Performance Comparison of Devanagari Handwritten Numerals Recognition

Mahesh Jangid Kartar Singh
Department of CSE
Dr. B R Ambedkar NIT
Jalandhar (India)

Renu Dhir
Department of CSE
Dr. B R Ambedkar NIT
Jalandhar (India)

Rajneesh Rani
Department of CSE
Dr. B R Ambedkar NIT
Jalandhar (India)

ABSTRACT

In this paper an automatic recognition system for isolated Handwritten Devanagari Numerals is proposed and compared the recognition rate with different classifier. We presented a feature extraction technique based on recursive subdivision of the character image so that the resulting sub-images at each iteration have balanced numbers of foreground pixels as possible. Database, provided by Indian Statistical Institute, Kolkata, have 22547 grey scale images written by 1049 persons and obtained 98.98% highest accuracy with SVM classifier. Results are compared with KNN and Quadratic classifier.

General Terms

Pattern Recognition, Classification, Preprocessing

Keywords

Devanagari Numeral, Indian Script, SVM (Support Vector Machine), KNN, Quadratic

1. INTRODUCTION

Handwriting is one by which the people communicate with each other from the long time. Letter is an example of such type of communication. Today the e-mailing is growing very fast but the Postal Letters have its own importance. Optical character recognition is a field of automatic recognition of different characters from a document image. This field is divided into two parts one is recognition of machine printed characters and second is recognition of handwritten characters. Now a day's, recognition of handwritten characters is very challenging task because different people have different handwriting styles and. So, handwritten OCR is still a subject of active research.

Devanagari script is the most widely used Indian script and round 500 million people use it. Recognizing Handwritten Numerals have numerous applications including those in postal sorting, bank cheque processing, job application form sorting and automatic scoring of tests containing multiple choice questions.

Techniques used in OCR system follow mainly two steps (a) a feature vector is form from character image (b) classify the feature vector into classes. The feature extraction method plays very important role to achieve high accuracy. So the feature extraction algorithm must be capable to generate similar feature sets for a variety of instance of the same symbol. Ivind and jain [5] present a survey of various feature extraction methods used in character recognition. On the other hand, choice of classifier, to discriminate given features, is not an easy task because of classifier depends on training set, number of free parameters etc.

In the literature survey we found that numbers of authors have attempted to recognize the Handwritten Devanagari Numerals with different-2 techniques. G S Lehal and Nivedan Bhatt [10] proposed a contour extraction technique and obtained 89% accuracy. Reena Bajaj et al. [9] employed three different kinds of feature namely, density features, moment features and descriptive features for classification of Devanagari Numerals and obtained 89.68% accuracy. R J Ramteke et al [11] proposed a method based on invariant moments and the divisions of image for the recognition of numerals and achieved 92% accuracy. U.Bhattacharya et al. [13] used a combination of ANN and HMM classifier on 16273 samples and obtained 95.64 % accuracy. N Sharma et al. [12] have proposed a quadratic qualifier based technique and used 22546 samples for his experiment and achieved 98.86% accuracy.

The technique proposed in the paper is first time applied on the Devanagari Numerals. Feature extraction method based on structure of the character image and the topological and geometrical properties of the character. In our work, the idea of recursive subdivision of the handwritten character image as in [14, 17, 19] is used as a way of extract the features which based on different levels of granularity. At each level, features are extracted based on the point, at the intersection of the horizontal and vertical lines, which divides the handwritten character image into four sub-images that consist of about the same amount of foreground pixels. The process of division of the image gives 4, 16… sub-images. Initially at each level, the feature is calculated and achieved recognition rate choose the level at which the highest recognition rate is achieved.

2. DATABASE

The database is provided by the ISI (Indian Statistical Institute, Kolkata) [18, 24]. Initially Devanagari script was developed to write Sanskrit but was later adapted to write many other languages such as Hindi Marathi and Nepali. The printed Devanagari Numerals are shown in figure 1 and it is seen that there are variations in the shapes of numerals 5, 8 and 9 in their printed forms. In figure 2, there are shown the samples of the Handwritten Devanagari Numerals database. The distributions of training data and testing data are shown in table 1.

