IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

A Survey on Recent Advances to Read Handwritten Devanagari Script

Gauri Kolte[1*], Jennifer Fernandes[2], Prashant Vishwakarma[3], Samira Nikharge[4], Shalaka Deore[5]

[1,2,3,4,5]Dept. Computer, Modern Education Society's College of Engineering, Savitribai Phule Pune University, Pune, India

[*]*Corresponding Author: vprashant29@live.com, Tel.: +91-7276173229*

DOI: https://doi.org/10.26438/ijcse/v7i2.589595 | Available online at: www.ijcseonline.org

Accepted: 20/Feb/2018, Published: 28/Feb/2019

***Abstract*—** In the realm of advances in Processing Capabilities as well as Algorithms and their Efficiencies, transliteration mechanisms between Handwritten and Digital data namely called as Recognition Systems or Machine Reading Systems have been able to reach reliable precision. Devanagari and its variant scripts are widely used in the Indian Subcontinent. Being used by the second largest population in the world, it is practical to have research for Devanagari as well. While the current advances in recognition of Devanagari suggest requirement of more work and scope for accuracy levels, this survey aims to enlist the approaches taken in research to read handwritten Devanagari script. Citing works from different papers using different classifiers and techniques, it attempts to compare results and also imply the need of taking research forward. The survey contains methodologies followed in recent times, mentions data collection strategies or datasets available, uses classifiers and their recognition rates respectively.

***Keywords*—** Devanagari, Recognition, Segmentation, Pre-processing, Classification

I. INTRODUCTION

India is home to the third most spoken language in the word; Hindi. It is based on Devanagari Script which also serves as a base script to various other languages in the Indian subcontinent like Gujarati, Punjabi and Marathi hence Devanagari is the most widely used script in India. It is used in 120 languages making it the most adopted writing systems. It originates from the Brahmi script with the improvement in computational power and a considerable growth in data sources, there has been scope for newer methods to recognize handwritten scripts in different languages. In the computational genre, introduction of GPUs and cluster architecture has made substantial improvement and increased the scope for performance demanding procedures. The scale of available data has increased considerably in recent times. This gives an opportunity to develop language processing models for Devanagari, as it is a compound script based language.

Character Recognition deals with recognizing various kinds of handwritten/printed characters such as digits, cursive scripts, symbols. We keep our focus of research on Devanagari. Devanagari has evolved over a period of two thousand years and it is highly cursive and continuous in nature. Machine driven reading of a script requires the input to go through different phases in a specific order, broadly classified as Pre-processing, Segmentation of Text, Feature extraction and Recognition using Classification.

Several methods are available for implementing a recognition system. Each method specifies its own way so as to improve the accuracy. In this survey paper, we strive to offer an updated view of this topic. We give an overview of the existing recognition systems for Devanagari. Furthermore, the paper introduces new challenges that have to be addressed.

The remainder of this paper is structured as follows; section II specifies the various features of Devanagari script, section III explains the various phases for Devanagari Recognition. Section IV discusses performance of stated methods while the last section concludes and highlights comparably better performing procedures for the same.

II. FEATURES OF DEVANAGARI SCRIPT

Devanagari script represents phonetic sounds in a compound manner. A character is combination of a consonant (Vyanjan) and a vowel (Swara). A vowel can be an individual character or be added to a consonant in the form of Modifier. There are mainly 11 independent vowels in Devanagari:

अ आ इ ई उ ऊ ऋ ए ऐ ओ औ

Figure *1. Vowels (Swara)*

Considering 12 major vowels, their permutations with consonants is called Barakhadi for the particular consonant.

The following table demonstrates major vowels used in both their independent and modifier forms individually as well as added to a consonant क. With each modifier, middle character remains the same. Only the upper and lower zones are altered.

Table 1. Modifiers with Consonants

Independent	Dependent	With क
अ	(none)	क
आ	□	□□
इ	□	□□
ई	□	□□
उ	□	□□
ऊ	□	□□
ऋ	□	□□
ए	□	□□
ऐ	□	□□
ओ	□	□□
औ	□	□□

Vowels therefore have an independent form as well as its modifier form (Matra). Most characters distinctly have a baseline drawn above them. This is known as the Shirorekha. Being common to all letters, it is not the most eligible feature during recognition phase but can greatly help to distinctly identify individual words. Each word in the script has its common continuous Shirorekha. Long discontinuity signifies completion of a word. A character in Devanagari has three zones of strokes; middle (main character), above Shirorekha and below middle character (Matra).

