



# An Evaluation of Different Feature Extractors and Classifiers for Offline Handwritten Devnagari Character Recognition

**Brijmohan Singh**

*bmsingh1981@gmail.com* and

**Ankush Mittal**

*dr.ankush.mittal@gmail.com*

*Dept. of CSE, College of Engineering Roorkee,  
Roorkee, Uttarakhand 247 667, India*

**Debashis Ghosh**

*ghoshfec@iitr.ernet.in*

*Dept. of ECE, Indian Institute of Technology Roorkee,  
Roorkee, Uttarakhand 247 667, India*

## Abstract

Research on Optical Character Recognition (OCR) of Devnagari script is very challenging due to the complex structural properties of the script that are not observed in most other scripts. Devnagari is the script for Hindi, which is the official language of India. Recognition of Devnagari characters poses great challenge due to the large variety of symbols and their proximity in appearance. In this paper, we use two different methods for extracting features from handwritten Devnagari characters, the Curvelet Transform and the Character Geometry, and compare their recognition performances using two different classifiers, viz., the Support Vector Machine (SVM) with Radial Basis Function (RBF), and the  $k$ -Nearest Neighbour ( $k$ -NN) classifier. Different classification accuracy measures, such as True Positive (TP) Rate, False positive (FP) Rate, Precision, Recall and F-Measure, are used for the purpose. Results obtained show that Curvelet features with  $k$ -NN classifier performs the best, yielding accuracy as high as 93.8%.

*Keywords:* Optical Character Recognition, Curvelet Transform, Character Geometry, Support Vector Machine,  $k$ -Nearest Neighbour Classifier

## 1. Introduction

Offline handwritten character recognition is an important area of research in Document Analysis and Recognition (DAR). The main objective of DAR is to read the intended information from the document using computer which acts as a surrogate human. The outcome of a DAR system is usually in the ASCII format [1]. Applications of DAR include office and library automation, and finds extensive use in publishing houses [2, 3]. It helps visually impaired to “read” when interfaced with a voice synthesizer. Some other applications include postal service assistance, reading forms, automatic processing of criminal records in police station, etc. However, for reliable working of all these, it is desired to have character recognition accuracy close to 100%. A mistake in interpreting the characters may lead to a serious consequence in the automation process. For example, a letter may be despatched to an incorrect address, data entered in official records may be wrong, and so on.

While significant advances have been achieved in recognizing Roman based scripts like English, ideographic characters (Chinese, Japanese, Korean, etc.) and Arabic to some extent, OCR research on Indian scripts is very thin. Only few works on some of the major Indian scripts like Devnagari, Bangla, Gurumukhi, Tamil, Telugu, etc. are available in the literature. The era of handwritten Devnagari character recognition was started in the early days of OCR research by Sethi and Chatterjee [4]. Later, Kumar and Singh proposed a Zernike moment based technique and obtained 80% recognition rate [5]. Sharma

et al. made important contribution using Chain Code Histogram with an accuracy of 80.36% [6]. Hanmandlu et al. published a Fuzzy based system in [7] and achieved 90.65% accuracy. In [8], Pal et al. proposed a method based on directional information obtained from the arc tangent of the gradient yielding 94.24% recognition rate. Another significant contribution is due to Arora et al. in which they proposed a multi-feature extraction based technique thereby achieving 92.8% accuracy [9]. Recently, Pal et al. proposed SVM and MQDF (Modified Quadratic Discriminant Function) based scheme and achieved recognition accuracy as high as 95.13% [10]. A comparative study of different Devnagari character recognizers using features based on curvature and gradient information obtained from binary as well as gray-scale document images is available in [11].

