Journal of Pattern Recognition Research 1 (2007) 17-26

Bangla Basic Character Recognition Using Digital Curvelet Transform

Angshul Majumdar *angshul@gmail.com*
Pricewaterhouse Coopers India Pvt. Ltd.
Salt Lake, Kolkata – 700091, India

Received: 12 December 2006 Accepted: 12 January 2007 Published online: 30 March 2007

Abstract

This paper addresses the problem of Bangla basic character recognition. Multi-font Bangla character recognition has not been attempted previously. Twenty popular Bangla fonts have been used for the purpose of character recognition. A novel feature extraction scheme based on the digital curvelet transform is proposed. The curvelet transform, although heavily utilized in various areas of image processing, has not been used as the feature extraction scheme for character recognition. The curvelet coefficients of an original image as well as its morphologically altered versions are used to train separate k–nearest neighbor classifiers. The output values of these classifiers are fused using a simple majority voting scheme to arrive at a final decision.

Keywords: Optical Character Recognition, Digital Curvelet Transform, Feature Extraction, Bengali Characters

1. Introduction

Pattern recognition in image processing encompasses several areas of research, viz., face recognition, signature recognition, text recognition, and fingerprint recognition. High accuracy text recognition or optical character recognition (OCR) is a challenging task for scripts of languages. The OCR research for the English script has matured. Commercial software is available for reading printed English text. However, for the majority of other scripts such as Arabic [1] and Indian, OCR is still an active domain of research. For English and Kanji scripts, good progress has been made towards the recognition of printed scripts, and the focus nowadays is on the recognition of handwritten characters [2, 3]. OCR research for different Indian languages is still at a nascent stage. There has been limited research on recognition of Oriya [4], Tamil [5], Devanagari [6] and Bengali [7].

This paper focuses on the recognition of basic Bengali characters. There are 50 Bengali characters. And since there are deep thongs in Bengali language, with vowel and consonant modifiers, these 50 characters can be modified to result in more than a thousand. Pal and Chaudhuri's OCR system for Bengali printed script [7] takes into account all these modified characters; however, it was limited to only a single font. This paper concentrates on the recognition of the fifty basic characters and takes into account twenty popular Bengali fonts. Fig. 1 shows the variability of some of the Bengali fonts used in this work.

Various feature extraction schemes have been employed for the purpose of OCR. Multiresolution methods in recent years have proved to be potent for the purpose of feature extraction. Excellent accuracy for handwritten Bangla numerals was achieved by Chaudhuri and his colleagues [8].

In this paper we introduce a new multiresolution feature extraction technique based on the


(a) Raghu Bengali (b) Ekushey Godhuli


(c) Ekushey Punarbhaba (d) Ekushey Sumit

Fig. 1: Variability of Bengali fonts.

digital curvelet transform of Candes and Donoho [9]. The proponents of curvelets showed that wavelets, although good at representing point discontinuities, are not very good at representing edges. The curvelet transform overcomes this limitation and has been successfully used in astrophysics, seismic imaging, and image de-noising. However, we have not found any application of curvelets to feature extraction for OCR.

In an attempt to increase the recognition accuracy, the characters are morphologically thinned and thickened, and the curvelet coefficients of these transformed images are utilized for training separate classifiers. Finally, the output values of these classifiers are fused using a majority voting scheme to make a final decision.

The rest of the paper is organized as follows. Section 2 describes the curvelet-based feature extraction scheme. Section 3 briefly discusses the k-nearest neighbor (KNN) classifier. In Section 4, the implementation along with the experimental setup and the results obtained are discussed. Finally in Section 5, conclusions are made and future work is discussed.

2. Feature Extraction

In this paper we introduce a new feature extraction method based on the curvelet transform of morphologically altered versions of an original character. Previously, Haralick and Kanungo [10] have used morphology for the purpose of character recognition. Their work centered on the extraction of certain primitives, such as blobs, straight lines etc., morphologically and on the usage of a decision tree to determine the character.

