local area information. There are hundreds of thresholding algorithms which have been published.

Global thresholding method is used to reduce the grey-level image to a binary image. The candidate image is assumed to have two classes of pixels called foreground and background. The purpose of a global thresholding method is to automatically specify a threshold value, T, where the pixel values below it are considered foreground and the values above are background. A simple method is chosen where the mean or median value of all the pixels in the input image, the mean or median will work well as the threshold, however, this may not be the case if the pixels are not uniformly distributed in an image. A more sophisticated approach is to use a histogram of the image pixel intensities and use the valley point (minimum) as the threshold.

Local thresholding technique is used with the candidate image having non-uniform background illumination or different backgrounds. If the global thresholding method fails to separate the foreground from the background then this is due to the fact that the histogram of such image has more than two peaks making it difficult for a global thresholding technique to separate the objects from the background. Hence local thresholding method is used. The local thresholding technique developed for customized applications may not be consistent. The results could be over thresholding or under thresholding depending on the contrast and illumination.

III. RELATED WORK

A novel algorithm to Optical Character Recognition (OCR) for Kannada numerals is discussed [25]. The novelty exists in segmentation of the numeral into four equal parts and using one of these parts i.e., left bottom segment to extract recognition features. The algorithm also proposes a single conflict resolution technique to resolve conflicts while conflicting features are encountered. A minimum number of features are extracted by the algorithm so as to improve the response time.

In Brown et al. [1], a recognition system for the unconstrained handprinted numerals was proposed, which used topological, geometrical and local measurements to identify the character or to reject the character as unrecognizable. The recognition system yielded a recognition rate of 97% with a substitution error rate of 0.3% and a rejection rate of 2.7%.

In L.Stringa [2], a pattern recognition system was applied to the unconstrained alphanumeric character recognition. The recognition system was designed to allow hierarchical re-description of the input images and the phrase-structure grammars were developed. The experiments conducted on handwritten digits indicated that the recognition rates were comparable to the best OCR system at that time, but with a considerable reduction in computing time.

In Suen et al. [3], four experts for the recognition of handwritten digits were proposed. In expert one, the skeleton of a character pattern was decomposed into branches. The pattern was then classified according to the features extracted from these branches. In expert two, a fast algorithm based on decision trees was used to process the more easily recognizable samples, and a relaxation process was applied to those samples that could not be uniquely classified in the first phase. In expert three, statistical data on the frequency of occurrence of features during training were stored in a database. This database was used to deduce the identification of an unknown sample. In expert four, structural features were extracted from the contours of the digits. A tree classifier was used for classification. The resulting multiple-expert system proved that the consensus of these methods tended to compensate for individual weakness, while preserving individual strengths. The high recognition rates were reported and compared favorably with the best performance in the field.

Mitchell and Gillies [4] used the tools of mathematical morphology to extract cavity features as the starting input for their specialized digit recognizers. A classification system was implemented by a symbolic model matching process.

Le Cun et al. [5] achieved excellent results with the convolutional neural networks, which were specifically designed to deal with the variability of two dimensional (2-D) shapes. For the recognition of handwritten numerals, the recognition rate with this method could be as high as 99.18% on the MNIST database. Recently, many improvements have been reported, especially in pursuing a higher recognition rate.

In Simard et al [105], authors expanded the training set of the MNIST dataset by adding a new form of distorted data, and the convolutional neural networkswere better suited for classification purposes. The recognition rate was achieved at 99.60%.

Shi et al. [6] proposed a handwritten digit recognition system using the gradient and curvature of the gray character image in order to improve the accuracy of handwritten numeral recognition. The experiments were conducted on IPTP CDROM1, NIST SD3, and SD7 databases. The recognition rates could reach from 98.25% to 99.49%.

Teow and Loe [7] proposed a handwritten digit recognition system based on a biological vision model. The features were empirically extracted by the model, which could linearly separate over a large training set (MNIST). The high recognition rate was reported, where the error rate was 0.59%.