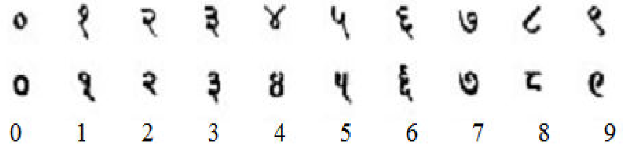


Figure 1: Devanagari Numerals

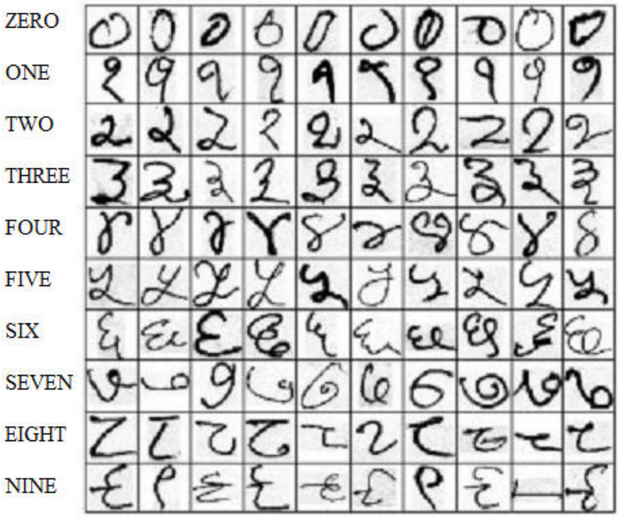


Figure 2: Handwritten Devanagari Numerals Samples

Table 1: Distribution of numerals in Devanagari Database

Digits	Training Set	Test Set	Total
0	1844	369	2213
1	1891	378	2269
2	1891	378	2269
3	1882	377	2259
4	1876	376	2252
5	1889	378	2267
6	1869	374	2243
7	1869	378	2247
8	1887	377	2264
9	1886	378	2264
	18784	3763	22547

3. PROPOSED MOTHED

The entire database is gray scale image that contains noise and is not in normalized form. All experiments have done on Matlab 7.10.0.

3.1 Preprocessing

i) Adjust image intensity values of the image using imadjust () function of Matlab.

ii) Convert the image into binary image by choosing threshold value 0.8.

iii) Remove from a binary image all connected components (objects) that have fewer than 30 pixels

iv) Apply median filtering, is a nonlinear operation often used in image processing to reduce "salt and pepper" noise

v) Normalized the image into 90*90

3.2 Feature Extraction Algorithm

Suppose that im(x, y) is a handwritten character image in which the foreground pixels are denoted by 1’s and background pixels are denoted by 0’s. Feature extraction algorithm sub-divided the character image recursively. At granularity level 0 the image divided into four parts and gives a division point (DP) (x_0, y_0). The following algorithm shows that how x_0 is calculated and likewise y_0.

Algorithm:

Step 1: input im (x_{max}, y_{max}) where x_{max} and y_{max} be the width and the height of the character image

Step 2: Let v_0 [x_{max}] be the vertical projection of image (fig 3.b)

Step 3: Create v_1 [2*x_{max}] array by inserting a ‘0’ before each element of v_0 (fig 3.c)

Step 4: Find x_q in v_1 that minimizes the difference between the sum of the left partition [1, x_q] and the right partition [x_q, 2*x_{max}] or left partition should be greater than right if not able to equally divide.

Step 5: $x_0 = x_q/2$;

Step 6: if x_q mod 2 = 0

2 sub-images are

[(1, 1), (x_0, y_{max}) and (x_0, 1), (x_{max}, y_{max})]

Else

2 sub-images are [(1, 1), (x_0, y_{max}) and

(x_0+1, 1), (x_{max},y_{max})]

Figure 3 shows the vertical division of handwritten character image where the x_q=10 and x_0=5 and x_q mod 2 is 0 than the co-ordinates of two sub-images are [(1,1),(5,8)] and [(5,1),(8,8)]. Another example of an image have the x_q mod 2 = 1 is demonstrated in figure 4.

The number of sub-images, at the specified granularity level (L) will be $4^{(L+1)}$. Let L=0 then the number of sub-images are four and when the L=1 it will be 16. The number of DP (division point) equals to 4^L (figure 5). At level L, the co-ordinates (x_i, y_j) of all DPs are stored as features. So for every L a $2*4^L$-dimensional feature vector is extracted.