Many of the challenges in offline Devnagari handwritten character recognition research is due to the complex structural properties of the script. Mixing of cursive and non-cursive character symbols makes the problem still more difficult. This calls for extensive research in Devnagari OCR. Most of the Indian scripts like Devnagari, Bangla, Gujrati, Gurumukhi, etc. belong to the same Brahmic family of scripts having a common origin [12]. Consequently, many structural similarities are observed among their characters. Therefore, a character recognition scheme developed for one Brahmic script may also be applicable to other scripts of Brahmic origin. In view of this, we propose to use Curvelet features for isolated handwritten Devnagari character recognition which has recently been used successfully for recognition of handwritten Bangla characters in an SVM recognizer with an overall accuracy of 95.5% [13]. Following this, we classify the extracted features in two different classifiers, viz., SVM and  $k$ -NN, and choose the one that gives better performance. Finally, the performance of the proposed handwritten Devnagari character recognition scheme using Curvelet features is compared with that using Character Geometry features.

## 2. Features of Devnagari Script

Devnagari is the script used for writing Hindi which is the official language of India [14]. It is also the script for Sanskrit, Marathi and Nepali languages. Devnagari script consists of 13 vowels and 33 consonants shown in Fig. 1. These characters are called the basic characters. The most challenging task in a Devnagari OCR system is to distinctly recognize similar looking characters, as shown in Fig. 2.

## 3. Steps Involved in Character Recognition

As with most pattern recognition tasks, Devnagari character recognition also involves the following important steps: Pre-processing, feature extraction and classification.

### 3.1 Pre-processing

In off-line OCR, document image to be recognized is first captured by a scanner. Pre-processing of the scanned image significantly improves the efficiency of the document analysis process [15]. Binarization is an important preprocessing step which converts gray image

|   |   |   |   |   |   |   |    |   |    |     |      |
|---|---|---|---|---|---|---|----|---|----|-----|------|
| अ | आ | इ | ई | उ | ऊ | ए | ऐ  | औ | अं | अः  | अँ   |
| a | ā | i | ī | u | ū | e | ai | o | au | ang | angh |

(a) Devnagari Vowels

|      |     |    |     |     |     |     |    |     |     |     |      |     |
|------|-----|----|-----|-----|-----|-----|----|-----|-----|-----|------|-----|
| क    | ख   | ग  | घ   | ङ   | च   | छ   | ज  | झ   | ञ   | ट   | ठ    | ड   |
| KA   | KHA | GA | GHA | NGA | CH  | CHA | JA | JHA | NYA | TTA | TTHA | DDA |
| ढ    | ण   | त  | थ   | द   | ध   | न   | प  | फ   | ब   | भ   | म    | य   |
| DDHA | NNA | TA | THA | DA  | DHA | NA  | PA | PHA | BA  | BHA | MA   | YA  |
| र    | ल   | व  | श   | ष   | स   | ह   |    |     |     |     |      |     |
| RA   | LA  | VA | SHA | SSA | SA  | HA  |    |     |     |     |      |     |

(b) Devnagari Consonants

**Fig. 1:** Devnagari Characters

|    |     |    |    |    |     |    |     |     |    |     |    |     |   |   |
|----|-----|----|----|----|-----|----|-----|-----|----|-----|----|-----|---|---|
| अ  | आ   | इ  | ई  | उ  | ऊ   | ए  | ऐ   | ओ   | औ  | क   | ख  | ग   | घ | ङ |
| a  | ang | ah | ba | va | bha | ma | sha | cha | pa | ssa | ka | pha |   |   |
| य  | भ   | ट  | ड  | ढ  | ण   |    |     |     |    |     |    |     |   |   |
| ya | bha | ta | da | dh | na  |    |     |     |    |     |    |     |   |   |

Fig. 2: Some similar looking characters in Devnagari.

into a binary image. However, the existing global binarization methods are generally not suitable for a non-uniformly illuminated document as the threshold value is same for the whole document image but not the illumination. Therefore, locally adaptive binarization techniques are usually better suited in case of non-uniformly illuminated and degraded documents. Nevertheless, in our work, all scanned images are assumed to be uniformly illuminated. Accordingly, the global thresholding method proposed by Otsu [16] is used without any problem. The sample results of binarization are shown in Fig. 3.

