Script Identification of Pre-Segmented Multi-Font Characters and Digits

Rajneesh Rani
Department of Computer Science and Engineering
National Institute of Technology
Jalandhar, Punjab ,India
Email: ranir@nitj.ac.in

Renu Dhir
Department of Computer Science and Engineering
National Institute of Technology
Jalandhar, Punjab ,India
Email: dhirr@nitj.ac.in

Gurpreet Singh Lehal
Department of Computer Science and Engineering
Punjabi University
Patiala, Punjab ,India
Email: gslehal@gmail.com

***Abstract*—Character recognition problems of distinct scripts have their own script specific characteristics. The state-of-art optical character recognition systems use different methodolgies, to recognize different script characters, which are most effective for the corresponding script. The identificaton of the script of the individual character has not brought much attention between researchers, most of the script identification work is on document, line and word level. In this multilingual/multiscript world presence of different script characters in a single document is very common. We here propose a system to encounter such adverse situation in context of English and Gurumukhi Script. Experiments on multifont and multisized characters with Gabor features based on directional frequency and Gradient features based on gradient information of an individual character to identify it as Gurumukhi or English and also as character or numeral are reported here. Treating it as four class classification problem, multi-class Support Vector Machine(One Vs One) has been used for classification. We got promising results with both types of features.The average identification rates obtained with Gabor and Gradient features are 98.9% and 99.45% respectively.**

I. Introduction

The characters of distinct scripts have different properties It makes trouble for optical characeter recognition(OCR) of multiscript mixed documents. A common strategy for all type of characters is not suitable. Identifying type of character and utilizing distinct strategies and methods corresponding to type of charater is helpful to improve OCR performance.

Generally, this technique belongs to script identification. Most of the script identification reseraches are based on page/paragraph level, line level, word level and character level [1] and [2]. For multiscript/multilingual country India, documents containing more than one indian script is very common at distinct levels. Script identification has been discussed at paragraph level in [3] and [4] for Indian documents. Line level script identification for some Indian languages has been explained in [5], [6] and [7]. Recognition of word-wise Indian and Roman scripts using various techniques have been explained in [8], [9], [10], [11], [12], [13] and [14]. Most of the Indian documents have individual text lines mixed with English words and Numerals [15]. Word segmentation usually breaks words into characters for multi-script documents containing one script as English. The reason is that the characters in a word in some Indian scripts like Gurmukhi, Devanagri and Bangla are all joined and in English some intercharacter gap is present in a word. For such sitautions, character level script identification will work better than word level script identification. This motivates to identify the script of individual character. Script identification at character level for Indian scripts is seldom discussed.

In this paper, a technique for script identification at character level is proposed that consist of English and Gurmukhi characters and digits. English character set of 52 characters(26 Uppercase letters and 26 Lowecase letters), Gurmukhi character set of characters and 10 digits of each script are shown in Figure 1. In this paper, technique to identify the script of a presegmented character and also to identify it as character or digit is proposed. A comparative study of two different types of features using different kernel functions of SVM classifier is discussed. Experiments are done to check the font type and font size dependency in the training dataset.

The organization of the paper is as follows: the algorithims to extract Gabor and Gradient features are explained in section II. Classifier details are given in section III. Experimental results are provided in section IV followed by conclusion and future perspectives in section V.

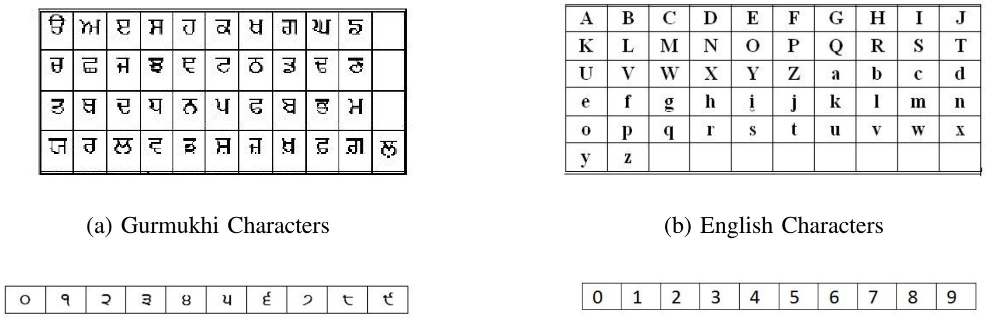
II. Feature Extraction

Here, we use two sets of features. The first set of 189 features is gabor filter based features and second set of 200 features is gradient based features. The introduction and computation methods of these two types of features is given as follows:

A. Gabor Feature Extraction

Gabor filters are mostly used as a directional frequency extractor. These filters effectively capture the concentration of energies in various directions [12]. Distinct scripts have different features along various directions. Gabor filters corresponding to these directions can give good response, considering this, we explore the use of Gabor filters for script recognition of Gurmukhi and Roman characters by dividing the image into different zones. A Gabor filter is a linear filter whose impulse response is defined by a harmonic function multiplied by a Gaussian function as given in the following equation

$$h(x,y) = g(x,y)s(x,y)$$



(a) Gurmukhi Characters

(b) English Characters

(c) Gurmukhi Digits

(d) English Digits

Fig. 1: Characters and Digits of Gurumukhi and English Script

where s(x,y)is a complex sinusoid, known as carrier and g(x,y) is a Gaussian shaped function, known as envelope. Thus the 2D Gabor filter with orientation θ and centerd at frequency f can be written as in equation

$$h_{x,y,\theta,f} = \exp^{-\frac{1}{2}(\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2})} \exp^{j2\pi fx} \tag{1}$$

where $\sigma_x$and σ_y the spatial spread and are the standard deviations of the Gaussian envelope along x and y direction and x' and y' are defined as:

$$x' = x\cos(\theta) + y\sin(\theta) \qquad y' = y\cos(\theta) - x\sin(\theta)$$

Steps for Gabor Feature Extraction

1) Normailze the given character Image into size 32 $\times$32 and call it A.
2) Divide the normalized image into four equal non overlapping subregions and call it A_1, A_2, A_3andA_4.
3) Divide further each subregion into four equal non overlapping sub-subregions and call it$A_{11}...A_{14}$, $A_{21}..A_{24}$, $A_{31}..A_{34}$ and $A_{41}..A_{44}$ and thus obtain 16 small regions in different parts of the image.
4) Filter each of these twenty one images(one obtained in step 1, four in step 2 and sixteen in step 3) with gabor filter given in Equation 1 in nine different angles of orientation (0, $\pi/9$,$2\pi/9$, $3\pi/9$, $4\pi/9$, $5\pi/9$, $6\pi/9$, $7\pi/9$, $8\pi/9$). Calculate Radial frequency for each image as 2 divided by n, for image of size n$\times n$.
5) Sum-square to evaluate the energy content of these images and normalize this energy by dividing it by the size of each corresponding image region. Thus each region gives nine features, So, a feature vector of size 189 (21 $\times$9) is obtained.

B. Gradient Feature Extraction

The gradient measures the magnitude and direction of the greatest change in intensity in a small neighbourhood of each pixel. Gradient Vector $[G_x, G_y]$ at a pixel (i, j) of the image, where G_x and G_y are the horizontal and vertical gradient components, is determined by convolving input image with Sobel operators and thus are given as in Equation 2 and Equation 3.

$$\begin{aligned} G_x(i,j) = I(i+1, j-1) + &\ 2*I(i+1, j) \\ &+I(i+1, j+1) - I(i-1, j-1) \\ &-2*I(i-1, j) - I(i-1, j+1) \end{aligned} \tag{2}$$

$$\begin{aligned} G_y(i,j) = I(i-1, j+1) + &\ 2*I(i, j+1) \\ &+I(i+1, j+1) - I(i-1, j-1) \\ &-2*I(i, j-1) - I(i+1, j-1) \end{aligned} \tag{3}$$

The Gradient Strength $|G(i,j)|$ and Direction $theta(i,j)$ can be computed from the Gradient Vector $[G_x, G_y]$ as shown below

$$|G(i,j)| = \sqrt{((G_x(i,j))^2 + (G_y(i,j))^2} \tag{4}$$

$$theta(i,j) = \tan^{-1}\frac{G_y(i,j)}{G_x(i,j)} \tag{5}$$

A Gradient Feature Vector is composed of the strength of gradient accumulated separately in different directions. The Gradient Feature Vector used in this research approach comprises of 200 features per character image.

Steps for Gradient Feature Extraction

1) Normailze the given character image into size 63 $\times$63 and call it A.
2) Convolve the input image (I) with Sobel Masks to calculate horizontal and vertical components of gradient vector.
3) For every pixel of input image (I), calculate gradient direction and gradient magnitude as in Equation 4 and 5.
4) For every pixel (i, j), determine the two gradient directions from eight chaincode directions, in which the gradient vector of that pixel lies and decompose the gradient vector along these two directions.
5) Compute gradient directional matrix of size 63 $\times$63 such that each element of this matrix would correspond to a directional row vector for each pixel of input image I and each element of the directional row vector of size 8 would correspond to value of gradient component along the 8 chaincode directions.