All feature vectors are scaled to (0, 1), by the help of normalized dimension value in our case it is 90. All the co-ordinates of feature vector are divided by 90.

$$f'' = f / 90 \qquad (1)$$

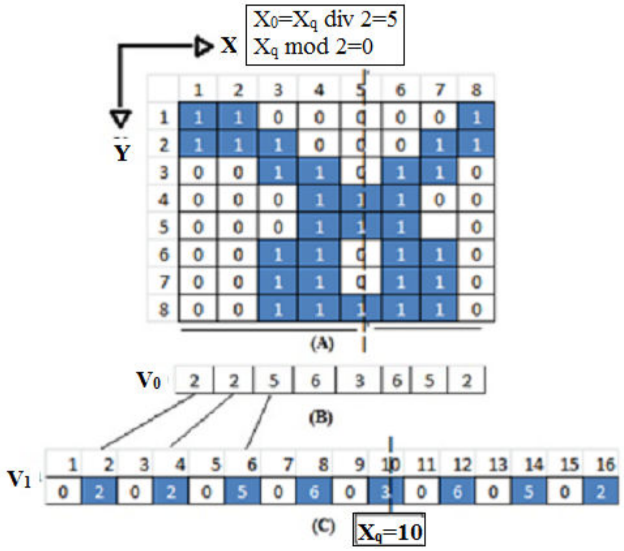


Figure 3(a) Vertical division of an image array (x_{max}=8, y_{max}=8) (b) vertical projection of image (c) v_1 created from v_0 to calculate x_q

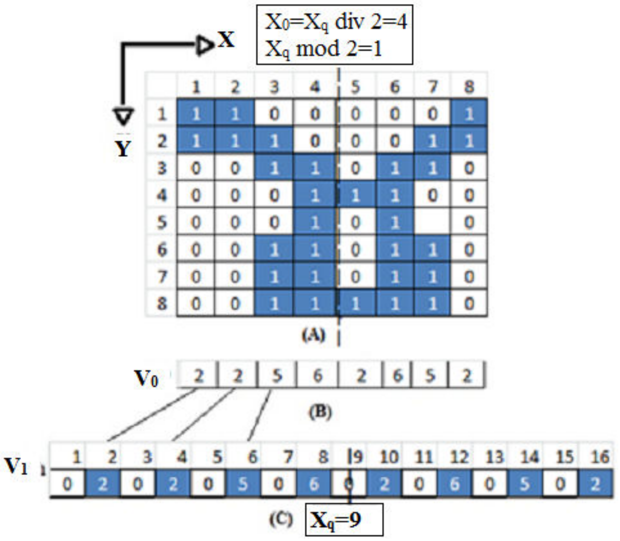
3.3 Classification

Classification step is divided into two phases.

3.3.1 Training phase

In this phase, gradually increase the higher levels of granularity starting with level 1, features are extracted. The recognition rate is calculated at particular level and drawn a graph (figure-6) that shows the level of granularity and the recognition rate. By the help of graph examine the highest recognition rate at corresponding level (L_{best}).

Figure 4 : Example where the x_q mod 2 = 1



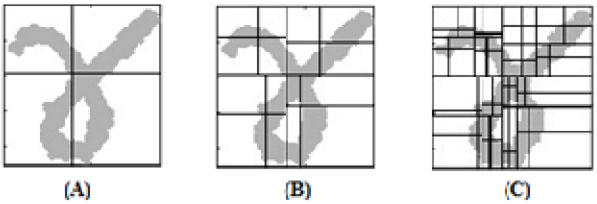


Figure 5: Devanagari Handwritten Numeral 4 segmentation at Level 0, 1, 2 shown in corresponding (A) (B) and (C)

3.3.2 Recognition phase

After the examining the best level of granularity, Test dataset feature, extracted at L_{best}, is fed to the classifier. The classifier recognizes the test dataset by the help of training dataset.