Figure *2. Example of Devanagari word*

Another frequent modifier form is the nasalization of a consonant. It is done by introducing two more character namely Bindu "□" and Chandrabindu "□". It is possible to nasalize all vowels in Hindi except ऋ. These do not carry an independent form but add to upper zone of any character.

अँ आँ इँ ईं उँ ऊँ एँ ऐं ओं औं
कँ काँ किं कीं कुँ कूँ कें कैं कों कौं

Figure *3. Vowels*

Devanagari unlike English does not have capitalization of characters.

Table 2. Vowels, Modifiers and Consonants

Vowels	अ	आ	इ	ई	उ	ऊ	ऋ	ए	ऐ	ओ
	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
	ऑ	औ	□□	□□	□□					
	[11]	[12]	[13]	[14]	[15]					
Consonants	क	ख	ग	घ	ङ	च	छ	ज	झ	ञ
	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
	ट	ठ	ड	ढ	ण	त	थ	द	ध	न
	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]	[20]
	प	फ	ब	भ	म	य	र	ल	व	श
	[21]	[22]	[23]	[24]	[25]	[26]	[27]	[28]	[29]	[30]
	ष	स	ह							
	[31]	[32]	[33]							

Being based on phonetic sound representations, Devanagari allows writing short continuation of a consonant as a half character added to another character.

व्यास

Figure *4. Example of a word with half character*

Here, व is used as a half conjunct to the next character in word (या).

Structural features of Devanagari vary according to different styles of writing. Though the standard script distinctly has a baseline (Shirorekha), any consonant is distinctly present in middle zone except cases with half consonant conjuncts. There can only be one vowel added to a consonant. Therefore, modifier to a consonant would

either be in upper zone or lower zone but not in both at a time.

III. RECOGNITION OF DEVANAGARI CHARACTERS

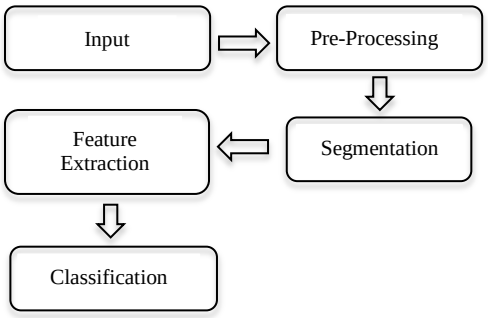

Figure 5. *Block diagram of Devanagari Text Recognition*

1. Data Generation/Collection

There are two types of data generation techniques, Online and Offline. With the rapid development of technologies in touchscreen-based and pen-enabled mobile devices, the online method is becoming more and more popular for data generation. On the other hand offline method includes scanned documents. The Online method is the preferred choice of researchers as it gives the advantage of understanding the natural aspect of writing in terms of speed, angle, and pressure and so on.

The dataset is divided into two sets i.e. Training Set and Testing Set. The Training set is used to teach the machine about the features of different characters which it has to recognize in the future. After the training is done the testing set is used to check whether the characters are accurately recognized by the classifier. The more the classifier is trained; higher are the chances of identifying the characters appropriately.

The experiment is carried out on 1000 characters in [1]. The training set consists of 904 characters and the testing set contains 204 characters. In [2] the data set is generated by collecting samples from different individuals. Samples of 5000 characters are considered. The printed dataset is formed of ISM fonts, in which font size 16 is used and different font styles. Character image set of size 430 is taken for implementation in [3]. Of which 140 characters are used for training and rest 290 images of characters are taken as test data for classification using SVM. In [4] the dataset is prepared using a HP Tablet PC TC1000. For training, samples were collected from 20 writers, providing 7 instances of each character of the Devanagari script. Dataset for the testing phase was prepared separately.

In [5], 40 people of different age group and field were told to write 10 samples of each character on a sheet which was then scanned and stored in bmp format. Only 20 handwritten characters were used which resulted to 2000 characters by 100 people in [6]. Databases in [8] consist of various languages with large datasets like Devanagari (109 individuals comprising of 23,891 samples). [9] Specifies a survey that experiments on the Indian Language Technologies (ILT) dataset. The method proposed in [11] uses ICDAR 2013 handwriting segmentation contest databases. ICDAR 2013 includes Latin-based and Indian scripts.

