Accuracy Improvement of Devnagari Character Recognition Combining SVM and MQDF

Umapada Pal, Sukalpa Chanda
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute
203 B.T. Road, Kolkata-700108, India
E-mail: umapada@isical.ac.in

Tetsushi Wakabayashi, Fumitaka Kimura
Graduate School of Engineering
Mie University
1577 Kurimamachiya-cho, TSU
Mie 514-8507, Japan

Abstract

This paper deals with the recognition of off-line handwritten Devnagari characters. Here two sets of feature are computed and two classifiers are combined to get higher accuracy of Devnagari character recognition. Dimension of the features vector of each set is 392. First feature set is computed based on the directional information obtained from the arc tangent of the gradient. Since most of the Devnagari handwritten characters have some curve-like parts, curvature-based feature guided by gradient information is computed for the second set of features. Combined use of Support Vector Machines (SVM) and Modified Quadratic Discriminant Function (MQDF) are applied here for better performance of Devnagari character recognition.

Keywords: Handwritten character recognition, Indian script, Devnagari character, Curvature feature.

1. Introduction

Recognition of handwritten characters has been a popular research area for many years because of its various application potentials. Some of its potential application areas are postal automation, bank cheque processing, automatic data entry, etc. There are many pieces of work towards handwritten recognition of Roman, Japanese, Chinese and Arabic scripts, and various approaches have been proposed by the researchers towards handwritten character recognition [3,6,12]. Although there are many script and languages in India but not much research has been done for the recognition of handwritten Indian characters [8,9]. In this paper, we propose a system based on Support Vector Machines (SVM) and Modified Quadratic Discriminant Function (MQDF) for the recognition of off-line handwritten Devnagari characters.

First research report on handwritten Devnagari characters was published in 1977 [13] but not much research work is done after that. At present researchers have started to work on handwritten Devnagari characters. Few research reports are available towards Devnagari numeral recognition [4,11] but to the best of our knowledge there are only three research reports available on Devnagari off-line handwritten character recognition [7,10,14] after the year 1977. One of these work is due to Kumar and Singh [7] and they proposed Zernike moments based approach for Devnagari character recognition. Other two pieces of work on Devnagari character recognition are proposed by us [10,14] where MQDF has been used.

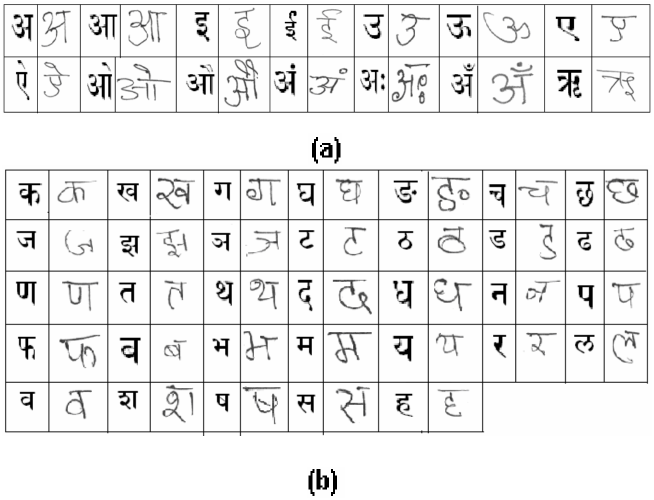
In this paper, an off-line Devnagari handwritten character recognition scheme is proposed using SVM and MQDF. Two sets of feature are computed for the experiment. Dimension of the feature vector of each set was 392. The first feature set is computed based on the directional information obtained from the arc tangent of the gradient [10,15]. Since most of the Devnagari handwritten characters have some curve-like parts, curvature-based feature guided by gradient information is computed in the second set. To get the curvature features, at first, a non-linear size normalization technique is applied and the normalized image is then segmented into 49 x 49 blocks. Next, curvature is computed using bi-quadratic interpolation method and the curvatures are quantized into 3 levels according to concave, linear and convex regions. Direction of gradient is then quantized into 32 levels with $\pi/16$ intervals, and strength of the gradient is accumulated in each of the 32 directions and in each of the 3 curvature levels of each block. A spatial resolution is made to get 7×7 blocks from 49×49 blocks and a directional resolution is made to get 8 directions from 32 directions. Because of the use of curvature features for 3 levels, we get 1176 (7×7 blocks × 8 directions × 3 levels) dimensional features. Finally, using Principal Component Analysis (PCA) we reduce the dimension 1176 to 392. In the proposed scheme at first gradient-based feature is used in a MQDF. To get higher reliability of the system, some samples are rejected by MQDF. Since many of the rejected samples of MQDF have curve-like shape, curvature feature is computed on the rejected samples and fed to SVM. We got higher accuracy by the proposed scheme.