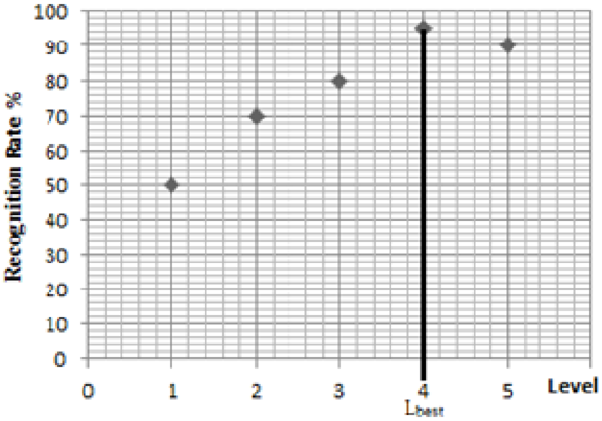


Figure 6: Example finding the best level (L_{best})

4. CLASSIFIERS

4.1 KNN

K-Nearest-Neighbor (KNN) classifier: Nearest neighbor classifier is an effective technique for classification problems in which the pattern classes exhibits a reasonably small degree of variability. The k-NN classifier is based on the assumption that the classification of an instance is most similar to the classification of other instances that are nearby in the vector space. It works by calculating the distances between one input patterns with the training patterns. A k-Nearest-Neighbor classifier takes into account only the k nearest prototypes to the input pattern, and the majority of class values of the k neighbors determine the decision. In the k-Nearest neighbor classification, we compute the distance between features of the test sample and the features of every training sample. The class of majority among the k-nearest training samples is based on the minimum distance criteria.

4.2 Quadratic

A quadratic discriminant function (QDF) of an n-dimensional feature vector is given as

$$g_0^{(l)}(x) = (x-\mu^{(l)})^t \{\Sigma^{(l)}\}^{-1}(x-\mu^{(l)}) + \log|\Sigma^{(l)}| - 2\log P(w^{(l)}) \qquad (2)$$

for a class $\omega^{(l)}$ where $\mu^{(l)}$ and $\sum^{(l)}$ denote the mean vector and the covariance matrix for x in the class $\omega^{(l)}$, respectively, and $P(\omega^{(l)})$ is the a priori probability for the class $\omega^{(l)}$. The QDF becomes optimal in the Bayesian sense for normal distributions with known parameters. In this case, the QDF has the following properties: 1) optimality-the QDF achieves the minimum mean error probability, and 2) monotonicity-the average error rate1 of the QDF decreases monotonically with an increase of the feature size.

4.3 SVM

Support Vector Machine is supervised Machine Learning technique. It is primarily a two class classifier. Width of the margin between the classes is the optimization criterion, i.e. the empty area around the decision boundary defined by the distance to the nearest training pattern [23]. These patterns called support vectors, finally define the classification function.

LIBSVM [22] implements the “one against one” approach (Knerr et al .., 1990) [20] for multi-class classification. Some early works of applying this strategy to SVM include, for example, Kressel (1998) [21]. If k is the number of classes, then k (k-1)/2 classifiers are constructed and each one trains data from two classes. For training data from the ith and jth classes, we solve the following two class classification problem:

In classification we use a voting strategy: each binary classification is considered to be a voting where votes can be cast for all data points x - in the end a point is designated to be in a class with the maximum number of votes. In case that two classes have identical votes, though it may not be a good strategy, now we simply choose the class appearing first in the array of storing class names.

LIBSVM is used with Radial Basis Function (RBF) kernel, a popular, general-purpose yet powerful kernel, denoted as

$$K(x_i, x_j) \square \exp(-\gamma||x_i-x_j||^2) \qquad (3)$$

Now a search is applied to find the value of γ which is parameter of RBF as like find the value of c that is cost parameter of SVM using cross-validation. The value of both variance parameters are firstly select in the range of (0, 2] and (0, 1000] and examines the recognition rate.

5. EXPERIMENTS AND RESULTS

In order to classify the handwritten numeral and evaluate the performance of the technique, we have carried out the experiment by setting various parameters, examples L_{best}, gamma, and cost parameter for SVM.

The training set of Devanagari Handwritten Numerals provided by ISI, Kolkata contains 18784 samples used to determinate the best granularity level. Here to recognition accuracy at different granularity level used cross validation function of LIBSVM with n=10 and set the γ =0.5 and c=500. The recognition accuracy at different-2 granularity level shows in fig 7. At level 3, the highest accuracy 98.98 obtained.

After obtaining the best granularity level by experiment 1, train the LIBSVM by ISI training set. The size of feature vector is 170 (2*4L – 2*40+2*41+2*42+2*43). Some granularity level applies on the test data to form the feature vector and obtained the 98.49 % accuracy when values of γ, c set to 1.1 and 500.

Same testing and testing dataset is apply on the KNN and Quadratic classifiers and record the accuracy results that is shown the table 3. Figure 8 shows the wrongly identified samples by different classifier. SVM is very affective classifier than others.