2. Pre-processing

The data collected over various sources is prone to inconsistencies and errors. Human input and sensorial data both involve some degree of entropy. This may include missing values, incorrect readings or even irrelevant attributes.

This step involves tasks and operations applied on the dataset to prepare it for specific computations that may require a standard input. Some common procedures that are required include:

- Binarization: In case of image based input, Binarization reduces image to a matrix format. This may also include threshold to reduce dataset into relevant portion only.
- Sampling: It is to take a subset of large data to partially build a model and evaluate feasibility of system. Also used to divide the system implementation through multiple iterations.
- Noise Reduction: Noise in data may incur from inaccurate values as well as outliers. Various Filtering techniques use utilized to reduce such entries.
- Thinning and Smoothing: Stroke edges must be distinct in the input to achieve higher accuracy. These techniques reduce variations in the image.
- Normalization: To increase readability of features, normalization transforms the data values to a similar range. In Image Processing, it can increase pixel intensities.

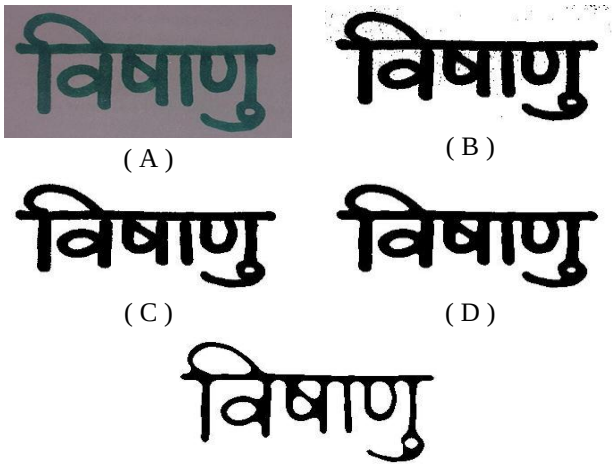

Figure 6. *A) Input Image B) Binarization & Thresh-holding C) Noise Removal D) Smoothing E) Thinning*

In [1],[7] pre-processing involves tasks such as; Noise Removal, Binarization, Edge Detection, Dilation & Filtering and Process image for feature extraction. The Binarization process helps convert the grey-scale into its binary image using thresholding technique. Also Edge Detection of binary image is possible using canny technique. Along with Dilation these are two important steps to convert the scanned image into usable format for feature extraction. In [3] the image acquired as input is processed in two steps, first Binarization is performed followed by Removal of horizontal bar and Separation of characters. The scanned image is binarized and dilated both horizontally and vertically and then the horizontal bar is removed. Sampling and Resampling are common procedures followed by [4]. It also implements filters such as low pass filter to highlight features which might not be distinct due to noise present. Additionally, [4] uses a 5-tap Gaussian Filter on both x and y coordinate sequences exclusively and preprocessing is carried out on the input image after removal of special strokes as part of structural recognition. [9] Uses OCR to read scanned images for text, hence it loosely performs Binarization, noise removal and skew correction. For instance, [10] proposed a method to recognize Barakhadi characters in which image is preprocessed by applying a 3x3 averaging filter before Binarization because it exhibits undesirable effects, and then Normalization is used to fit the image into specific size followed by Thinning. In this paper, baseline (Shirorekha) is detected (using Hough transformation) is removed to divide the image into two regions and each region is further processed separately as individual entities to be recognized.

3. Segmentation

Figure *7. Example of Segmented Word*

As the name suggests segmentation is nothing but dividing the word into different characters. It is an important factor which helps to decide the accuracy of the classification process. Segmentation process can be categorized into four types:(1)*Dissection Methods* which is termed as a classical approach that consists of methods which are used to partition the input image into sub images, (2) *Implicit Segmentation* considers the image as whole and classifies the subset of spatial features collected and *Explicit Segmentation* performs classification according to predefined windows, (3) integration of segmentation and recognition gives rise to the *recognition-based* methods, (4) *Holistic Approach* avoids segmentation while considering the entire word as a unit.