Curvelets are known to provide a very good representation of edges in an image. By thinning or thickening an image morphologically, the position of the edges can be changed. Using the curvelet transform, the variation in position of edges are encoded in the transformed domain.

In this work, two thickened and two thinned versions of the image are made. Along with the morphologically changed images, the original image is also used. For each character image there will be five sets of curvelet coefficients. These five sets are used to train five KNN classifiers. If a character is not recognized in its original form from its edge information, it might be recognized if the edge positions are slightly varied. Also, during recognition of test characters, five versions of the input image are made, and curvelet coefficients for the five versions are classified using the corresponding KNN classifiers. The block diagram of the classification scheme is shown Fig. 2.

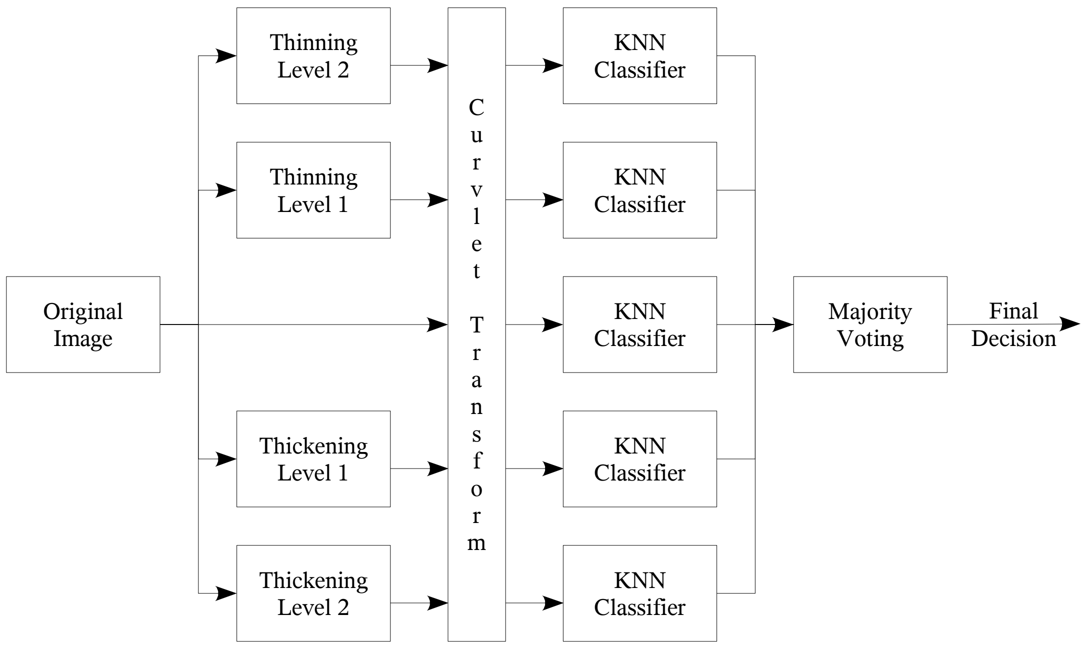

Fig. 2: Curvlet transform-based classification scheme.

The main idea behind using this scheme is that if a character is failed to be recognized in its original form from its edge information, it may be recognized if the edge positions are slightly varied.

2.1 The Curvelet Transform

Wavelets and related classical multiresolution methods use a limited dictionary made up of roughly isotropic elements occurring at all scales and locations. The dictionaries do not exhibit highly anisotropic elements, and there are only a fixed number of directional elements, independent of scale. Despite the success of the classical wavelet viewpoint, there are objects, e.g. images that do not exhibit isotropic scaling and, thus, call for other types of multiscale representation. Classical multiresolution ideas only address a portion of the whole range of possible multiscale phenomena, and there remains a possibility to develop other multiscale transforms.

Candes and Donoho [9] introduced a new system of multiresolution analysis called the curvelet transform. This system differs from wavelet and related systems. Curvelets take the form of basis elements, which exhibit a very high directional sensitivity and are highly anisotropic. In two-dimensions, for instance, curvelets are localized along curves and in three dimensions along sheets.