### 3.2 Feature Extraction

After pre-processing the document image, features relevant to the classification problem are extracted. These feature values are subsequently fed to the classifier.

#### 3.2.1 Curvelet transform based feature extraction

Some earlier proposed character features used for recognition include directed chain code [6], intersection, shadow feature, chain code histogram and straight line fitting features [7], gradient and curvature information [11]. In handwritten text, one common but important feature is the orientation of the text written by the writer. Also, for large set of characters, as in Bangla, Devnagari, etc., automatic curve matching is highly useful. Accordingly, Curvelet transform has been proposed in [13] for extracting features from handwritten Bangla characters and has been used successfully with an overall accuracy of 95.5%. Since Devnagari and Bangla belong to the same Brahmic family of scripts having a common origin, many structural similarities are observed among their characters. Considering this, we explore the use of Curvelet Transform for handwritten Devnagari character recognition. Curvelet represents edges and singularities along curves more precisely with the needle-shaped basis elements [17]. The elements possess super directional sensitivity and capability to capture smooth contours. Since curvelets are two dimensional waveforms that provide a new architecture for multi-scale analysis, they may be used to distinguish similar appearing characters better. In our proposed curvelet-based feature extraction, the characters in a document image are first extracted using conventional methods. Each character sample is then cropped and resized so as to fit within a frame of standard width and height. Following this, the digital Curvelet transform at a single scale is applied to each of the character samples in the document to obtain Curvelet feature coefficients characterizing the character. In this work, we compute 1024 (i.e.,  $32 \times 32$ ) feature coefficients, as shown in Fig. 4.

The most widely used wavelet transform works well with edge discontinuities but not with curve discontinuity [18]. Since many of the Devnagari characters not only consist

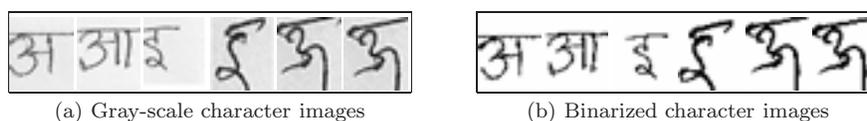
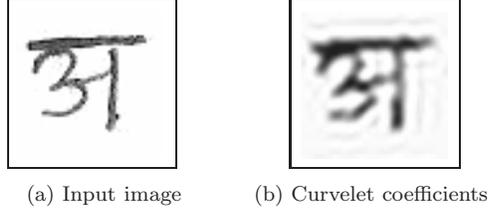


Fig. 3: Sample results of binarization.



**Fig. 4:** Curvelet coefficients of an input character image.

of edge discontinuities but also consist of curve discontinuities, wavelet transform is not suited for feature extraction in Devanagari characters. On the other hand, the curve discontinuities in any character are well handled by Curvelet transform even with very few numbers of coefficients [13]. Candes and Donoho [17, 19] showed that curvelets are better than wavelets at representing edges. The Curvelet frame preserves the important properties, such as parabolic scaling, tightness and sparse representation for surface-like singularities of co-dimension one. Hence, curvelet-based features are likely to work well for Devanagari character recognition. The Curvelet transform of an image function  $f$  is achieved by the following algorithm.

**Algorithm 1: Curvelet Transform**

1. *Sub-band decomposition:* The image is divided into resolution layers where each layer contains details of different frequencies, i.e.,

$$f \mapsto (P_0 f, \Delta_1 f, \Delta_2 f, \dots) \quad (1)$$

where  $P_0$  is the low-pass filter, and  $\Delta_k$  are band-pass/high-pass filters.

2. *Smooth Partitioning:* Each sub-band is smoothly windowed into “squares” of an appropriate scale, i.e.,

$$\Delta_s f \mapsto (\omega_Q \Delta_s f)_{Q \in Q_s} \quad (2)$$

where  $Q_s$  denotes the dyadic square of side  $2s$  and  $\omega$  is a smooth windowing function with ‘main’ support of size  $2s \times 2s$ .

