

# USE OF JPEG ALGORITHM IN HANDWRITTEN DEVNAGRI NUMERAL RECOGNITION

Gajanan Birajdar<sup>1</sup> and Mansi Subhedar<sup>2</sup>

Department of Electronics and Telecommunication Engineering, SIES Graduate School of Technology, Navi Mumbai, India

gajanan123@gmail.com<sup>1</sup>, mansi\_subhedar@rediffmail.com<sup>2</sup>

## **ABSTRACT**

*Automatic recognition of handwritten Devnagri numerals is a difficult task in pattern recognition. It has numerous applications including those in postal sorting and bank cheque processing. In this paper, a fast and effective method is proposed for recognition of isolated handwritten Devnagri numerals based on JPEG image compression algorithm, which is less time consuming as compared to artificial neural network based recognition systems. In JPEG image compression, which has high compression ratio, a unique vector which helps to identify each character is generated. By using this unique vector, the proposed system has recognized the input numeral after measuring the Euclidean distance between the vector corresponding to the numeral and the vectors in the codebook as well as the length of vector. Then the shortest distance pointed to the corresponding letter is obtained. The result was considerably high in terms of recognition rate.*

## **KEYWORDS**

*Devnagri numerals, JPEG, offline Handwritten character recognition*

## **1. INTRODUCTION**

Optical Character Recognition (OCR) is a process of automatic recognition of different characters from a document image. OCR systems are considered as a branch of artificial intelligence and a branch of computer vision as well. Researchers classify OCR problem into two domains. One deals with the image of the character by scanning which is called Off-line recognition. The other has different input way, where the writer writes directly to the system using, for example, light pen as a tool of input. This is called On-line recognition. Fig. 1 shows the block diagram of the typical OCR system. The online problem is usually easier than the offline problem since more information is available, like the movement of the pen may be used as a feature of the character [1]. These two domains (offline & online) can be further divided into two areas according to the character itself: the recognition of machine printed data and the recognition of handwritten data. Machine printed characters are uniform in size, position and pitch for any given font. In contrast, handwritten characters are non-uniform; they can be written in many different styles and sizes by different writers and at different times even by the same writer. The OCR system based on three main stages: preprocessing, feature extraction and discrimination (also called classifier or recognition engine).

Recognizing handwritten numerals is an important area of research because of its various application potentials. Automating bank cheque processing, postal mail sorting, and job application form sorting, automatic scoring of tests containing multiple choice questions and other applications where numeral recognition is necessary. Character recognition engine for any script is always a challenging problem mainly because of the enormous variability in

handwriting styles. A recognition system must be therefore robust in performance so that it may cope with the large variations arising due to different writing habits of different individuals.

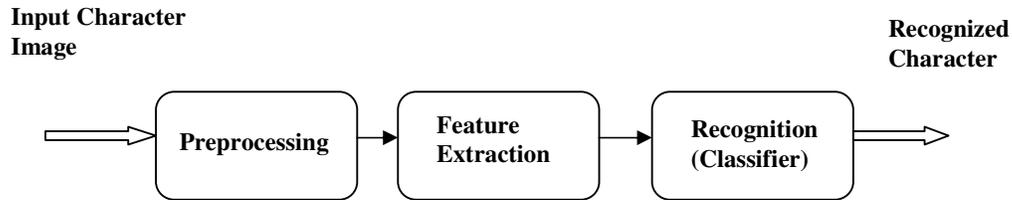


Figure 1: Typical OCR block diagram [2]

Traditional OCR systems are suffering from two main problems, one comes from feature extraction stage and the other comes from classifier (recognition stage). Feature extraction stage is responsible for extracting features from the image and passing them as global or local information to the next stage in order to help the later taking decision and recognizing the character. Two challenges are faced: if feature extractor extracts many features in order to offer enough information for classifier, this means many computations as well as more complex algorithms are needed. Thus long processor time will be consumed. On the other hand, if few features are extracted in order to speed up the process, insufficient information may be passed to classifier. The second main problem that classifier is responsible for, is that most of classifiers are based on Artificial Neural Networks (ANNs). However, to improve the intelligence of these ANNs, huge iterations, complex computations and learning algorithms are needed, which also lead to consume the processor time. Therefore, if the recognition accuracy is improved, the consumed time will increase and vice versa.