Studies by neuro-scientists showed that edge processing neurons is done in the earliest and most fundamental stages of the processing pipeline upon which mammalian visual processing is built [17]. But due to this crude directional representation, wavelets, although good at representing point discontinuities, are not good at representing edge discontinuities. Candes and Donoho [9, 11] showed that curvelets are better than wavelets at representing edges.

Field and Olshausen [12] set up a computer experiment for empirically discovering the basis that best represents a database of 16 by 16 image patches. Although this experiment is limited in scale, they discovered that the best basis is a collection of needle shaped filters occurring at various scales, locations and orientations. The interested reader will find a stark similarity between curvelets, which derive from mathematical analysis, and these empirical basis elements arising from data analysis [13].

It is not possible to go into the details of the digital curvelet transform within this paper. The interested reader can refer to the works of Candes and Donoho [14]. A brief procedural definition of the curvelet transform is provided below. The detailed discussion can be found in [14].

2.1.1 The Ridgelet Transform

A basic tool for calculating ridgelet coefficients is to view ridgelet analysis as a form of wavelet analysis in the Radon domain. The Radon transform

$$R: L^2(R^2) \rightarrow L^2([0,2\pi], L^2(R)) \tag{1}$$

is defined by

$$Rf(\theta, t) = \int\int f(x_1, x_2)\,\delta(x_1\cos\theta + x_2\sin\theta - t)\,dx_1\,dx_2\,, \tag{2}$$

where δ is the Dirac delta.

The ridgelet coefficients $R_f(a, b, \theta)$ of an object f are given by the analysis of the Radon transform via

$$R_f(a, b, \theta, t) = \int Rf(\theta, t)\,a^{\frac{1}{2}}\,\psi\left(\frac{t-b}{a}\right)\,. \tag{3}$$

Therefore, the ridgelet transform is precisely the application of a one-dimensional wavelet transform to the slices of the Radon transform, where the angular variable θ is constant and t is varying.

2.1.2 The Discrete Ridgelet Transform

A basic strategy for calculating the continuous ridgelet transform is first to compute the Radon transform $Rf(\theta, t)$ and second to apply a one-dimensional wavelet transform to the slices $Rf(\theta)$.

The projection formula [15] is a fundamental fact about the Radon Transform

$$\hat{f}(\omega\cos\theta, \omega\sin\theta) = \int Rf(\theta, t)\,e^{-2\pi i a\ t}\,dt\,. \tag{4}$$

This says that the Radon transform can be obtained by applying the one-dimensional inverse Fourier transform to the two-dimensional Fourier transform restricted to radial lines through the origin. This enables one to arrive at the approximate Radon transform for digital data based on the FFT. The steps to obtain the approximate Radon transform are summarized in the following algorithm.