3. *Renormalization:* Each resulting square is renormalized to unit square, i.e.,

$$g_Q = 2^{-s} (T_Q)^{-1} (\omega_Q \Delta_s f), \quad Q \in Q_s \quad (3)$$

4. *Ridgelet analysis:* Each square is analyzed in the ortho-ridgelet system, i.e.,

$$\alpha_\mu = \langle g_Q, p_\lambda \rangle, \quad \mu = (Q, \lambda) \quad (4)$$

### 3.2.2 Dimensionality reduction

A significant problem in using Curvelet transform is that it gives a large dimensional feature space resulting in increased consumption of memory-space and computational time. Dimensionality reduction is therefore an obvious choice to reduce the computational complexity. There are several methods for dimensionality reduction. Some methods select a few prominent features out of all the features such as in [20, 21]. The Principal Component Analysis (PCA) method, as used in our proposed scheme, reduces the feature space while preserving the information as much as possible [22]. In our experiment, we observe that only 190 principal components out of the original 1024 features are good enough to cover 95% of input signal (feature vector) energy.

### 3.2.3 Character Geometry based feature extraction

In this feature extraction technique, the geometric features [23] of the character contour are extracted. These features are based on the basic line types that form the character skeleton.

#### Algorithm 2: Character Geometry Feature Extraction

1. *Universe of Discourse*: First, universe of discourse is selected because the features extracted from the character image include the positions of different line segments in the character image.
2. *Zoning*: The image is divided into windows of equal size and feature extraction is applied to each individual zone rather than the whole image. In our work, the image was partitioned into 9 equal sized windows.
3. *Starters, Intersections and Minor Starters*: To extract different line segments in a particular zone, the entire skeleton in that zone should be traversed. For this purpose, certain pixels in the character skeleton are defined as starters, intersections and minor starters.
4. *Character traversal*: Character traversal starts after zoning by which line segments in each zone are extracted. First, the starters and intersections in a zone are identified and then populated in a list. Algorithm starts by considering the starter list. Once all the starters are processed, minor starters obtained along the course of traversal are processed. The positions of pixels in each of the line segments obtained during this process are stored. Once all the pixels in the image are visited, the algorithm stops.
5. *Distinguishing line segments*: After all the line segments in the image are extracted, they are classified into any one of the following line-types – Horizontal line, Vertical line, Right-diagonal line, or Left-diagonal line.
6. *Feature Extraction*: After the line type of each segment is determined, feature vector is formed based on this information which include the number and the normalized length of the four different types of lines in each zone. The normalized length of a line is given as

$$\text{Normalized.Length} = \frac{\text{No. of Line Pixels}}{\text{No. of Zone Pixels}} \quad (5)$$

After zonal feature extraction, certain features are extracted for the entire image based on the regional properties, viz., Euler number, regional area, and eccentricity.

### 3.3 Classification

The main task of classification is to use the feature vectors provided by the feature extraction algorithms to assign the object to a category [24]. In our work, we used SVM with RBF kernel, and  $k$ -NN for the classification of Devnagari characters.

#### 3.3.1 Support Vector Machine

Support vector machine (SVM) was developed by Vapnik [25] and is an extensively used tool for pattern recognition due to its many attractive features and promising empirical performance, especially in classification and nonlinear function estimation. SVMs are used for time series prediction and are comparable to radial basis function network.

The performance of SVM classification is based on the choice of kernel function and the penalty parameter  $C$ . In our work, we used SVM classifier with Radial Based Kernel for the classification of isolated characters. The RBF kernel maps nonlinear samples into a higher

dimensional space, and thus can handle the case when the relation between class labels and attributes is nonlinear. The RBF kernel can be described as

$$k(x, z) = \exp(-\gamma \times \|x - z\|^2) \quad (6)$$

Thus, when RFB kernel function is used, there are two parameters  $C$  and  $\gamma$  that need to be selected. Usually, these parameters are selected on a trial and error basis. To obtain a more accurate model, we used the value of cost factor  $C = 20$ .