6) Divide gradient directional matrix horizontally and vertically into subblocks of size (7 × 7) to get 81 (9 × 9) blocks.
7) For each block, sum together the directional vector of each pixel in that block to form a directional vector of that block.
8) Perform downsampling on number of blocks from 9 ×9 to 5 ×5 using Gaussian filter of size 5×5 to produce a feature vector of size 200 (5 horizontal, 5 vertical, 8 directional resolution).
9) Apply variable transformation $(y = x^{0.4})$ to feature set to make the distribution of the features Gaussian-like. Thus feature vector of size 200 (5 horizontal, 5 vertical, 8 directional resolution)is formed.

III. Classification

The main task of classification is to use the feature vectors provided by feature extraction algorithm to assign the object/pattern to a category. Support Vector Machine(SVM) is a classification technique successfully used in a wide range of applications.

A. SVM Classifier

Binary (two-class) classification using support vector machines (SVMs) is a very well developed technique to find the optimal hyperplane to maximize the distance or margin between two classes .

Given a training set of instance-label pairs $(x_i, y_i), i = 1, 2, ...l$ where $x_i \epsilon R^n$, i.e. having n features for a particular training sample and $y_i \in \pm 1$, i.e. class label either 1 or -1 for corresponding training instance x_i. If the training data are linearly separable, we can select two hyperplanes in a way that they separate the data and there are no points between them, and then try to maximize their distance. The region bounded by them is called "the margin".The distance between these two hyperplanes is $\frac{2}{\|w\|}$, so $\| w \|$ should be minimium [16] .

If there exists no hyperplane that can split the 'yes' and 'no' examples, the Soft Margin method will choose a hyperplane that splits the examples as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. The objective function is then increased by a function which penalizes non-zero ξ_i and the optimization becomes a trade off between a large margin and a small error penalty.The support vector machines (SVM) require the solution of the following optimization problem, i.e. minimization of error function[17] as given in Eq. 6 :

$$min_{w,b,\xi} \frac{1}{2} W^T W + C \sum_{i=1}^{l} \xi_i \tag{6}$$

subject to the constraints:

$$y_i(W^T \phi(x_i) + b) \geq 1 - \xi_i$$

and

$$\xi_i \geq 0$$

where C is the penality parameter, W is the vector of coefficients, b a constant and ξ_i are parameters for handling non-separable data (inputs). The index i labels the N training cases or instances. Here $y_i \in \pm 1$ are the class labels and x_i are the independent variables. The kernel ϕ is used to transform data from the input (independent) to the feature space.

In testing phase, for a given input pattern x, the decision function of an SVM binary classifier is

$$f(x) = sign(\sum_{i=1}^{n} y_i \alpha_i K(x, x_i) + b) \tag{7}$$

where:

$$sign(u) = \begin{cases} 1 & for \quad u > 0 \\ -1 & for \quad u < 0 \end{cases}$$

b is the bias, α_i is the langrage multiplier and $K(x, x_i)$ is the kernel function.There are several number of kernels used in support vector machines. Some of the popularily used kernel functions are:

- Linear Kernel:
$$K(x, x_i) = x^T x_i \tag{8}$$
- Polynomial Kernel:
$$K(x, x_i) = (x^T x_i + 1)^d \tag{9}$$
where d is the degree of polynomial.
- Gaussian (RBF)Kernel:
$$K(x, x_i) = \exp(-\gamma * \| x - x_i \|^2) \tag{10}$$
where $\gamma = (1/2\sigma^2)$ and σ is the standard deviation of the x_i values.

The solution of multi (More than two classes) classification is by combining several binary classifiers.There are two approaches for combining binary SVM classifier: "One versus All" (OVA) and "One versus One" (OVO). In our case as it is four class problem, we have used "OVO" approach as it takes less training time as compared to "OVA" [18]

IV. Experimental Setup, Results and Discussions

To evaluate the classifier performance and validate the effectiveness of the proposed techniques, different experiments have been performed which are as follows: (i) Global Script Recognition Accuracy based on ten fold cross validation on the total dataset. (ii) Script Recognition Accuracy of characters with Fonts not present (Fonts) in Training Dataset (iii) Script Recognition Accuracy of characters with Font Sizes not present in Training Dataset. A brief on our dataset can be found in subsection A.