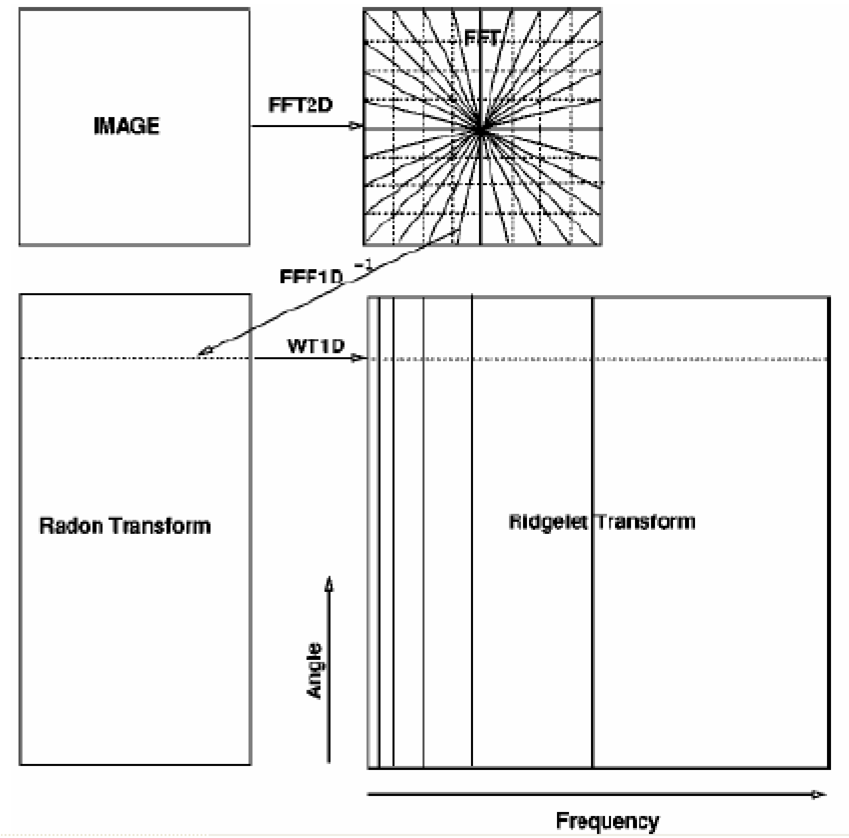

Fig. 3: Pictorial scheme for discrete ridgelet transform.

Algorithm 1 The approximate Radon transform.

1. 2D-FFT - Compute the two-dimensional Fast Fourier Transform (FFT) of f .
2. Cartesian to polar conversion - Using an interpolation scheme, substitute the sampled values of the Fourier transform obtained on the square lattice with sampled values of $\hat{f}$ on a polar lattice: that is, on a lattice where the points fall on lines through the origin.
3. 1D-IFFT - Compute the one-dimensional Inverse Fast Fourier Transform (IFFT) on each line; i.e., for each value of the angular parameter.

2.1.3 The Digital Curvelet Transform

The digital curvelet transform of f is achieved by the following algorithm.

Algorithm 2 The digital curvelet transform.

1. Sub-band Decomposition: A bank of filters is defined. The image f is filtered into subbands with àtrous algorithm

$$f \rightarrow (P_0 f, \Delta_1 f, \Delta_2 f, ...) \quad (5)$$

The different sub-bands $\Delta_s f$ contain details about 2^{-2s} wide.
2. Smooth Partitioning: Each sub-band is smoothly windowed into "squares" of an appropriate scale

$$\Delta_s f \rightarrow (\omega_Q \Delta_s f), Q \in Q_s, \quad (6)$$

where ω_Q is a collection of smooth window localized around dyadic squares

Fig. 4: The first image is that of the coefficients at scale 1, the rest are at scale 2 at different angles.

$$Q=[k_1/2^s,(k_1+1)/2^s]\times[k_2/2^s,(k_2+1)/2^s] \ .$$

3. Renormalization: Each resulting square is renormalized to unit scale

$$g_Q=T_Q^{-1}(\omega_Q \Delta_s f), Q\in Q_s , \tag{7}$$

where $T_Q f(x_1, x_2)=2^s f(2^s x_1-k_1, 2^s x_2-k_2)$ *is a renormalization operator.*
4. Ridgelet Analysis: Each square is analyzed in the orthonormal ridgelet system. This is a system of basis elements p_λ *making an orthonormal basis for* $L^2(R^2)$ *:*

$$\alpha_\mu=\langle g_Q, p_\lambda\rangle \ . \tag{8}$$

In Fig. 4 the curvelet coefficients of the Bangla letter 'O' are shown at two scales. In the first scale there is only a single set of coefficients. In the second scale, the curvelet coefficients are taken at 4 different angles. The images shown are complements of the actual transform coefficients, since the original coefficients only show white edges in a rather dark figure which is difficult to discern.