### 3.3.2 $k$ -Nearest Neighbor Classifier

The  $k$ -Nearest Neighbor ( $k$ -NN) classifier classifies an unknown sample based on the known classification of its neighbors [26, 27, 28]. Given an unknown data, the  $k$ -nearest neighbour classifier searches the pattern space for the  $k$  training data that are closest to the unknown data. These  $k$  training tuples are the  $k$  “nearest neighbours” of the unknown data. “Close-ness” is defined in terms of a distance metric, such as the Euclidean distance. Typically, we normalize the values of each attribute. This helps to prevent attributes with initially large ranges from outweighing attributes with initially smaller ranges (such as binary attributes). Min-max normalization, for example, may be used to transform a value of a numeric attribute  $A$  to  $\nu$  in the range  $[0, 1]$  by computing

$$\nu = \frac{A - \min_A}{\max_A - \min_A} \quad (7)$$

## 4. Evaluation Measures

In this work, we have used True Positive (TP) Rate, False positive (FP) Rate, Precision (P), Recall (R), and F-measure to evaluate the performance efficiency of the classification schemes. TP Rate is the ratio of correctly classified cases to the total number of instances, while FP Rate is the ratio of incorrectly classified cases to the total number of instances. Recall (R) is the ratio of relevant documents found in the search result to the total of all relevant documents. Thus, higher recall value implies that relevant documents are returned more quickly. Precision is the proportion of relevant documents in the results returned. On the other hand, precision is defined as the proportion of the true positives against all the positive results. Precision and Recall are better descriptors when one class is rare. Precision can be seen as a measure of exactness or fidelity, whereas recall is a measure of completeness. F-measure is a way of combining recall and precision scores into a single measure of performance, as follows

$$\text{F-measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (8)$$

## 5. Experimental Results and Discussion

Since standard benchmark data-set for handwritten characters are not available for Devnagari, we collected a data-set of 31860 samples of 46 Devnagari characters written by different writers to evaluate and compare the performance of different feature extractors and classifiers. The data-set was split into two subsets – sets of training and test samples – in the ratio 75:25, respectively. Feature extraction was done on each sample using both Curvelet Transform and Character Geometry at a single scale. The Curvelet-based feature vectors obtained had a dimensionality of 1024. Principal component analysis of the coefficients was performed to reduce the dimension of the feature vectors to 190. In Character Geometry-based algorithm, we extracted 86-dimensional feature vectors and so there was no need to reduce their dimension.

**Table 1:** Character recognition rate in terms of percentage accuracy

|                        | $k$ -NN Classifier |            |         | SVM Classifier |            |         |
|------------------------|--------------------|------------|---------|----------------|------------|---------|
|                        | Vowels             | Consonants | Overall | Vowels         | Consonants | Overall |
| Geometry-based feature | 86.2               | 72.4       | 73.4    | 75.4           | 44.3       | 35.9    |
| Curvelet-based feature | 96.6               | 93.8       | 93.8    | 97.6           | 90.7       | 91.5    |

**Table 2:** Comparison of weighted averages of different evaluation measures

| Feature-type   | Classifier-type | TP Rate | FP Rate | Precision | Recall | F-Measure |
|----------------|-----------------|---------|---------|-----------|--------|-----------|
| Geometry-based | SVM             | 0.565   | 0.023   | 0.572     | 0.565  | 0.566     |
|                | $k$ -NN         | 0.781   | 0.011   | 0.787     | 0.781  | 0.782     |
| Curvelet-based | SVM             | 0.925   | 0.004   | 0.927     | 0.925  | 0.925     |
|                | $k$ -NN         | 0.947   | 0.003   | 0.95      | 0.947  | 0.948     |

We evaluated and compared both the types of features in two different classifiers, viz.,  $k$ -NN and SVM using RBF kernel. Recognition results obtained in our experiments, as given in Table 1, show that Curvelets with  $k$ -NN gives better result (overall accuracy of 93.8%) than the SVM classifier. Performance comparison on the basis of weighted average of different evaluation measures such as True Positive (TP) Rate, False positive (FP) Rate, Precision (P), Recall (R) and F-Measure is given in Table 2. It is observed that Curvelets with  $k$ -NN has the highest value of TP rate (0.947), lowest FP rate (0.003), highest Precision (0.95), highest Recall (0.947) and highest F-measure (0.948) and hence, Curvelets features are the most suitable features for  $k$ -NN classifier.