Table 3: Shows the accuracy by different classifiers

S.No.	Classifier Name	Accuracy Obtained
1	Quadratic	94.65%
2	KNN-(Correlation and K=3)	96.65%
3	SVM (γ=1.1 and c=500)	98.49 %

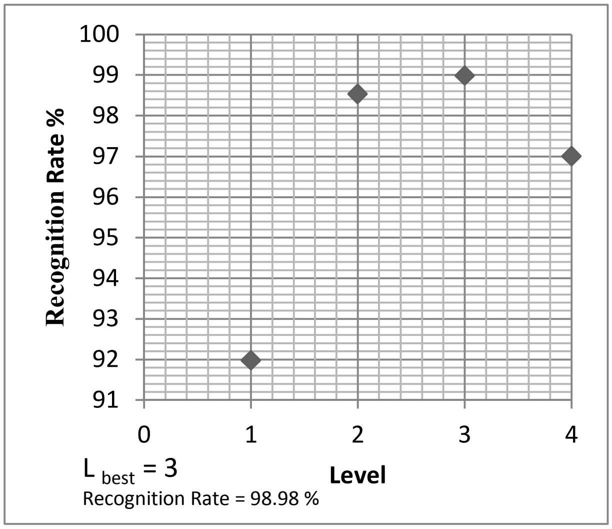


Figure 7: Define the best level for accuracy

Test dataset and training dataset have been combined to perform the cross validation function of LIBSVM with n=10 and set the γ =0.5 and c=500. Features vector for whole dataset (22546) is calculated at level 3 (L_{best}) and obtained 98.98% recognition rate.

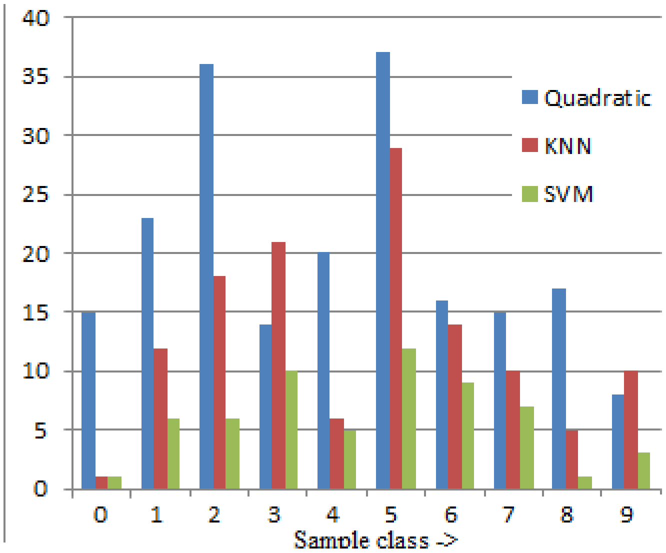


Figure 8: shows the wrongly identified samples by classifiers

Table 4: Comparison of Numerals Results by Researchers.

S.No	Method proposed by	Data Size	Accuracy Obtained
1	R. Bajaj et al [9]	400	89.6 %
2	R. J. Ramteke et al [11]	169	92.68 %
3	U. Bhattacharya et al. [13]	16274	95.64 %
4	N. Sharma et al. [12]	22,546	98.86 %
5	Proposed System	22,547	98.98 %

6. CONCLUSION & FUTURE WORK

In the literature, we found the many techniques apply to recognition of Devanagari Handwritten Numerals. In this paper an effort make towards to get higher accuracy and obtained 98.98% on the database, which have approximately all the variation occurred in handwritten character, provided by ISI, Kolkata. We have compared the algorithm with different-2 classifier. SVM done its work very perfectly as compare to KNN and Quadratic classifier.

This technique is very successful for the Devanagari Handwritten Numerals. So we can apply on the Devanagari Character.

7. REFERENCES

[1] Satish Kumar, "Performance and Comparison of Features on Devanagari Hand-printed Dataset", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009.

[2] U. Pal, T. Wakabayashi, N. Sharma and F. Kimura, "Handwritten Numeral Recognition of Six Popular Indian Scripts", Proc. 9th ICDAR, Curitiba, Brazil, Vol.2 (2007), 749-753.