To tackle these problems, a new OCR construction is not proposed in this paper, where features extractor nor is ANN needed. The proposed construction relies on the image compression technique (JPEG). Taking advantages of the compressor, that it compresses the image by encoding only the main details and quantizes or truncates the remaining details (redundancy) to zero. Then generates a unique vector (code) corresponding to the entire image. This vector can be effectively used to recognize the character since it carries the main details of the character's image. The importance of the main details is that they are common amongst the same character which is written by different writers.

## 2. RELATED WORK

In this paper, Devanagiri numeral recognition algorithm is proposed based on JPEG image compression algorithm. The aim of handwritten numeral recognition (HNR) system is to classify input numeral as one of K classes. Over the years, considerable amount of work has been carried out in the area of HNR. Various methods have been proposed in the literature for classification of handwritten numerals. These include Hough transformations, histogram methods, principal component analysis, and support vector machines, nearest neighbour techniques, neural computing and fuzzy based approaches [3]-[4]. A study on different pattern recognition methods are given in [5]-[6]. In comparison with HNR systems of various non Indian scripts [e.g. Roman, Arabic, and Chinese], we find that the recognition of handwritten numerals for Indian scripts is still a challenging task and there is spurt for work to be done in this area. Few works related to recognition of handwritten numerals of Indic scripts can be found in the literature [7]-[10]. A brief review of work done in recognition of handwritten numerals written in Devanagiri script is given below:

Many schemes for digit classification have been reported in literature. They mostly differ in feature extraction schemes and classification strategies (Govindan & Shivaprasad 1990; Trier et al 1996) [11]. Features used for recognition tasks include topological features, mathematical moments etc. Classification schemes applied include nearest neighbour schemes and feed forward networks. In order to make their systems robust against variations in numeral shapes, researchers have also used deformable models, multiple algorithms and learning. A survey of the techniques is provided by Amin (1997) [12] and Plamondon & Srihari (2000) [13], Lam & Suen (1986) [14] have used a fast structural classifier and a relaxation-based scheme which uses deformation for matching.

A knowledge-based system using multiple experts has been used by Mai & Suen (1990) [15]. Kimura & Sridhar (1991) [16] developed a statistical classification technique that utilized profiles and histograms of the direction vectors derived from the contours. Chen & Lieh (1990) [17] proposed a two layer random graph based scheme which used components and strokes as primitives. Jain & Zongkar (1997) [18] have proposed a recognition scheme using deformable templates. LeCun et al (1989) [19] suggested a novel back propagation based neural network architecture for handwritten zip code recognition. Knerr et al (1992) [20] suggested the use of neural network classifiers with single layer training for recognition of handwritten numerals. Wang & Jean (1993) [21] suggested use of neural networks for resolving confusion between similar looking characters. Among studies on Indian scripts, notable work has been done on recognition of printed Devanagari characters by Sinha and others (Sinha & Mahabala 1979[22]; Bansal & Sinha 2001) [23]. They also suggested contextual post processing for Devanagari character recognition and text understanding. For handwritten Bengali character recognition, Dutta & Chaudhury (1993) [24] presented a curvature feature based approach. Chaudhuri & Pal (1998) [25] presented a complete Bangla OCR system.

### 3. DATA SET CHARACTERISTICS:

Devanagari script, originally developed to write Sanskrit, has descended from the Brahmi script sometime around the 11<sup>th</sup> century AD. It is adapted to write many Indic languages like Marathi, Mundari, Nepali, Konkani, Hindi and Sanskrit itself. Marathi is an Indo-Aryan language spoken by about 71 million people mainly in the Indian state of Maharashtra and neighbouring states. Since 1950 Marathi has been written with the Devanagari alphabet. Figure 2 below presents a listing of the symbols used in Marathi for the numbers from zero to nine.