## 6. Conclusion

This paper proposes a framework for evaluation and comparison of performances of Curvelets and geometry-based features for handwritten Devnagari character recognition using  $k$ -NN and SVM classifiers. The novelty of this work also lies in using PCA for reducing dimensionality of Curvelet features. On the basis of our observations, Curvelet-based feature is found to be highly effective while character geometry features are not found suitable for Devnagari characters having very complex structural properties. Comparison results show that the Curvelet features are more suitable for Devnagari character recognition and with  $k$ -NN classifier gives better results than SVM. Hence, Curvelet transform in combination with  $k$ -NN classifier proves to be useful in Devnagari character recognition. However, the accuracy of proposed scheme may be enhanced by increasing the number of training samples and/or applying the proposed scheme at different resolution levels.

## References

- [1] S. Marinai, Introduction to Document Analysis and Recognition. *Studies in Computational Intelligence*, vol. 90, pp. 1 – 20, 2008.
- [2] Y.Y. Tang, C.Y. Suen, C.D. Yan, and M. Cheriet, Document Analysis and Understanding: A Brief Survey. *Proc. First Intl. Conf. Document Analysis & Recognition*, Saint Malo (France), 1991, pp. 17 – 31, 1991.
- [3] R. Plamondon, and S.N. Srihari, Online and Off-line Handwriting Recognition: A Comprehensive Survey. *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 22, no. 1, pp. 63 – 84, 2000.
- [4] I.K. Sethi, and B. Chatterjee, Machine Recognition of Constrained Hand Printed Devnagari. *Pattern Recognition*, vol. 9, no. 2, pp. 69 – 75, 1977.