The whole image consisting of a sequence of characters is split into a set of sub-images in [1], one for each character. The segmentation is performed by assigning a unique number to each isolated character using a labeling technique. This labeling helps to identify the total number of characters present in the image. Each of the individual characters is also uniformly resized to 90*60 pixels for classification. In [3] the segmenting of characters is performed under the preprocessing phase where in the horizontal bar is detected first and then accordingly the different characters are separated. The approach in [4] uses structural recognition in which the input test pattern is first preprocessed to extract structural features at stroke level such as mean (x, y) value, length, offline features and directional codes. Using this, Shirorekha is first detected followed by detection of vowel modifiers such as Anuswara, Visarga, Chandrabindu and Ardhachandra and then separated/removed from the character. [9] Performs segmentation.

[11] Proposes an optimal segmentation method for handwritten documents. Segmentation of handwritten documents is considered challenging due to irregular spacing between words and variations in writing styles. This method provides a way to classify between intra/inter word gaps. In this approach the word segmentation is formulated as an optimization problem which maximizes the similarity between inter-word gaps and dissimilarities between inter-word and intra-word gaps. In this approach just like conventional approach, the documents will be first segmented into text-lines which would be represented as super pixels corresponding to a letter or group of letters. For a super pixel, the gaps are to be labeled according to inter or intra gaps, hence this approach presents a normalized super pixel extraction method that extracts a set of candidate gaps in each text-line. The assignment problem is formulated as a binary quadratic problem which allows considering pairwise relations and local properties for gaps. A super pixel can be represented using connected components but this approach may miss some candidates due to cursive writings, this works fine for inter word gaps but not suitable for intra word gaps hence a normalized connected components representation is adopted which estimates the average stroke width W in a document so that they have normalized sizes in terms of W.

$$\hat{y} = \boldsymbol{arg\ max}\big(\boldsymbol{w}, \boldsymbol{\psi}(\boldsymbol{x}, \boldsymbol{y})\big) \quad (1)$$

Equation (1) is a Function to Evaluate Likelihood of being a Word Separator or not. It is designed that would reflect the pairwise correlations and specify a unary property of the individual being a word separator or not. Observations made for labeling are:

- Inter word gap should have similar features
- The features of inter word and intra word gap should be different

4. Feature Extraction

In order to map an input to an output class, both must exhibit a similar set of features. Challenge is to select important and distinct features that uniquely identify the input. The process depends on nature of classifier. Some classifiers like CNN do not require explicit specification while Decision Trees may require distinct label attributes.

The diagonal-feature extraction scheme is used in [1] for offline character recognition. Each individual image of size 90*60 pixels is divided into 54 zones of 10*10 pixels each. Features are extracted from every pixel of the zone by moving in diagonal direction in the zone. Thus 19 sub-features are extracted from every zone and these values are further averaged to give a single feature value for each zone. This procedure is repeated for all the zones. Zones having their diagonal values as empty are denoted with 'zero' value. In this way every character is assigned with 54 features in all. In order to extract the gradient feature [2] makes use of two operators separately i.e. Robert operator and Sobel operator, which forms two separate gradient vectors, which are used to get gradient at each image pixel separately. [3] Uses K-means clustering for feature extraction. In case of low illumination K-means gives best results. Its implementation is dependent on center points and data values that are assigned to respective center points depending on their distance forming clusters. The input image is first resized into 70*50 pixels and then clustering is performed to generate 35 clusters in all.

Directional vector is used in [8] and based on the sequence of preprocessing two feature extraction methods Chain-code histogram feature (height of 100) and Point-float feature (height of 10) are used and further applied with different classifiers. Moments like Geometric moment, Central moments, Scale invariant moments and Rotational moments in [10] are used for Devanagari handwritten character recognition as they are robust to noise of high frequency. Moment features are extracted from different regions(top and bottom) to detect Matra and lower modifiers and send to classifier for recognition.[7] uses three major types of feature extraction methods like (1)Global Transformation and Series Expansion, (2)Statistical Features and (3)Geometrical and Topological Feature. Due to the advantage of Wavelet transform of multi resolution analysis, it decomposes image into several levels of resolution. It reduces ambiguity, improves accuracy, practical applications and fast computations. In [6] this paper 2D input image wavelet filtering is done, first on row index and then column index. [9] Uses GSC and Gradient feature set. [11] Constructs a feature map $\psi(x, y)$ in order to extract the features that classify between the inner and the intra gap, the following features are adopted:

- Normalized distances between neighboring super pixels.
- Projection profile features.
- Width ratio between current gap and the largest gap.

5. Classification

It is the process of mapping the features from the previous stage to the characters in the data set to find a proper match. For this various classifiers are used such as: Support Vector Machine (SVM), Artificial Neural Networks (ANN), Template Matching, Statistical Techniques, and Genetic Algorithms etc. The method of implementation differs with every classifier.