Decoste and Scholkopf [8] proposed a handwritten digit recognition system where the prior knowledge about invariance of a classification problem was incorporated into the training procedure. Support Vector Machines (SVMs) were used as classifiers. The system achieved a low error rate of 0.56% when using this procedure with the MNIST dataset.

IV. EXISTING FEATURE EXTRACTION METHODS

The purpose of feature extraction is to get the most relevant and the least amount of data representation of the character images in order to minimize the within-class pattern variability while enhancing the between-class pattern variability. There are two categories of features: statistic features and structural features. In the statistic feature domain, Hu introduced the use of moment invariants as features for pattern recognition. Hu’s absolute orthogonal moment invariants (invariant to translation, scale and rotation) have been extensively used in the recognition systems.

In Krzyzak et al.,[9] features were firstly extracted from the contours of numerals: 15 complex Fourier descriptors were extracted from the outer contours and simple topological features were extracted from the inner contours. These features were directly presented as the input of a three-layer ANN for recognition.

In recent years, wavelet transform has been an emerging tool for feature extraction. In Chen, Bui and Krzyzak’s paper[18], a multiwavelet orthonormal shell expansion was used on the contour of the character to get several resolution levels and their averages.

The shell coefficients were used as the features input into a feed-forward neural network to recognize handwritten numerals. Tao et al. [7] investigated the utility of several emerging techniques to extract features.

The central projection transformation was applied to describe the shape of the characters; then the wavelet transformation was used to aid in the boundary identification, and the fractal features were employed to enhance image discrimination for the recognition of printed Chinese characters and English letters of varying fonts.

In Lee’s and Kirsch[10] masks were adopted for extracting four directional local feature sets and one global feature set. A three-layer cluster neural network with five independent subnetworks was developed for classifying similar numerals.

In the structural feature extraction domain, in Suen et al. paper[3], the comprehensive structural features were systemically implemented, such as the combined branch features, giving information on the following: shape, length, angular change, degree of curvature, vertical and horizontal general directions, nature of the starting and ending points (J points and E points), their coordinates, the distance and the primitive features such as line segments, (open) convex polygons, and loops, etc.

Liu et al. [11] summarized state-of-the-art feature extraction techniques, which included the extraction of chaincode features, gradient features, profile structure features and peripheral direction contributivity. The recognition performance comparisons among different types of features were given in the paper.

In Gader et al., [12] a linear correlation feature extractor for handwritten digit recognition was described. Two different evaluation measures: orthogonality and information, were used to guide the search for features. ANNs with Back Propagation (BP) algorithms were used as classifiers in the recognition experiments of handwritten digits. The classification rates compared favorably with results published in the literature.

Weideman et al.[13] extracted 36 normalized moment features, 18 topological features, 24 2-D FFT features, and 16 shadow features that were found by projecting the character onto the nearest bars in the horizontal, vertical, and diagonal directions. The length of the shadow on each bar was used as a feature. The comparisons of a neural network and a nearest-neighbor classifier for the recognition of numeric handprint characters were reported.

Oh et al. [14] proposed two feature sets based on distance transformation. In the first feature set, the distance from each white pixel to the nearest black pixel in the character image without the thinning operation was considered as a distance transformation feature.

The second feature was called Directional Distance Distribution (DDD), which contained rich information encoding both black/white and directional distance distributions. A new method of map tiling was also introduced and applied to the DDD feature to improve its discriminative power. The experiments were conducted on three sets of characters (numerals, English letters, and Hangul initial sounds). The results confirmed the superiority of both the DDD feature and the map tiling.

In Yang et al.,[15] high-order B-splines were used to calculate the curvature of the contours of handwritten numerals. The concept of a distribution center was introduced so that a one-dimensional periodic signal could be normalized as a shift invariant. The curvature of the contour of a character became rotation invariant. ANNs and SVMs classifiers were employed to train the features. High verification rates on similar numeral pairs were reported.

Oliveira et al. [17] proposed a specific concavity, contour-based feature sets for the recognition and verification of handwritten numeral strings. The OCR system could process either isolated digits or handwritten numeral strings.