2.2 Morphological Thinning and Thickening

Curvelets are good at representing edges in images. The recognition of alphabets using curvelet transform is primarily dependent on the edge information. In the paper we attempted to increase the recognition accuracy of the alphabets by changing the edge information in the spatial domain, and taking the curvelet transform of the changed versions of the image to separately train different classifiers. The premise is that if a character is not recognized in its original form, it might be recognized from the changed versions.

The alphabets are morphologically thinned and thickened to derive altered versions. Two levels of thinning and two levels of thickening are performed. The algorithm for thinning and thickening starting from the basic definition of translation is described below.

Given vector x and set A the translation $A+x$, is defined as

$$A+x=\{\alpha+x|\alpha\in A\} \ . \tag{9}$$

The basic Minkowski set addition is defined as follows. Given two sets A and B (the individual elements that comprise B are not only pixels but also vectors as they have a clear coordinate position with respect to $[0,0]$), the Minkowski sum is the result of adding every element of A to every element of B :

$$A+B=\bigcup_{b\in B}(A+b) \ . \tag{10}$$

The Minkowski subtraction is defined as

$$A-B=\bigcap_{b\in B}(A+b) \ . \tag{11}$$

Fig. 5: Original image along with thinned and thickened versions.

From these Minkowski operations, the fundamental morphological operations of dilation and erosion can be defined as

$$D(A,B)=A+B=\bigcup_{b\in B}(A+b) \text{ and} \tag{12}$$

$$E(A,B)=A-(-B)=\bigcap_{b\in B}(A-b) \tag{13}$$

respectively.

Dilation and erosion can be combined to build two higher-order operations of opening and closing. Opening is defined as

$$O(A,B)=A\circ B=D(E(A,B),B)\,. \tag{14}$$

Closing is defined as

$$C(A,B)=A\cdot B=E(D(A,-B),-B)\,. \tag{15}$$

For digital images, A is considered as the image and B is the structuring element used for morphological operations. The Hit-and-Miss operator is defined as follows. Given image A and two bounded and disjoint structuring elements B_1 and B_2, the Hit-and-Miss operation is defined as

$$\text{HitMiss}(A,B_1,B_2)=E(A,B_1)\cap E^C(A^C,B_2)\,. \tag{16}$$

From the above definitions, Thinning and Thickening are defined as

$$\text{Thin}(A,B_1,B_2)=A\cup\text{HitMiss}(A,B_1,B_2) \text{ and} \tag{17}$$

$$\text{Thick}(A,B_1,B_2)=A\cap\text{HittMiss}(A,B_1,B_2)\,. \tag{18}$$

Based on a choice of B_1 and B_2, Thinning and Thickening can be implemented in a variety of ways. In this paper two very simple symmetric structuring elements have been used for this purpose:

$$B_1=\begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix} \text{ and } B_2=\begin{bmatrix}0&1&0\\1&0&1\\0&1&0\end{bmatrix}.$$

The thinned and thickened versions of the Bengali alphabet 'O' along with the original one are illustrated in Fig. 5.

3. K-Nearest Neighbor Classification

The k-nearest neighbors classifier is used for classifying the Bengali alphabets. A detailed discussion on the K-NN classification technique can be found in Dasarathy [16]. A short but formal definition of the K-NN classification is as follows. Given a set of prototype vectors, $T_{XY}=\{(x_1,y_1),(x_2,y_2),\ldots,(x_l,y_l)\}$, the input vectors being $x_i\in X\subseteq R^n$ and corresponding targets being $y_i \in Y = \{1,2,\ldots,c\}$, let $R^n(x) = \{x' : \|x - x'\| \leq r^2\}$ be a ball centered in the vector x in which K prototype vectors $x_i, i \in \{1,2,\ldots,l\}$ lie, i.e. $|x_i : x_i \in R^n(x)| = K$. The k-nearest neighbor classification rule $q : X \rightarrow Y$ is defined as

$$q(x) = \arg\max \; v(x, y), \tag{19}$$

where $v(x, y)$ is the number of prototype vectors x_i with targets $y_i = y$, which lie in the ball $x_i \in R^n(x)$.