The features extracted in [1] in the previous step are further classified using genetic algorithm technique. These features are used to form chromosomes which are used for recognizing the label to which the character belongs. The features extracted from each zone is converted into 7 bit string , there are 54 zones in each character image so, 54x7=378 bit string is used to represent each character image. This phase consists of two functions; Chromosome function generation and Chromosome fitness function. SVM is used for classification in [2] and [3]. It is a type of supervised learning which makes use of a hyper plane as a decision factor. An optimal hyper plane is one which has maximum margin of separation of hyper plane and closest point from the dataset. A linear kernel is used for implementation. The training data set consists of handwritten characters which are stored in the form of feature vectors. In [4] Subspace method is used for classification. For classification and recognition ANN is used in [6], Back Propagation algorithm is used to train and test the Neural Networks. [7] Gives the different types of classification techniques like Template matching, Statistical techniques, Neural Networks, SVM Classifier and Combination classifier. [9] Specifies the use of Neural network using Gradient Features, has 512 input nodes, 128 hidden nodes and 40 output nodes. It uses another neural network classifier with GSC features. The third classification algorithm it uses is K- nearest neighbour that uses GSC feature set. [11] Uses Structured SVM for classification.

6. Post Processing

For improving results over traditional recognition strategy, in case of meaningful input, post processing can be a solution to correct minimal errors in the script. Any meaningful input from a language works on multiple aspects such as lexical, syntactic and semantic contexts.

A recognition system can considerably improve output accuracy if it utilizes a module to check context of the input. Some features of the script that might be removed during preprocessing can thus be corrected by post processing. It can also induce some new errors in result if strict checking or incorrect additions are made. Post processing therefore is an

optional step in recognition used in two ways mainly, Context Checking and Dictionary Checking.

IV. RESULTS AND DISCUSSION

The recognition system in [1], [3] is implemented using MATLAB 7.10.0. It takes a scanned image as input using a scanner, digital camera or any other suitable digital input device. Also the image should have a specific format .jpeg, .bmp etc. The precision of the offline Devanagari system is 85.78% match, 13.35% mismatch. [2] Makes use of Sobel (3*3) and Robert (2*2) operator separately for L = 8, 12, 16, 32 dimensions for extracting gradient features respectively. The following Table 1 & Table 2 shows the results on both datasets using SVM classifier. [3] Shows that results using Support Vector Machine are better than results using Euclidean distance. Characters with good performance i.e. which has more than 75% performance percentage, are 25 using Euclidean distance method. Using SVM, performance of all characters is better than 75% except one character and that is /k. Computation overhead in Support Vector Machine is less than Euclidean distance approach. So classification using SVM is better than classification using Euclidean distance approach. In [4] the accuracy of the FREQ set (set of frequently occurring characters) was of the order of 93%. The accuracy of the RAND set (set of randomly selected characters) varied between 82% and 85% depending on the presence of the *Shirorekha*. Hence the presence of the *Shirorekha* as an additional stroke tends to reduce the accuracy of classification. Using Neural Network with Wavelet Feature [6] gave an accuracy of only 70% over 2000 samples of only 20 characters. The Neural network classifier with Gradient features gives an overall accuracy of 94.36% for an isolated consonant, the neural network classifier with GSC feature set gives an accuracy of 95.25% and the k-Nearest neighbour gives an accuracy of 94.27% for isolated consonant images [9]. The word segmentation method based on SVM provides with an overall accuracy of 92.82% [11]. The accuracy is an average of results acquired after testing performed on two different datasets.

Table 3. Result Using Sobel Operator Showing % of Accuracy

L =	8	12	16	32
Handwritten Dataset (Char)	94%	94.76%	96%	97%
Printed ISM Dataset (Char)	98%	98%	98.45%	98.78%

Table 4. Result Using Robert Operator Showing % of Accuracy

L =	8	12	16	32
Handwritten Dataset (Char)	94.45%	95.06%	95.67%	96.09%
Printed ISM Dataset (Char)	97%	98%	98.02%	98.0%

V. CONCLUSION

The survey carried out in this paper helps us in understanding the various methods used for Devanagari script recognition. The different strategies that use a combination of multiple features, classifiers and templates have been considered extensively and each combination has the ability to give distinct results. Devanagari script recognition finds its application in fields like, postcode reader, financial document reader, and form processing and signature verification. Also as India is well known for its vast historical background, the historical documents and books that provide its proof are of great importance. These books are basically handwritten or printed in Devanagari script. As we know that the Devanagari script is quite extensive, typing these documents to transliterate is highly tedious as compared to digitalizing it for better access, sharing etc. In this way these historical books can be made available to various research communities in India in the area of social science, linguistics and economics.