०	१	२	३	४	५	६	७	८	९
0	1	2	3	4	5	6	7	8	9

Figure 2: Numerals 0 to 9 in Devanagari script

The dataset of Marathi handwritten numerals 0 to 9 is created by collecting the handwritten documents from writers. Data collection is done on a sheet specially designed for data collection. Writers from different professions and age groups were chosen and were asked to write the numerals. A sample sheet of handwritten numerals is shown in figure 3.

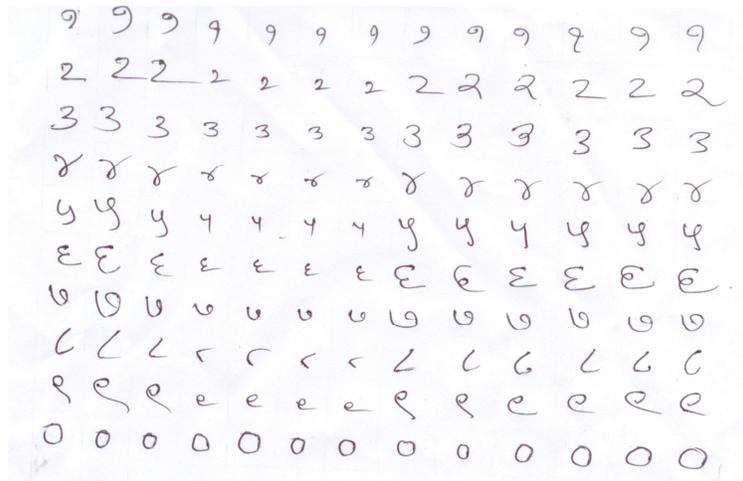


Figure 3: Sample sheet of handwritten numerals

The collected data sheets were scanned using a flat bed scanner at a resolution of 300 dpi and stored as colour images. The raw input of the digitizer typically contains noise due to erratic hand movements and inaccuracies in digitization of the actual input. To bring uniformity among the numerals, the cropped numeral image is size normalized to fit into a size of 60x60 pixels. A total of 400 binary images representing Marathi handwritten numerals are obtained from 20 different subjects.

#### 4. JPEG COMPRESSION TECHNIQUE:

JPEG may be adjusted to produce very small compressed images that are of relatively poor quality in appearance but still suitable for many applications. Conversely, JPEG is capable of producing very high quality compressed images that are still far smaller than the original uncompressed data. JPEG is also different in that it is primarily a lossy method of compression. Most popular image format compression schemes such as RLE, LZW or the CCITT standards are lossless compression methods. That is, they do not discard any data during the encoding process. An image compressed using a lossless method is guaranteed to be identical to the original image when uncompressed.

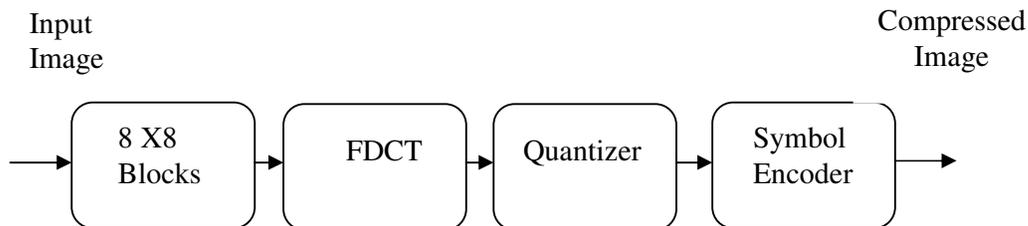


Figure 4(a): JPEG Encoder

Lossy schemes, on the other hand, throw useless data away during encoding. This is in fact, how lossy schemes manage to obtain superior compression ratios over most lossless schemes. JPEG was designed specifically to discard information that the human eye cannot easily see. Slight changes in color are not perceived well by the human eye while slight changes in intensity (light and dark) are. Therefore JPEG's lossy encoding tends to be more frugal with the gray scale part of an image and to be more frivolous with the color.