- [5] S. Kumar, and C. Singh, A Study of Zernike Moments and Its Use in Devnagari Handwritten Character Recognition. *Proc. Intl. Conf. Cognition & Recognition*, Mandya (India), pp. 514 – 520, 2005.
- [6] N. Sharma, U. Pal, F. Kimura, and S. Pal, Recognition of Offline Handwritten Devnagari Characters Using Quadratic Classifier. *Proc. Indian Conf. Computer Vision Graphics & Image Processing*, Madurai (India), pp. 805 – 816, 2006.
- [7] M. Hanmandlu, O.V. Ramana Murthy, and V.K. Madasu, Fuzzy Model Based Recognition of Handwritten Hindi Characters. *Proc. Ninth Biennial Conf. Australian Pattern Recognition Society on Digital Image Computing Techniques & Applications*, Glenelg (Australia), pp. 454 – 461, 2007.
- [8] U. Pal, N. Sharma, T. Wakabayashi, and F. Kimura, Off-line Handwritten Character Recognition of Devnagari Script. *Proc. Ninth Intl. Conf. Document Analysis & Recognition*, Curitiba (Brazil), pp. 496 – 500, 2007.
- [9] S. Arora, D. Bhattacharjee, M. Nasipuri, D.K. Basu, and M. Kundu, Combining Multiple Feature Extraction Techniques for Handwritten Devnagari Character Recognition. *Proc. IEEE Region 10 Colloquium and Third Intl. Conf. Industrial & Information Systems*, Kharagpur (India), 2008.
- [10] U. Pal, S. Chanda, T. Wakabayashi, and F. Kimura, Accuracy Improvement of Devnagari Character Recognition Combining SVM and MQDF. *Proc. Eleventh Intl. Conf. Frontiers in Handwriting Recognition*, Montreal (Canada), pp. 367 – 372, 2008.
- [11] U. Pal, T. Wakabayashi, and F. Kimura, Comparative Study of Devnagari Handwritten Character Recognition Using Different Feature and Classifiers. *Proc. Tenth Intl. Conf. Document Analysis & Recognition*, Barcelona (Spain), pp. 1111 – 1115, 2009.
- [12] D. Ghosh, T. Dube, and A.P. Shivaprasad, Script Recognition – A Review. *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 32, no. 12, pp. 2142 – 2161, 2010.
- [13] A. Majumdar, and B.B. Chaudhuri, Curvelet-based Multi SVM Recognizer for Offline Handwritten Bangla: A Major Indian Script. *Proc. Ninth Intl. Conf. Document Analysis & Recognition*, Curitiba (Brazil), pp. 491 – 495, 2007.
- [14] P.S. Deshpande, L. Malik, and S. Arora, Characterizing Handwritten Devnagari Characters Using Evolved Regular Expressions. *Proc. IEEE Region 10 Conf. TENCON*, Hong Kong (China), 2006.
- [15] T. Vasudev, G. Hemanthkumar, and P. Nagabhushan, Transformation of Arc-form-text to Linear-form-text Suitable for OCR. *Pattern Recognition Letters*, vol. 28, no. 16, pp. 2343 – 2351, 2007.
- [16] N. Otsu, A Threshold Selection Method from Grey-Level Histogram. *IEEE Trans Systems, Man & Cybernetics*, vol. 9, no. 1, pp. 62 – 66, 1979.
- [17] E. Candes, L. Demanet, D. Donoho and L. Ying, Fast Discrete Curvelet Transform. *SIAM J. Multiscale Modeling & Simulation*, vol. 5, no. 3, pp. 247 – 275, 2006.
- [18] A. Majumdar, Bangla Basic Character Recognition Using Digital Curvelet Transform. *J. Pattern Recognition Research*, vol. 1, pp. 17 – 26, 2007.
- [19] E.J. Candes, and D.L. Donoho, Curvelets – A Surprisingly Effective Nonadaptive Representation for Objects with Edges. *Curves and Surface Fitting*, A. Cohen, C. Rabut, and L. Schumaker (eds.), Vanderbilt University Press, pp. 105 – 120, 2000.
- [20] B. Abraham, and G. Merola, Dimensionality Reduction Approach to Multivariate Prediction. *Computational Statistics & Data Analysis*, vol. 48, no. 1, pp. 5 – 16, 2005.
- [21] S. De Jong, and H.A.L. Kiers, Principal Covariates Regression: Part 1, Theory. *Chemometrics and Intelligent Laboratory Systems*, vol. 14, pp. 155 – 164, 1992.
- [22] I.T. Jolliffe, Principal Component Analysis. *Springer Series in Statistics*, 2nd ed., 2002.
- [23] D. Dileep, A Feature Extraction Technique Based on Character Geometry for Character Recognition. *Proc. Dept. Electronics & Commun. Engg.*, Amrita School of Engineering, pp. 1 – 4, available online at [www.mathworks.se/matlabcentral/fileexchange/24624](http://www.mathworks.se/matlabcentral/fileexchange/24624).
- [24] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, 2nd ed., Wiley, 2000.
- [25] V.N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed., Springer, 2000.
- [26] B.V. Dasarathy, *Nearest Neighbor: Pattern Classification Techniques*, IEEE Computer Society Press, 1990.

- [27] Y. Yang, Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval. *Proc. 17th Annual Intl. ACM SIGIR Conf. Research & Development in Information Retrieval*, Dublin (Ireland), pp. 13 – 22, 1994.
- [28] Y. Yang, and C.G. Chute, An Example-based Mapping Method for Text Classification and Retrieval. *ACM Trans. Information Systems*, vol. 12, no. 3, pp. 252 – 277, 1994.