4. Bangla Basic Character Recognition Implementation

The experiments were carried out in Matlab 6.5, on a 32-bit AMD Athlon 1.4 GHz processor, with 256 MB RAM. The curvelet transformation was done using the Curvelet 2.0 toolbox, available from http://www.curvelet.org. The k-nearest neighbor classifier was implemented using the Statistical Pattern Recognition Toolbox (stprtool) available from http://cmp.felt.cvut.cz. The morphological operations were performed using Matlab's Image Processing Toolbox.

Twenty Bengal fonts were used in this paper. The considered fonts are Raghu Bengali, Aakash, Ekushey Aazad, Ekushey Durga, Ekushey Godhuli, Ekushey Lohit, Ekushey Mohua, Ekushey Puja, Ekushey Punarbhaba, Ekushey Saraswati, Ekushey Sharifa, Ekushey Sumit, Likhan, Mukti, Mitra Mono, Rupali, Sagar, Solaimanlipi and SutonnyBanglaOMJ. For each of the fonts 10 different font sizes viz., 8, 10, 12, 16, 20, 24, 28, 36, 48 and 72 points were taken. The entire data set consisted of 50 characters for each of the font sizes and each of the 20 fonts, making a total of 10,000 samples.

The experiments on recognition of Bengali printed characters were set into two parts. In the first part eight different font sizes of all the 20 fonts were used as the prototype for the K-NN classifier. The remaining two sizes were used for testing. For each of the fonts, the two different sizes that need to be tested were selected randomly, i.e. say for the Aakash font the testing set comprised of sizes 10 and 28 while for Solaimanlipi the font sizes for testing were randomly selected to be 16 and 24 points for the other different fonts the selected sizes were still different. For each of the fonts, the rest eight font sizes served as the prototype for the K-NN.

In the second part of the experimentation, the four fonts were randomly selected for testing and the rest 16 served as the prototypes for the K-NN classifiers. No division was made based on the font sizes, for the testing all the sizes for the four fonts were considered; the same being true for the 16 prototypes used by the K-NN.

As discussed earlier, the original image was thickened twice and thinned twice to form four morphologically changed version of the image. The digital curvelet transform was performed on these five images of the same character at a single scale. The curvelet coefficients served as the feature set for training five separate K-NN classifiers.

The final decision from the five classifiers was made by majority voting. Let $D = \{D_1, D_2, D_3, D_4, D_5\}$ be the set of the five classifiers such that $D_i : R^n \rightarrow \Omega$, where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_{50}\}$, assigns $x \in R^n$ a class label $\omega_j \in \Omega$. The majority vote method for combining classifier decisions is to assign a class label ω_j to x that is supported by the majority of the classifiers D. In a situation where there was no clear majority among the classifiers, the output from classifier with the highest recognition accuracy was chosen to be the output.

Both the experiments were repeated five times. The results of the experiments are presented in the tables below.

Table 1: Results of testing with different font sizes.

Experiment	1	2	3	4	5
Accuracy (%)	97.35	96.80	96.70	97.20	97.15

Table 2: Results of testing with different fonts.

Experiment	1	2	3	4	5
Accuracy (%)	95.80	96.80	96.45	95.90	96.70

While testing for different font sizes, the best results were achieved during experiment No. 1. And for testing with different fonts, the best results are obtained for experiment No. 2. The detailed results for these two experiments are as follows.

Table 3: Detailed results for experiment No. 1 for testing different font sizes.

Classifier trained with	**Recognition Accuracy (%)**
Original image	94.90
Single level thinning	94.55
Two level thinning	92.75
Single level thickening	93.80
Two level thickening	93.65
Overall accuracy	**97.35**

Table 4: Detailed results for experiment No. 2 for testing different fonts.