Gao and Ding [16] proposed two new feature extraction strategies: the modified multiple discriminant analysis and the difference principal component analysis. The proposed algorithms were useful in automatic feature extraction from different patterns.

Experiments have shown that the two new methods provided more effective feature metrics for pattern discrimination in the recognition of Chinese character fonts and handwritten digits.

V. NOISE DETECTION AND REMOVAL

Noise is broadly classified as Global and Local noise. This noise is an inherent addition to digitization. Based on the causes of this noise, they can be divided into noise caused by aging, storage or other physical natural phenomena. Various techniques are used to remove noise from digitized images.

Worapoj Peerawit et.al[19] indicated the use of Sobel detector in combination with edge density property to detect the edges helping to bifurcate noise and text resulting in removal of edge noise.

The other method is to identify the content area rather than noise recognition and removal. This is more an efficient method in Kannada language as even a dot in the right area of the character makes an high impact.

Methods proposed by F. Shafait, J. van Beusekom, D. Keysers, and T. M. Breuel [20] [21] are for effective methods which identify actual content by page frames. These methods find the page frame through alignment properties. The two step process involves a geometric model which is built for the page frame of a scanned document. Then second involves a geometric matching method used to find the globally optimal page frame with respect to a defined quality function. These are common methods in good practice, they require prior extraction of text lines and zones from the document images, the methods are considered slow. But for the curved type of characters in Kannada the method could be used more efficiently.

Jalal Uddin Mahmud et. al[22] noise is removed from character images. Noise removal includes removal of single pixel component and removal of stair case effect after scaling. Stair case effect occurs when the scaled characters have junctions so thin that inner and outer contour required for chain code representation cannot be found. Each pixel has been replaced by a filtering function to avoid such effect. However, this does not consider background noise and salt and pepper noise.

Md. Abul Hasnat et al.[23], used connected component information and eliminated the noise using statistical analysis for background noise removal. For other type of noise removal and smoothing they used wiener and median filters.

Tinku Acharya et al.[24], proposed that connected component information can be found using boundary finding method. Pixels are sampled only where the boundary probability is high. This method however needs elaboration if the character’s features change in the boundary.

VI. ALGORITHM FOR RECOGNITION OF BASIC KANNADA CHARACTERS

Main:

Step 1: Consider the Candidate image C1 (Fig 1).

Step 2: C1 is scaled to width of 60 pixels and height to 50 pixels

Step 3: A Grid is introduced with line width as 0.4 mm, Horizontal spacing of 11.6, Vertical spacing of 10.6, Horizontal offset of 0.8 and vertical offset of 0.8. (Fig 2).

Step 4: Consider the Lower most half of C1. (Fig 3)

Step 5: Each cell is preset with a weight starting from 51 to 58 for the first row and 61 to 68, 71 to 78 and 81 to 88 for the subsequent rows. Set Total weight W = 0.

Step 6: Each rectangle cell is considered as a window and the amount of pixels in black is calculated for all the windows as :Using Natural Ratio method [28] If there is a minimum difference of 50% between foreground and background the weight of cell is added to W. Repeat Step 6 from cell (5,1) to (8,8).

Step 7: Based on W the value is compared with the referral database to recognize the character as in step 8 to 18.

Step 8: If W is between 710 to 725 the character is either "ಅ" or "ಆ" Jump to subroutine 710

Step 9: If W is between 900 to 930 then Character is "E".

Step 10: If W is between 940 to 960 then Character is encoded as "ಎ" or"ಏ". If W is between 940 to 960 then jump to Subroutine 1000

Step 11: If W is between 1040 to 1050 then C is "ಈ"

Step 12: If W is between 1051 to 1070 C is "ಐ"

Step 13: If W is between 1200 to 1250 then C is either "N" or "O". Jump to Subroutine 1000.

Step 14: If W is between 1280 to 1299 C is " ಒ"

Step 15: If W is between 1300 to 1340 C is "ಊ ;

Step 16: If W is between 1340 to 1370 "ಉ'

Step 17: If W is between 1520 and 1550 C is "ಋ'.