[3] G. G. Rajput, S. M. Mali," Fourier Descriptor based Isolated Marathi Handwritten Numeral Recognition", International Journal of Computer Applications, Volume 3 –No.4, June 2010.

[4] U. Bhattacharya and B. B. Chaudhuri, "A Majority Voting Scheme for Multiresolution Recognition of Hand printed Numerals", Seventh International Conference on Document Analysis and Recognition (ICDAR 2003).

[5] Ivind due trier, anil Jain, torfiinn Taxt, "A feature extraction method for character recognition-A survey ", pattern Recg, vol 29, No 4, pp-641-662,1996

[6] Sandhya Arora, Debotosh Bhattacharjee, Mita Nasipuri, D. K. Basu, M. Kundu, " Recognition of Non-Compound Handwritten Devnagari Characters using a Combination of MLP and Minimum Edit Distance", International Journal of Computer Science and Security (IJCSS),Volume (4) : Issue-1 pp 107-120.

[7] P M Patil, T R Sontakke," Rotation, scale and translation invariant handwritten Devanagari numeral character recognition using general fuzzy neural network", Pattern Recognition, Elsevier, 2007.

[8] Anil K. Jain, Robert P.W. Duin, and Jianchang Mao,"Statistical Pattern Recognition: A Review", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, pp- 4-37, January 2000.

[9] Reena Bajaj, Lipika Day, Santanu Chaudhari, "Devanagari Numeral Recognition by Combining Decision of Multiple Connectionist Classifiers", Sadhana, Vol.27, Part-I, 59-72, 2002.

[10] G S Lehal and Nivedan Bhatt, "A Recognition System for Devnagri and English Handwritten Numerals", Proc. Of ICMI, 2000.

[11] R.J.Ramteke and S.C.Mehrotra, "Recognition Handwritten Devanagari Numerals", International journal of Computer processing of Oriental languages, 2008.

[12] U. Pal, T. Wakabayashi, N. Sharma and F. Kimura, "Handwritten Numeral Recognition of Six Popular Indian Scripts", Proc. 9th ICDAR, Curitiba, Brazil, Vol.2 (2007), 749-753.

[13] U. Bhattacharya, S. K. Parui , B. Shaw, K. Bhattacharya, "Neural Combination of ANN and HMM for Handwritten Devnagari Numeral Recognition".

[14] J. Park, V. Govindaraju, S. N. Shrihari, "OCR in Hierarchical Feature Space", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, Vol. 22, No. 24, pp. 400-408.

[15] Samet H, “The Design and Analysis of Spatial Data Structures", Addison-Wesley Longman Publishing Co., Inc., 1990.

[16] S. Mozaffari, K. Faez, M. Ziaratban, "Character Representation and Recognition using Quadtree-based Fractal Encoding Scheme ", Proceedings of the 8th International Conference on Document Analysis and Recognition,Seoul, Korea, 2005, Vol.2, pp. 819-823.

[17] A. P. Sexton, V. Sorge, "Database-Driven Mathematical Character Recognition", Graphics Recognition, Algorithms and Applications (GREC), Lecture Notes in Computer Science (LNCS), Hong Kong, 2006, pp. 206-217.

[18] U. Bhattacharya and B.B. Chaudhuri, “Databases for Research on Recognition of Handwritten Characters of Indian Scripts,” Proc. Eighth Int□ Conf. Document Analysis and Recognition (ICDAR □05), vol. 2, pp. 789-793, 2005.

[19] Georgios Vamvakas, Basilis Gatos, Stavros J. Perantonis,” Handwritten character recognition through two-stage foreground sub-sampling”,” Pattern Recognition,” 43 (2010) 2807–2816

[20] S. Knerr, L. Personnaz, and G. Dreyfus. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In J. Fogelman, editor, Neurocomputing: Algorithms, Architectures and Applications. Springer-Verlag, 1990.

[21] U. H.-G. Kressel. Pairwise classication and support vector machines. In B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, Advances in Kernel Methods { Support Vector Learning, pages 255{268, Cambridge, MA, 1998. MIT Press.

[22] LIBSVM -- A Library for Support Vector Machines, available online: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[23] Chih-Chung Chang and Chih-Jen Lin, LIBSVM -- A Library for Support Vector Machines, National Taiwan University, Taipei, Taiwan, http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf

[24] Handwritten Character Databases of Indic Scripts, ISI Kolkata website http://www.isical.ac.in/~ujjwal/download /database.html