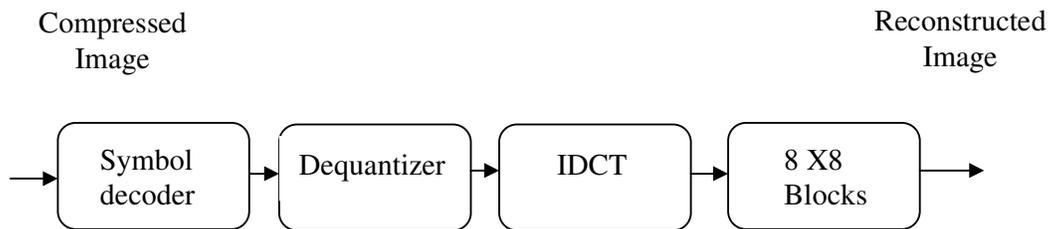


Figure 4(b): JPEG Decoder

In the JPEG baseline coding system, which is based on the discrete cosine transform (DCT) and is adequate for most compression applications, the input and output images are limited to 8 bits, while the quantized DCT coefficient values are restricted to 11 bits. The human vision system has some specific limitations which JPEG takes advantage of, to achieve high rates of compression.

As can be seen in the simplified block diagram of Figure 4, the compression itself is performed in four sequential steps: 8x8 sub-image extraction, DCT computation, quantization, and variable-length code assignment i.e. by using symbol encoder.

The JPEG compression scheme is divided into the following stages:

1. Transform the image into an optimal colour space.
2. Downsample chrominance components by averaging groups of pixels together.
3. Apply a Discrete Cosine Transform (DCT) to blocks of pixels, thus removing redundant image data.
4. Quantize each block of DCT coefficients using weighting functions optimized for the human eye.
5. Encode the resulting coefficients (image data) using a Huffman variable word-length algorithm to remove redundancies in the coefficients.

Since we do not concern in this work about the reconstruction part, the only part of compression is used (dashed box) and the vector will be tapped immediately after quantization stage.

## 5. PROPOSED ALGORITHM

Figure 5 illustrates the sequence of the proposed algorithm's steps based on reference [3]. After the character's image is scanned in the system the JPEG approximation will produce a vector. This vector is assumed to uniquely represent input image since it carries the important details of that image. Figure 6 shows a sample for Devnagri numeral 0. Then Euclidean distance between this vector and each vector in codebook will be measured. Finally, the minimum distance points to the corresponding character, and then the character is recognized. To obtain higher recognition accuracy additional data of length of the vector produced is also used in recognition process.

### 5.1 System Components:

The two main components are the code of the compression stage: 1. JPEG compressor and 2. The codebook as shown in Figure 6. JPEG compressor produces the vector which is assumed to uniquely represent input image since it carries the important details of that image. The code book is obtained by taking average of each group of Devnagri numeral. Code book design procedure is explained in following section.