Step 18: End

Subroutine 710:

Step 1: Consider the Candidate image C1;

Step 2: C1 is scaled to width of 60 pixels and height to 50 pixels

Step 3: A Grid is introduced with line width as 0.4 mm, Horizontal spacing of 11.6, Vertical spacing of 10.6, Horizontal offset of 0.8 and vertical offset of 0.8,

Step 4: Consider the top right of C1.

Step 5: Each cell is preset with a weight starting from 15 to 18 for the first row and 25 to 28, 35 to 38 and 45 to 48 for the subsequent rows.
Set Total weight W = 0.

Step 6: Using Natural Ratio method [28] If there is a minimum difference of 50% between foreground and background the weight of cell is added to W.

Step 7: Based on W the value is compared with database to find the character as in Step 7 to 10.

Step8: C1 character is either "ಅ" or "ಆ"

Step 9: The top right half is considered for calculation of W
If W is 290 to 320 the letter is "ಅ"
If W is 325 to 340 the letter is "ಆ"

Step 10: End.

Subroutine 1000:

Step 1: Consider the Candidate image C1;

Step 2: C1 is scaled to width of 60 pixels and height to 50 pixels

Step 3: A Grid is introduced with line width as 0.4 mm, Horizontal spacing of 11.6, Vertical spacing of 10.6, Horizontal offset of 0.8 and vertical offset of 0.8.

Step 4: Consider the top right of C1.

Step 5: Each cell is preset with a weight starting from 11 to 14 for the first row and 21 to 24, 31 to 34 and 41 to 44 for the subsequent rows.
Set Total weight W = 0.

Step 6: Using Natural Ratio method [28] If there is a minimum difference of 50% between foreground and background the weight of cell is added to W.

Step 7: Based on W the value is compared with database to find the character as in Step 8 to 10.

Step 8: C1 character is either "N" or "O" or "ಎ" or "ಏ".

Step 9: The top left half is considered for calculation of W
If W is 45 to 60 the letter is "ಎ"
If W is 95 to 110 the letter is "ಏ"
If W is 220 to 240 the letter is "N"
If W is 255 to 270 the letter is "O"

Step 10: End

Fig. 1: Candidate Character C1

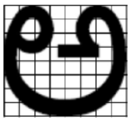

Fig. 2 : Character with 8 X 8 grid

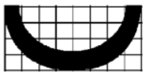

Fig. 3: Bottom 4 X 8 of C1

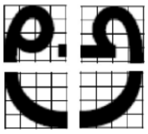

Fig. 4: C1 in 4X4 grid 4 structures

VII. CALCULATION OF WEIGHTS FOR ALL THE VOWELS

The weight of candidate character is estimated with a sample of eight different popular fonts of Kannada vowels. Considering each window's weight and on summation of all the individual cells' weight, the following estimate of weights is arrived at. It is also seen that the change in fonts have no/little effect on the weights as the dimension of the grid, scaling during preprocessing and the assignment of weights to different window in the grid nullify the little differences that could have on the weights.

The following indicates the character C1, window weights with 50% threshold and final weight 'W'.