Rest of the paper is organized as follows. In Section 2 properties of Devnagari language are discussed. Feature extraction procedures are reported in Section 3. In Section 4, we briefly explain the classifiers used for the recognition purpose. The experimental results are discussed in Section 5. Finally, conclusion is given in Section 6.

2. Properties of Devnagari language

Devnagari is the most popular script in India. Also Hindi, the most popular and national language in India is written in Devnagari script. Moreover, Hindi is the third most popular language in the world [8]. Thus, the work on Devnagari script is very useful for the country.

The alphabet of the modern Devnagari script consists of 14 vowels and 33 consonants. These characters are called *basic characters*. The basic characters of Devnagari script are shown in Fig.1. Writing style in Devnagari script is from left to right. The concept of upper/lower case is absent in Devnagari script. In Devnagari script a vowel following a consonant takes a modified shape. Depending on the vowel, its modified shape is placed at the left, right (or both) or bottom of the consonant. These modified shapes are called *modified characters*. A consonant or vowel following a consonant sometimes takes a compound orthographic shape, which we call as *compound character*. Compound characters can be combinations of two consonants as well as a consonant and a vowel. Compounding of three or four characters also exists in the script. There are about 280 compound characters in Devnagari [8]. At present no dataset on Devnagari handwritten compound characters is available and hence we consider only basic characters of Devnagari script.

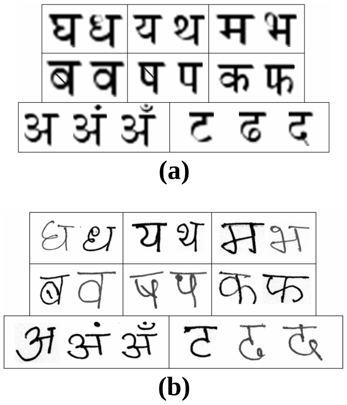


(a)

(b)

Figure1. Samples of printed and handwritten Devnagari basic characters (a) Vowels (b) Consonants. To get an idea about the shape of the character, samples of printed characters are shown on the left side of the respective handwritten characters.

The complexity of a handwritten character recognition system increases mainly because of various writing styles of different individuals. Most of the errors in such system arise because of the confusion among the similar shaped characters. In Devnagari there are many similar shaped characters. Examples of some groups of similar shaped characters are shown in Fig.2. To get an idea of similar shaped printed as well as handwritten characters, we provide the samples of both printed and handwritten Devnagari characters in Fig.2. Although there are some differences between the samples of a group in the printed characters but the difference in the corresponding handwritten samples is very less. From the Fig.2(b) it can be seen that shapes of two or more characters of a group are very similar due to handwritten style of different individual and such shape similarity is the main reason of low recognition rate.



(a)

(b)

Figure 2. Examples of some similar shaped Devnagari characters. (a) printed samples (b) handwritten samples of (a).

3. Feature extraction

Here we use two sets of 392-dimensional feature. The first set is gradient-based feature and the second set is curvature-based feature. The computation methods of the two feature sets are given as follows.

3.1 Computation of gradient feature

To get gradient feature, at first, a 2 x 2 mean filtering is applied 4 times on the input gray level image and a non-linear size normalization is done on the image [6]. Here the image is normalized into 148 x 148 pixels and this size is decided from the experiment. Normalized image is then segmented into 49 x 49 blocks. Compromising trade-off between accuracy and complexity, this block size is decided from the experiment. A Roberts filter is then applied on the normalized image to obtain gradient image. Next, the arc tangent of the gradient (direction of gradient) is initially quantized into 32 directions and the strength of the gradient is accumulated with each of the quantized direction. By strength of gradient (SG) we mean $SG=\sqrt{(\Delta u)^2+(\Delta v)^2}$, and by direction of gradient

$(q(x,y))$ we mean $q(x,y) = \tan^{-1}\frac{\Delta v}{\Delta u}$, where, $\Delta u = f(x+1, y+1) - f(x,y)$, and $\Delta v = f(x+1, y) - f(x, y+1)$. Here $f(x,y)$ is a gray scale at (x, y) point. Finally, blocks and the directional frequencies are down sampled using Gaussian filter to get 392-dimensional feature vector. For details of gradient feature see the paper [15].