From the current study we can infer that a lot of research has been carried out in the field of recognition and the rates obtained in character recognition are also very acceptable. However the accuracy rate for word recognition is much less as compared to that for character. Seeing this as an opportunity, efforts can be taken to achieve reliable accuracy in word recognition as there is in character.

REFERENCES

[1] V.P. Agnihotri, "*Offline Handwritten Devanagari Script Recognition*", International Journal Information Technology and Computer Science, pp. 37-42, 2012.

[2] A.N. Holambe, Dr. R.C.Thool, Dr. C.M.Jagade, "*Printed and Handwritten Character & Number Recognition of Devanagari Script using Gradient Features*", International Journal of Computer Applications, Vol.2, Issue 9, pp. 0975-8887, 2010.

[3] A.Gaur, S.Yadav, "*Handwritten Hindi Character Recognition using K-Means Clustering and SVM*", In the Proceedings of the 2015 IEEE 4th Symposium on Emerging Trends and Technologies in Libraries and Information Sciences, 2015.

[4] Niranjan Joshi, G.Sita, A.G. Ramakrishnan, Deepu V., Sriganesh Madhvanath, "*Machine Recognition of Online Handwritten Devanagari Characters*", In the Proceedings of the 2005 8th International Conference on Document Analysis and Recognition (ICDAR'05), IEEE, 2005.

[5] P.B. Khanale, S.D. Chitnis, "*Handwritten Devanagari Character Recognition using Artificial Neural Network*", Journal of Artificial Intelligence 4 (1), 2011.
[6] A. Dixit, A. Navghane, Y. Dandawate, "*Handwritten Devanagari Character Recognition using Wavelet Based Feature Extraction and Classification Scheme*", In the Proceedings of 2014 IEEE Indian Conference (INDICON), 2014
[7] V.J. Dongre, V.M. Mankar, "*A review of Research on Devanagari Character Recognition*", International Journal of Computer Applications, Vol.12, 2010.
[8] T. Mondal, U. Bhattacharya, S.K. Parui, K. Das, "*Online Handwritten Recognition of Indian Scripts - the first benchmark*", In the Proceedings of the IEEE 12th International Conference on Frontiers in Handwriting Recognition, 2010 .
[9] R. Jayadevan, S.R. Kolhe, P.M. Patil, U. Pal, "*Offline Recognition of Devanagari Script: A Survey*", IEEE Transactions on Systems, Man and Cybernetics- Part C: Applications and Reviews, Vol. 41, 2011.
[10] Shreya N. Patankar, Leena N. Ragha, "*Zonal Moments Based Handwritten Marathi Barakhadi Recognition*", International Journal of Engineering Research and Technology (IJERT), Vol. 1, Issue 6, August 2012.
[11] J. Ryu, H.I. Koo, N.I. Cho, *"Word Segmentation Method for Handwritten Documents based on Structured Learning",* IEEE Signal Processing Letters, Vol. 22, 2015.
[12] Shalaka Deore, Leena Ragha, *"Moment Based Online and Offline Handwritten Character Recognition",* CiiT International Journal of Biometric and Bioinformatics, Vol. 3, March 2011, ISSN: 0974-9675.

Authors Profile

Ms. Gauri S Kolte is pursuing Bachelor of Engineering in Computers from Modern Education Society's College of Engineering, Pune.

Ms. Jennifer Fernandes is pursuing Bachelor of Engineering in Computers from Modern Education Society's College of Engineering, Pune.

Mr. Prashant Vishwakarma is pursuing Bachelor of Engineering in Computers from Modern Education Society's College of Engineering, Pune.

Ms. Samira M. Nikharge is pursuing Bachelor of Engineering in Computers from Modern Education Society's College of Engineering, Pune.

Mrs. Shalaka P. Deore is currently working as a Professor in Modern Education Society's College of Engineering, Pune with an experience of 20yrs. Her areas of specialization are; Pattern Recognition, Image Processing, Neural Networks. She is also a member of the IETE.