ಅ 51 58 61 68 72 77 83 84 85 86
W= 725

ಆ 51 58 61 68 72 77 83 84 85 86
W= 725

ಇ 52 58 63 64 65 66 67 73 76 77 84
85 87 W= 917

ಈ 51 52 56 57 58 61 66 67 72 75 78
83 84 85 87 W= 1032

ಉ 51 53 54 55 56 58 61 63 64 65 66
68 71 73 76 78 82 83 86 87 W= 1350

ಊ 51 53 55 56 58 61 63 65 66 68 71
73 75 76 78 82 85 86 88 W= 1310

ಋ 51 52 53 55 58 61 62 63 65 66 68
71 73 75 76 78 81 82 84 85 86 87
W= 1532

ಎ 51 58 61 68 71 74 75 77 78 82 83
86 88 W= 951

ಏ 51 58 64 68 71 74 75 77 78 82 83
86 87 W= 954

ಐ 51 52 53 58 61 68 71 74 75 77 78
82 83 86 87 W= 1056

ಒ 51 52 53 54 61 68 71 72 74 75 77
78 82 83 84 85 86 87 W= 1293

ಓ 52 53 57 61 62 68 71 72 74 75 78
82 83 84 85 86 87 W= 1230

ಔ 52 53 57 61 62 68 71 72 74 75 78
82 83 84 85 86 87 W= 1230

VIII. CONCLUSION

The algorithms presented extract as minimum features as possible so as to reduce the computation and complexity involved. It has always been of prime importance in OCR to extract features effectively and decide the relevance of features extracted. The algorithm discussed acts in two fold and gets to additional features only when a conflict emerges. The conflict resolution again uses similar procedure with a different zone of the character extracting half the features extracted in the main procedure hence making the system of algorithms more effective and less time consuming. The paper also documents the weights of Kannada vowels so as to reveal the importance of the method. A total of 15 different Kannada fonts are considered to arrive at the range of weigh for each character. A good result of more than 95% recognition is achieved during the study. The character "ಋ though is considered as an vowel as per convention, in this study it is considered as combination of two characters, which assumption leads to effective recognition in further study and extension of these algorithms for all possible character and combinations possible.