3.2 Curvature feature detection

Curvature feature used in this paper has been calculated using bi-quadratic interpolation method and the procedure is as follows:

The curvature c at x_0 in a gray scale image is defined by

$$c = \frac{y^{''}}{\sqrt{(1+y^{'2})^3}} \tag{1}$$

where $y = g(x)$ is the equi-gray scale curve passing though x_0, (x, y) is the spatial co-ordinates of x_0, $y^{'}$ and $y^{''}$ are the first and second order derivative of y, respectively. The derivatives $y^{'}$ and $y^{''}$ are derived from bi-quadratic interpolating surface for the gray scale values in the 8-neighbourhood of x_0. (The eight neighborhood of x_0 is shown in Fig.3. The pixel value of x_k is denoted by f_k. The bi-quadratic interpolated surface is given by

$$z = [1 \quad x \quad x^2]\begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} 1 \\ y \\ y^2 \end{bmatrix} \tag{2}$$

Then the equi-gray curve passing through the point x_0 is given by

$$\begin{aligned}&(a_{22}x^2 + a_{12}x + a_{02})y^2 + (a_{21}x^2 + a_{11}x + a_{01})y \\ &+ a_{20}x^2 + a_{10}x + a_{00} - f_0 = 0\end{aligned} \tag{3}$$

Differentiation of both sides of Eq.3. with respect to *x* and substituting the co-ordinates (0,0) of x_0, the value of $y^{'}$ at x_0 is given by

$$y^{'} = -{a_{10}}\big/{a_{01}} \tag{4}$$

Similarly, considering second derivative of Eq.3. with respect *x*, the value of $y^{''}$ at x_0 is obtained as

$$y^{''} = -{2(a_{10}^2 a_{02} - a_{01}a_{10}a_{11} + a_{01}^2 a_{20})}\big/{a_{01}^3} \tag{5}$$

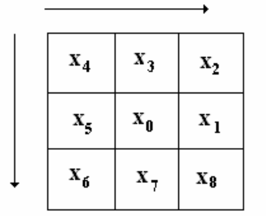
Solving the simultaneous liner equations (2) holding for 8-neighbour of x_0, the coefficients of the bi-quadratic surface are given by

$$\begin{aligned} a_{10} &= (f_1 - f_5)/2, \\ a_{20} &= (f_1 + f_5 - 2f_0)/2, \\ a_{01} &= (f_3 - f_7)/2, \\ a_{02} &= (f_3 + f_7 - 2f_0)/2, \\ a_{11} &= (f_2 - f_8) - (f_4 - f_6)/4 \end{aligned} \tag{6}$$

The coefficients a_{10} and a_{20} are respectively, the first and the second order partial derivatives of $f(x, y)$ with respect to x, a_{01} and a_{02} are similar partial derivatives with respect to y, and a_{11} is the derivative obtained with respect to x and y. Substituting Eqs. 4. and 5. to Eq.1., the curvature is given by

$$c = -2(a_{10}^2 a_{02} - a_{01}a_{10}a_{11} + a_{01}^2 a_{20})/(a_{10}^2 + a_{01}^2)^{\frac{3}{2}} \tag{7}$$

From the Eq.7. it can be noted that the curvature will be indefinite if $a_{10} = a_{01} = 0$. When such situation occurs, we assume curvature is zero in our algorithm.



x_4	x_3	x_2
x_5	x_0	x_1
x_6	x_7	x_8

Figure 3. Eight neighborhood of a pixel, x_0

To get the curvature feature the following steps are applied.

Step 1: The direction of gradient computed in Section 3.1 is quantized to 32 levels with $p\ /16$ intervals.

Step 2: The curvature c computed by the above formula (7) is quantized into 3 levels to get concave, linear and convex regions using a parameter t. For concave region $c \le -t$, for linear region $-t \prec c \prec t$ and for convex region $c \ge t$. We set t as 0.15 in our experiment.

Step 3: The strength of the gradient is accumulated in each of the 32 directions and in each of the 3 curvatures levels of each block to get 49x49 local joint spectra of directions and curvatures.