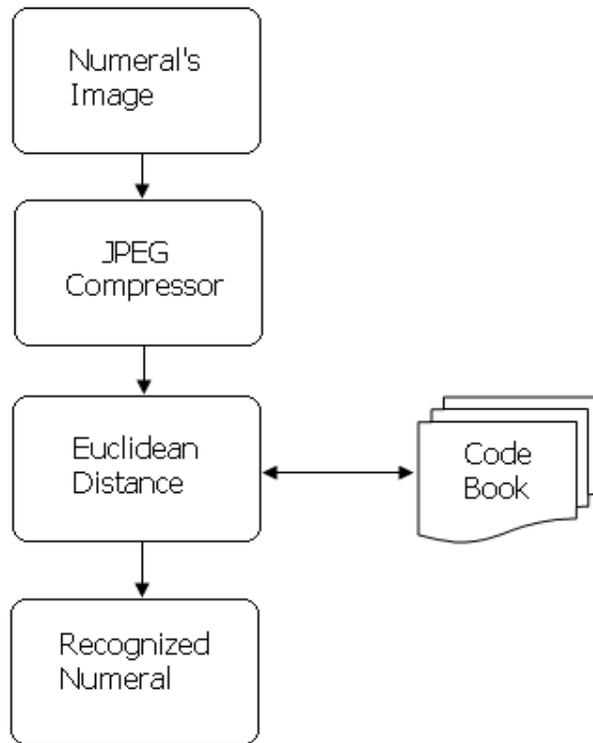


Figure 5: Flowchart of the proposed algorithm

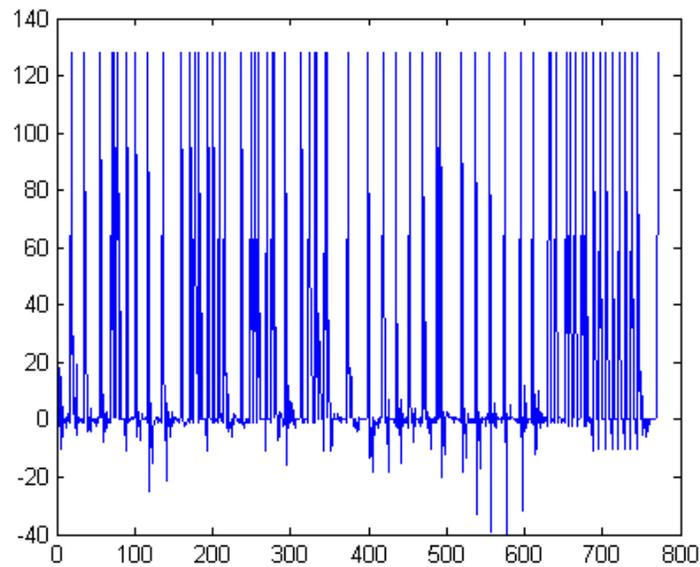


Figure 6: Graph of a sample JPEG approximation vector for Devnagri number 0

### 5.2 Codebook building

The codebook can be built as following:

1. Get 400 vectors for all available database (our database contains 400 written numerals).
2. Group the 400 vectors according to their represented numerals. For instance, the group of number 0 has (in our database) 40 different 0's that were written by 40 different writers so it will have 40 vectors.
3. Average each group which results in unique vector for each group. These are the codes located in the codebook.

**5.3 Classifier:**

After obtaining code book, last step is numeral recognition which is implemented using Euclidean distance classifier. The Euclidean distance classifier is used to examine accuracy of the system designed. The Euclidean distance ( $d$ ) between two vectors X and Y can be defined as:

$$d_E(x, y) = \sqrt{\sum_{i=1}^d (x_i - y_i)^2} \tag{1}$$

**6. RESULT AND DISCUSSION**

JPEG compression property yields high compression ratio which results in minimum image size. Every compressed image has a unique vector which helps to identify each numeral. By using this unique vector, the proposed system has recognized the input numeral after measuring the Euclidean distance between the vector and the vectors in the codebook, then the shortest distance pointed to the corresponding numeral. In addition to the advantage of speed using codebook, it can be universal by means of character's nature (language, writing mode) as well as character's image size. We used 60x60 8-pixel color image as input image.

Table 1: The recognition accuracy

Devnagri Numeral	Recognition Accuracy (%)
0	66
1	42
2	63
3	59
4	58
5	37
6	50
7	67
8	50
9	58
Overall	55

The code book is obtained with the help set of available database. The proposed algorithm is tested on input numerals and the accuracy of percentage recognitions for each character obtained. The individual and average recognition accuracy of numeral is shown table 1. The system was able to recognize the characters during short time comparing with any existing system using ANN because it saves time taken by features extractor as well as it uses codebook

(lookup table) instead of ANN. Misrecognition occurred to some letters is because of the nature of the handwritten character itself. The proposed algorithm is implemented in Matlab 7.9 The main recognition errors were due to abnormal writing and ambiguity among similar shaped numerals as shown in figure 7.