Classifier trained with	**Recognition Accuracy (%)**
Original image	94.10
Single level thinning	93.80
Two level thinning	91.95
Single level thickening	92.90
Two level thickenning	92.50
Overall accuracy	**96.80**

5. Conclusion

This paper addresses, for the first time, Bengali characters multifont recognition. A new feature extraction method based on the curvelet transform of morphologically altered versions of an original character is introduced. The results are promising, considering a 50 class pattern recognition problem. Future work includes grouping the characters in small clusters and using a hierarchical classification scheme to recognize the alphabets. An attempt will be made to collect as many Bengali fonts as possible so as to increase the generalization of the classification scheme.

References

[1] A. H. Hassin, X. L. Tang, J. F. Liu, and W. Zhao, "Printed Arabic character recognition using HMM," *Journal of Computer Science and Technology*, vol. 19, no. 4, pp. 538-543, 2004.

[2] M. Hanmandlu, K. Harish, and M. K. R. Murali, "Neural based handwritten character recognition." in *Proc. 5th Int. Conf. Document Analysis and Recognition*, 1999, pp. 241.

[3] L. Guo-hong and S. Peng-fei, "An approach to offline handwritten Chinese character recognition based on segment evaluation of adaptive duration," *Journal of Zhejiang University SCIENCE*, vol. 5, no. 11, pp. 1392-1397, 2004.

[4] B. B. Chaudhuri, U. Pal, and M. Mitra, "Automatic recognition of printed Oriya script," in *Proc. 6th Int. Conf. Document Analysis and Recognition (ICDAR01)*, 2001, pp. 0795.

[5] G. Siromoney, R. Chandrasekaran, and M. Chandrasekaran, "Computer recognition of printed Tamil characters," *Pattern Recognition*, vol. 10, no. 4, pp. 243-247, 1978.

[6] V. Bansal R. M. K. Sinha, "A complete OCR for printed Hindi text in Devanagari script," in *Proc. 6th Int. Conf. Document Analysis and Recognition (ICDAR01)*, 2001, pp.800–804.

[7] U. Pal and B. B. Chaudhuri, "OCR in Bangla: an Indo-Bangladeshi language," in *Proc. 12th IAPR Int. Conf. on Pattern Recognition, Conf. B: Computer Vision & Image Processing*, 1994, pp. 269–273.

[8] U. Bhattacharya and B. B. Chaudhuri, "Fusion of combination rules of an ensemble of MLP classifiers for improved recognition accuracy of handprinted Bangia numerals," in *Proc. 8th Int. Conf. Document Analysis and Recognition (ICDAR05)*, 2005, pp. 322-326.

[9] E. J. Candes and D. L. Donoho. (2006, December). Curvelets: A surprisingly effective nonadaptive representation for objects with edges [Online]. Available http://www.curvelet.org/papers/Curve99.pdf

[10] T. Kanungo and R. M. Haralick, "Character recognition using mathematical morphology," in *Proc. USPS 4th Advanced Technology Conference*, Washington, D.C., 1990, pp. 973-986.

[11] E. J. Candes and D. L. Donoho. (2006, December). Continuous curvelet transform: I. Resolution of the wavefront set [Online]. Available www.acm.caltech.edu/~emmanuel/papers/ContCurvelet Transform-I.pdf

[12] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 38, pp. 607–609, 1996.

[13] E. J. Candes and F. Guo. (2006, December). New multiscale transforms, minimum total variation synthesis: Applications to edge-preserving image reconstruction [Online]. Available www.acm.caltech.edu/~emmanuel/papers/TV_synthesis.pdf

[14] E. J. Candes, L. Demanet, D. L. Donoho, and L. Ying. (2006, December). Fast discrete curvelet transforms [Online]. Available http://www.curvelet.org/papers/FDCT.pdf

[15] S. R. Deans, *The Radon transform and some of its applications*. New York: John Wiley & Sons, 1983.

[16] B. V. Dasarathy, *Nearest neighbor pattern classification techniques*. IEEE Computer Society Press, New York, 1991.

[17] H. Moravec, *ROBOT: Mere Machine to Transcendent Mind*, Oxford University Press, 1998.