Step 4: A spatial and directional resolution is made as follows. A smoothing filter [1 4 6 4 1] is used to get 16 directions from 32 directions. On this resultant image, another smoothing filter [1 2 1] is used to get 8 directions from 16 directions. Further more, we use a 31 x 31 two-dimensional Gaussian-like filter (See Fig.4) to get smoothed 7 × 7 blocks from 49 x 49 blocks (shown in Fig.5). So, we get 7×7×8 = 392 dimensional feature vector. Using curvature feature in 3 levels we get 392 × 3 =1176 dimensional features.

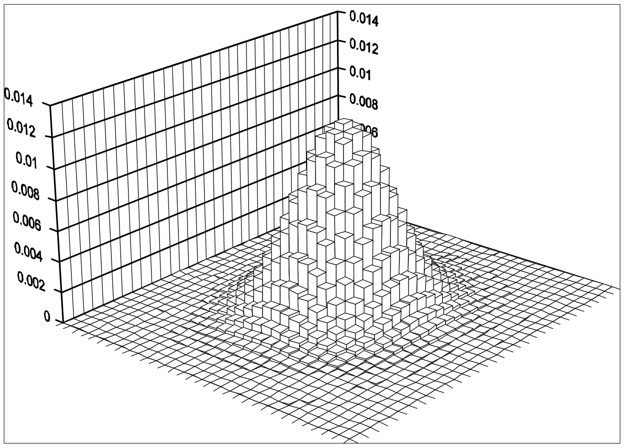


Figure 4. Example of 31 x 31 two-dimensional Gaussian-like filter used for smoothing.

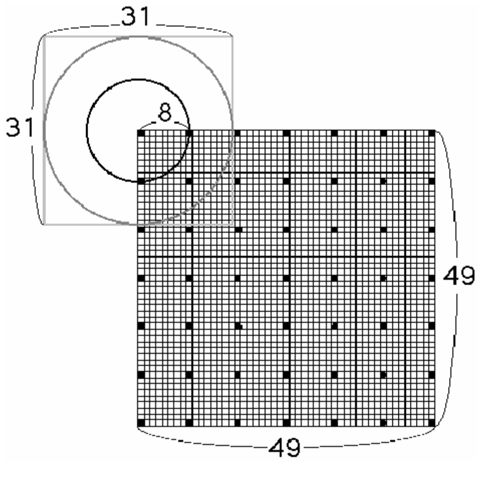


Figure 5. Illustration of getting 7 x 7 blocks from 49 x 49 blocks.

Step 5: Using PCA we reduce 1176-dimensional feature vector to 392-dimensional feature vector and we fed this 392-dimensional feature vector to the SVM classifier.

4. Character recognition

Two classifiers (MQDF and SVM) have been used for our experiment and their details are as follows. To get higher accuracy of the system here we combine both MQDF and SVM in this work. Here at first 392-dimensional gradient features are used in the MQDF. To get 99% reliability in the MQDF classifier we rejected 16.84% samples in MQDF and the curvature features of these rejected samples are fed to Support Vector Machine (SVM) for their recognition. As a result, we obtained better performance from the proposed scheme.

4.1 Modified Quadratic Discriminant Function

Modified Quadratic Discriminant Function is defined as follows [5-6].

$$D(X) = (N + N_0 + n - 1)\ln[1 + \frac{1}{N_0 s^2}[\|X - M\|^2$$
$$- \sum_{i=1}^{k} \frac{l_i}{l_i + \frac{N_0}{N} s^2}\{\Phi_i^T(X - M)\}^2]] + \sum_{i=1}^{k} \ln(l_i + \frac{N_0}{N} s^2)$$

where X is the feature vector of an input character; M is a mean vector of samples; Φ_i^T is the ith eigen vector of the sample covariance matrix; λ_i is the ith eigen value of the sample covariance matrix; k is the number of eigen values considered here; n is the feature size; σ^2 is the initial estimation of a variance; N is the number of learning samples; and N_o is a confidence constant for σ and N_0 is considered as 3N/7 from the experiment. We did not use all the eigen values and their respective eigen vectors for the classification. Here, we sorted the eigen values in descending order and took first 120 (k=120) eigen values and their respective eigen vectors for classification. Compromising trade-off between accuracy and computation time, we decided k as 120.