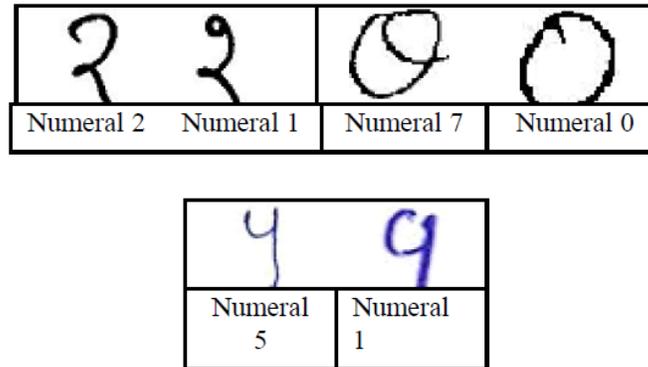


Figure 7: Confusing handwritten numerals

## 7. CONCLUSION

A fast and robust method is proposed in this paper for achieving better recognition rates for handwritten Devnagri numerals which is not based on ANN to avoid the time consuming problems. It is based JPEG image compression algorithm which generates unique vector which helps to identify each numeral. The result was considerably high in terms of recognition rate. Our future work aims to improve classifier to achieve still better recognition. The proposed method can be extended to recognition of numerals of other Indic scripts.

## REFERENCES

- [1] Liana M. & Venu G. (2006), "Offline Arabic Handwriting Recognition: A Survey", IEEE, Transactions on Pattern Analysis and Machine Intelligence, vol. 28, No. 5, pp. 712-724.
- [2] Abdurazzag Ali Aburas and Salem M. A. Rehiel Ariser (2007), "Off-line Omni-style Handwriting Arabic Character Recognition System Based on Wavelet Compression", Vol. 3 No. 4 123-135.
- [3] Abdurazzag Ali Aburas and Salem Ali Rehiel, "JPEG for Arabic Handwritten Character Recognition: Add a Dimension of Application", Advances in Robotics, Automation and Control ISBN 78-953-7619-16-9, pp. 472, October 2008.
- [4] V.N. Vapnik (1995), *the Nature of Statistical Learning Theory*, Springer, New York.
- [5] R.O. Duda, P.E. Hart, D.G. Stork Pattern Classification, second ed. Wiley-Inter science, New York (2001).
- [6] T. Hastie, R. Tibshirani, J. Friedman, (2001), *The elements of statistical learning*, Springer Series in Statistics, Springer, New York.
- [7] U. Pal and B. B. Chaudhuri (2000), "Automatic Recognition of Unconstrained Off-line Bangla Handwritten Numerals", Proc. Advances in Multimodal Interfaces, Springer Verlag Lecture Notes on Computer Science (LNCS-1948), pp371- 378.
- [8] N. Tripathy, M. Panda and U. Pal (2004), "A System for Oriya Handwritten Numeral Recognition, SPIE Proceedings", Vol.-5296, pp174-181.
- [9] Y. Wen, Y. Lu, P. Shi (2007), "Handwritten Bangla numeral recognition system and its application to postal automation", Pattern Recognition, Volume 40 pp 99-107.
- [10] G.G. Rajput and Mallikarjun Hangarge (2007), "Recognition of isolated and written Kannada numerals based on image fusion method", PreMI07, LNCS 4815, pp153-160.
- [11] Govindan V K, Shivaprasad A. P. (1990), "Character recognition a review", Pattern Recognition Volume 23, Issue 7, pp 671-683.