REFERENCES

[1] R. M. Brown, T. H. Fay, and C. L. Walker, Handprinted Symbol Recognition System, Pattern Recognition, Vol. 21, No. 2, 1988, pp. 91-118
[2] L. Stringa, A New Set of Constraint-Free Character Recognition Grammars, IEEE, Transactions on PAMI, Vol. 12, No. 12, 1990, pp. 1210-1217.
[3] C. Y. Suen, C. Nadal, R. Legault, T. A. Mai, and L. Lam, Computer Recognition of Unconstrained Handwritten Numerals, Proceedings of the IEEE, Vol. 80, No. 70, July 1992, pp.1162-1180
[4] B. T. Mittchell and A. M. Gillies, A Model-Based Computer Vision System for Recognizing Handwritten ZIP Codes, Machine Vision and Applications, Vol. 21, No. 4,
1989, pp.231-243.
[5] L. Cun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, Vol. 86, No. 11, November 1998, pp.2278-2324.
[6] M. Shi, Y. Fujisawa, T. Wakbayashi, and F. Kimura, Handwritten Numeral Recognition Using Gradient and Curvature of Gray Scale Image, Patter Recognition, Vol. 35, No. 10, 2002, pp. 2051-2059.
[7] L.N. Teow and K. F. Loe, Robust Vision-Based Feature and Classification Schemes for Off-Line Handwritten Digit Recognition, Pattern Recognition, Vol. 35, No.1, 2002,
pp. 2355-2364.
[8] D. Decoste and B. Scholkopf, Training Invariant Support Vector Machines, Machine Learning, Vol. 46, No. 1-3, 2002, pp. 160-190.
[9] L. Xu, A. Krzyzak, and C. Y. Suen, Methods of Combining Multiple Classifiers and their Applications to Handwritten Recognition, IEEE Transactions on Systems, Man, Cybernetics, Vol. 22, No. 3, 1992, pp. 418-435.
[10] C. H. Lee and D. A. Langdgrebe, Feature Extraction Based on Decision Boundaries, IEEE Transaction on PAMI, Vol. 15, No. 4, April 1993, pp. 388-400.
[11] S. W. Lee, Off-Line Recognition of Totally Unconstrained Handwritten Numerals Using Multilayer Cluster Neural Network, IEEE Transactions on PAMI, Vol. 18, No. 6,
June 1996, pp. 648-652.
[12] P. D. Gader and M. A. Khabou, Automatic Feature Generation for Handwritten Digit Recognition, IEEE Transactions on PAMI, Vol. 18, No. 12, pp. 1256-1261.
[13] W. E. Weideman, M. T. Manry, H. C. Yau, and W. Gong, Comparisons of a Neural Network and a Nearest-Neighbor Classifier Via the Numeric Handprint Recognition.
[14] II-Seok Oh and C. Y. Suen, Distance Features for Neural Network-Based Recognition of Handwritten Characters, International Journal on Document Analysis and Recognition,
Vol. 1, No. 1, 1998, pp. 73-88.
[15] L. H. Yang, C. Y. Suen, T. D. Bui, and P. Zhang, Discrimination of Similar Handwritten Numerals Based on Invariant Curvature Features, Pattern Recognition, Vol. 38, No. 7, July 2005, pp. 947-963.
[16] J. Gao and X. Q. Ding, On Improvement of Feature Extraction Algorithms for Discriminative Pattern Classification, 15th International Conference on PatternRecognition, 2000, pp. 2101-2104.
[17] L. S. Oliveira, R. Sabourin, F. Bortolozzi, and C. Y. Suen, Impacts of Verification on a Numeral String Recognition System, Pattern Recognition Letters, Vol. 24, No. 7, July
2002, pp. 1023-1031.
[18] G. Y. Chen, T. D. Bui, and A. Krzyzak, Contour-Based Handwritten Numeral Recognition Using Multiwavelets and Neural Networks, Pattern Recognition, Vol. 36, No. 7, 2003, pp. 1997-1604
[19] Worapoj Peerawit and Asanee Kawtrakul, Marginal Noise Removal from Document Images Using Edge Density, ICEP2004, Phuket, Thailand, 2004.
[20] F. Shafait, J. van Beusekom, D. Keysers, and T. M. Breuel. Page frame detection for marginal noise removal from scanned documents. In SCIA 2007, Image Analysis, Proceedings, volume 4522 of Lecture Notes in Computer Science, pages 651–660, Aalborg, Denmark, June 2007.
[21] F. Shafait, J. van Beusekom, D. Keysers, and T. M. Breuel. Document cleanup using page frame detection. Int. Jour. on Document Analysis and Recognition, 11(2):81–96, 2008.
[22] Jalal Uddin Mahmud, Mohammed Feroz Raihan & Chowdhury Mofizur Rahman (2003). A Complete OCR System For Continuous Bangali Characters, IEEE,PP. 1372-1376.
[23] Md. Abul Hasnat, S.M. Murtoza Habib, Mumit Khan (2007). A High Performance Domain Specific OCR for Bangla Script.
[24] Tinku Acharya and Ajoy K. Ray (2005). " Image Processing Principles and Applications, John Wiley & Sons, Inc., Hoboken, New Jersey.
[25] K.S. Prasanna Kumar, Optical Character Recognition (OCR) for Kannada numerals using Left Bottom 1/4th segment minimum features extraction, Int.J.Computer Techology & Applications,Vol 3 (1), 221-225
[26] R.Narasimhacharya "History of Kannada Literature: readership lectures" 1988.
[27] "Kannada script Evolution". Official website of CIIL, India. Classicalkannada.org. Retrieved 12 May 2008.
[28] http://www.water.rutgers.edu/Senior_Design/Final%20Documents%20(Part%201)/Appendix%20E%20-%20Image%20Processing.pdf

K.S. Prasanna Kumar received his Bachelor of Science in Computer Science and Master of Computer Applications from Bangalore University, Bangalore, India and Mysore University, Mysore, India respectively.

He is currently working as Assistant Professor in the PG Department of Computer Science-MCA at Acharya Institute of Technology, Bangalore, India. Currently, he has submitted his Ph.D. thesis on "Analysis and Design of Minimal Feature Extraction Algorithms in Optical Character Recognition (OCR) for KANNADA Characters" and successfully completed his viva-voce in CMJ University, Shillong, Meghalaya, India, where his research focuses primarily in the areas of OCR for Kannada language.

During his educational career he has actively participated in research resulting in peer reviewed publications, including nine conference proceedings and four journal publications. He is a Life time member of the Indian Society for Technical Education (ISTE) and Computer Society of India(CSI).