4.2 Support Vector Machine

An SVM is defined for two-class problem and it finds the optimal hyper-plane which maximizes the distance, the *margin*, between the nearest examples of both classes, named *support vectors* (SVs). Given a training database of M data: $\{x_m | m=1,...,M\}$, the linear SVM classifier is then defined as:

$$f(x) = \sum_j a_j x_j \cdot x + b$$

where $\{x_j\}$ are the set of support vectors and the parameters a_j and b have been determined by solving a quadratic problem [16].

The linear SVM can be extended to a non-linear classifier by replacing the inner product between the input vector x and the SVs x_j, to a kernel function k defined as: $k(x,y) = f(x) \cdot f(y)$. This kernel function should satisfy the Mercer's Condition [16]. There are many kernels and some examples of common kernel functions are Gaussian kernel $[k(x,y) = \exp(-\frac{\| x-y \|^2}{2s^2})]$, polynomial kernel $[k(x,y) = (x.y)^p]$, Tangent Hyperbolic $[k(x,y) = \tanh(x.y - q)]$, etc. In our work we have used Gaussian kernel because it shows highest performance in our experiment. Besides optimizing the kernel parameters (such as σ in a Gaussian kernel), one should consider trade-off parameter C. This parameter indicates how severely errors have to be punished. If the errors are punished too much, the SVMs can over fit the training data. From the experiment we considered C= 1000.0 and σ=5.0 for the best performance in our classification scheme. Details of SVM can be found elsewhere [1,2,16] so we are not giving its details here.

5. Result and discussion

Data used for the present work were collected from different individuals. We tested 36172 samples of Devnagari basic characters (vowels as well as consonants) for the experiment. We have used 5-fold cross validation scheme for recognition result computation. Here database is divided into 5 subsets and testing is done on each subset using rest of the subsets for learning. The recognition rates for all the test subsets are averaged to calculate recognition accuracy.

5.1 Results on Individual classifiers

We tested MQDF and SVM classifiers separately on gradient features and curvature features to compute recognition accuracy on these 36172 samples of Devnagari basic characters. The results are given in Table 1. We obtained 94.24% (94.66%) accuracy from MQDF classifier when 392-dimensional gradient (curvature) features were used and no rejection was considered. It can be noted that curvature features gave better accuracy than gradient feature and this is perhaps because of the presence of the curve-like parts of many Devnagari characters. From the Table 1. It can be noted that SVM classifier shows lower accuracy than MQDF classifier on gradient features. Also in can be noted from the table that SVM classifier shows higher accuracy than MQDF classifier on curvature features.

5.2 Results on combined classifier

As mentioned earlier, we combine both MQDF and SVM in this work to get higher accuracy. Since MQDF classifier shows higher accuracy than SVM on gradient feature, here initially we used MQDF on the gradient features. To get 99% reliability we rejected 16.84% samples in MQDF and the curvature features of these rejected samples were then fed to SVM for their further recognition. Our SVM classifier on curvature feature can recognize 77.02% of these rejected samples. As a result we obtained 95.13% accuracy from this combined method when zero percent rejection was considered.

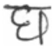
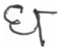
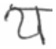
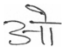
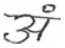
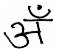
Table 1. Individual accuracy of MQDF and SVM classifiers on gradient and curvature features.

Classifier	Feature used	
	Gradient	Curvature
MQDF	94.24%	94.66%
SVM	94.15%	94.92%

5.3 Confusing pair computation

We also noticed the main confusing pairs of Devnagari characters and their error rates are computed. The characters and have maximum confusion. In the MQDF classifier with gradient feature they confused 0.37% cases whereas after using combined classifier they confused 0.26% cases. The next most confusing pair was and . Confusing pairs of some similar shaped Devnagari characters are shown in Table 2. Here percentage of confusion on MQDF as well as combined classifier (MQDF + SVM) is shown in two separate columns to have the comparative idea.

Table 2. Main confusing pairs of Devnagari characters.

Confusing character pairs		% of confusion on	
		MQDF	MQDF+SVM
		0.37%	0.26%
		0.32%	0.25%
		0.20%	0.14%
		0.17%	0.08%

5.4 Comparison of Results

To the best of our knowledge there exist only three pieces of work on off-line handwritten Devnagari characters and we compared our current results with those three existing pieces of work. Details comparative results are given in Table 3. In our previous work [10] we obtained 94.24%

accuracy from 36172 data and from the same dataset we obtained 95.13% accuracy from the current combined scheme.