- [12] Amin A. (1997), "Off-line Arabic character recognition: Survey", Proc. 4th Int. Conf. on Document Analysis and Recognition, Munich (IEEE Press).
- [13] Plamondon R, Srihari S N (2000), "On-line and off-line handwriting recognition: comprehensive survey", IEEE Trans. Pattern Anal. Machine Intel PAMI-22, pp 63-84.
- [14] Lam L, Suen C. Y. (1986), "Structural classification and relaxation matching of totally unconstrained handwritten ZIP codes", Pattern Recognition, pp15-19.
- [15] Mai T, Suen C. Y. (1990), "A generalized knowledge-based system for recognition of unconstrained hand-written numerals", IEEE Trans. Syst., Man Cybern. SMC-20: pp 835-848.
- [16] Kimura F, Shridhar M. (1991), "Handwritten numerical recognition based on multiple algorithms", Pattern Recognition, Volume 24, 969-983.
- [17] Chen L-H, Lieh J. R. (1990), "Handwritten character recognition using a two layer random graph model by relaxation matching", Pattern Recognition 23: pp1189-1205.
- [18] Jain A. K, Zongkar D. (1997), "Representation and recognition of handwritten digits using deformable templates", IEEE Trans. Pattern Anal. Machine Intel PAMI-19: pp 1386-1391.
- [19] LeCun Y, Boser B, Denker J S, Henderson D, Howard R B, Hubbard W, Jackel L D (1989), "Back propagation applied to Handwritten zip code recognition", Neural Comput. 1 pp 541-551.
- [20] Knerr S, Personnaz L, Dreyfus G (1992), "Handwritten digit recognition by neural networks with single layer training", IEEE Trans. Neural Networks 3: 303-314.
- [21] Wang J, Jean J (1993), "Resolving multi font character confusion with neural networks", Pattern Recogn. 26: 175-187.
- [22] Sinha R M K, Mahabala H (1979), "Machine recognition of Devanagari script" IEEE Trans. Syst., Man Cybern. SMC-9: pp 435-449.
- [23] Bansal V, Sinha R. M. K. (2001), "A complete OCR for printed Hindi text in Devanagari script", Proc. 6<sup>th</sup> Int. Conf. on Document Analysis and Recognition, Washington (IEEE Press).
- [24] Dutta A K, Chaudhury S (1993), "Bengali alpha-numeric character recognition using curvature features" Pattern Recogn. 26: 1757-1770.
- [25] Chaudhuri B., B, Pal U, (1998) "A complete printed Bangla OCR system" Pattern Recogn. 31: pp 531-549.
- [26] G. G. Rajput, S. M. Mali (2010), "Marathi Handwritten Numeral Recognition using Fourier Descriptors and Normalized Chain Code" IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition, RTIPPR,141.
- [27] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner (1998), "Gradient-based learning applied to document recognition", Proc. IEEE 86 (11), pp 2278-2324.

#### AUTHORS

**Gajanan Birajdar** is working as Assistant Professor in the department of Electronics & Telecommunication Engineering at SIES Graduate School of Technology, Navi Mumbai, India. He obtained B.E. (Electronics) from Dr. BAM University, Aurangabad, Maharashtra and M. Tech. (Elect. & Telecom) from Dr. BAM Technological University, Lonere, India. He has been in teaching for the past 14 years. He is life member of ISTE and IETE. He has attended several seminars and workshops. He has published papers in international journals. His area of research includes Ad hoc networks, image and speech processing.



**Mansi Subhedar** is working as Lecturer in the department of Electronics & Telecommunication Engineering at SIES Graduate School of Technology, Navi Mumbai, India. She obtained B.E. (Elect. & Telecom) from Dr. BAM University, Aurangabad, Maharashtra and M.E. (Electronics) from Mumbai University. She has been in teaching for the past five years. She is life member of ISTE. She has attended several workshops and conferences. She has published and presented papers in various national conferences across India. Her research area includes next generation networks, sensor networks and signal processing.