Table 3. Comparison of results.

Sl. no.	Method proposed by	Data size	Accuracy obtained
1.	Kumar and Singh [7]	200	80%
2.	Sharma et al.[14]	11270	80.36%
3.	Pal et al. [10]	36172	94.24%
4.	**Proposed method**	**36172**	**95.13%**

6. Conclusion

India is a multi-lingual and multi-script country comprising of eleven different scripts. But not much work has been done towards off-line handwriting recognition of Indian scripts. In this paper we present a combined approach towards the recognition of off-line Devnagari handwritten characters. Here two classifiers (MQDF and SVM) based on gradient and curvature features are combined for the recognition. We tested the proposed scheme on 36172 data and obtained 95.13% accuracy. In future we plan to experiment on other fusion methods to get higher recognition accuracy from our system.

References

[1] C. Burges, "A Tutorial on support Vector machines for pattern recognition", Data Mining and Knowledge Discovery, Vol. *2*, pp.1-43, 1998.

[2] H. Byan and S.W. Lee, "A Survey on pattern recognition application of support vector machines", International Journal of Pattern recognition and Artificial Intelligence, Vol.17, pp 459-486, 2003.

[3] R. El-Hajj, L. Likforman-Sulem and C. Mokbel, "Arabic handwriting recognition using base-line dependent features and Hidden Markov Modeling", *In Proc. 8th International Conference on Document Analysis and Recognition*, 2005, pp.893-897.

[4] M. Hanmandlu and O.V. Ramana Murthy, "Fuzzy Model Based Recognition of Handwritten Hindi Numerals", *In Proc. International Conference on Cognition and Recognition*, 2005, pp. 490-496.

[5] F. Kimura, K. Takashina, S. Tsuruoka and Y. Miyake, "Modified quadratic discriminant function and the application to Chinese character recognition", IEEE Trans. on PAMI, Vol. 9, pp 149-153, 1987.

[6] F. Kimura, T. Wakabayashi, S. Tsuruoka and Y. Miyake, "Improvement of handwritten Japanese character recognition using weighted direction code histogram", Pattern Recognition, Vol.30, pp. 1329-1337, 1997.

[7] S. Kumar and C. Singh, "A Study of Zernike Moments and its use in Devnagari Handwritten Character Recognition", *In Proc. International Conference on Cognition and Recognition*, 2005, pp. 514-520.

[8] U. Pal and B. B. Chaudhuri, "Indian script character recognition: A Survey", Pattern Recognition, Vol. 37, pp. 1887-1899, 2004.

[9] U. Pal, K. Roy and F. Kimura, "A Lexicon Driven Method for Unconstrained Bangla Handwritten Word Recognition", *In 10th International Workshop on Frontiers in Handwriting Recognition*, 2006, pp. 601-606

[10] U. Pal, N. Sharma, T. Wakabayashi and F. Kimura, "Off-Line Handwritten Character Recognition of Devnagari Script", *In Proc. 9th International Conference on Document Analysis and Recognition*, 2007, pp. 496-500.

[11] U. Pal, T. Wakabayashi, N. Sharma and F. Kimura, "Handwritten Numeral Recognition of Six Popular Indian Scripts", *In Proc. 9th International Conference on Document Analysis and Recognition*, 2007, pp. 749-753.

[12] R. Plamondon and S. N. Srihari, "On-Line and off-line handwritten recognition: A comprehensive survey", IEEE Trans on PAMI, Vol.22, pp.62-84, 2000.

[13] I. K. Sethi and B. Chatterjee, "Machine Recognition of constrained Hand printed Devnagari", Pattern Recognition, Vol. 9, pp. 69-75, 1977.

[14] N. Sharma, U. Pal, F. Kimura and S. Pal, "Recognition of Offline Handwritten Devnagari Characters using Quadratic Classifier", *In Proc. Indian Conference on Computer Vision Graphics and Image Processing*, 2006, pp. 805-816

[15] M. Shi, Y. Fujisawa, T.Wakabayashi, and F. Kimura, "Handwritten numeral recognition using gradient and curvature of gray scale images", Pattern Recognition, Vol.35, pp.2051-2059, 2002.

[16] V. Vapnik, *"The Nature of Statistical Learning Theory"*, Springer Verlang, 1995.