A. Dataset Details

To the best of our knowledge, there is no publicly available database suitable for our defined problem (script identification of multi font characters). We developed our own dataset with different fonts and size characters to show the efficency of proposed features. The printed documents of isolated characters having different font and size are scanned at a resolution of

TABLE I: Dataset Details

Character Type	No. of Characters	No. of Fonts	No. of Sizes	Total Dataset
Gurmukhi Character	41	14	11	$41\times14\times11 = 6314$
Gurmukhi Digit	10	14	11	$10\times14\times11 = 1540$
English Character	26+26=52	17	11	$52\times17\times11 = 9724$
English Digit	10	17	11	$10\times17\times11 = 1870$
Total Number of characters=19448				

TABLE II: Average Accuracy and Standard Deviation Results for Ten Fold Expeiments

Features Used		Linear	Polynomial	RBF
Average Accuracy	Gabor Features	97.85	98.89	98.9
	Gradient Features	97.17	99.23	99.45
Standard Deviation	Gabor Features	0.22	0.20	0.23
	Gradient Features	0.57	0.20	0.19

300 dpi. Commonly used seventeen fonts (Callibri, Arial, Cambria, ArialRoundedMTBold, Times, TimesNewRoman, Georgia, MicrosoftSansSerif, Comic, CenturySchoolBook, Garamond, Verdana, Helevetica, CourierPS, Patatino, Bookman and NewCenturySchoolBook) for English and fourteen fonts for Gurumukhi (AnmolKalmi, AnmollipiHeavy, Anmollipiilight, AnmollipiThick, GurbaniKalmi, WebAkharSlim, Rajaa5Medium,Amarlipi, AmarlipiHeavy, Amarlipilight, Amarlipislim, Anmollipi, KarmiSanjBook and PunjabiTypewriter) in eleven different Font sizes (10,11,12,14,16,18,20,22,24,26,28) have been considered for the present experiments. Thus the dataset of 19448 characters has been prepared as explained in Table I with 6314 Punjabi characters, 1540 Punjabi numerals, 9724 English characters and 1870 English digits.

B. Global Script Recognition Accuracy

To obtain the recognition results we have used 10-fold cross validation. First we created randomly generated 10-fold cross-validation index of the length of size of dataset. This index contains equal proportions of the integers 1 through 10. These integers are used to define a partition of whole dataset into 10 disjoint subsets. We used one division for testing and remaining nine divisions for training. We did so 10 times, each time changing the testing dataset to different division and considering remaining divisions for training. Thus we got 10 sets of feature vectors containing training and testing dataset in the size ratio of 9:1.

Our experiments are carried out using different kernel functions of SVM classifier with ‘OVO’ approach.The main cause of performance difference among different types of SVM classifiers is linked to feature data distribution. We have tested our results using Linear, Polynomial and Gaussian (RBF) kernel on Gabor and Gradient features as shown in Figure 2. It can be noted that RBF and Polynomial Kernel funcions give better accuracy for both types of features than linear Kernel.That demonstrates these features are non linearly separable. In order to further analysis of classification error depending on the kind of SVM chosen, we have computed the standard deviation and average accuracies obtained from ten subsets as shown in Table II. It can be noted that Gradient features with RBF kernel function showed the maximum average accuracy 99.45% with lowest standard deviation 0.19.

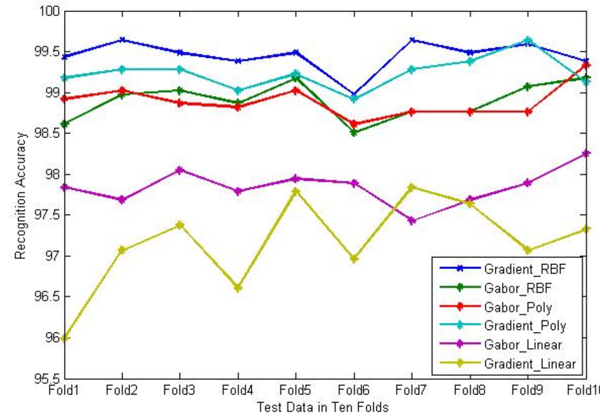


Fig. 2: Accuracy Results during Ten Fold Experiments for Gabor and Gradient Features with Linear, Polynomial and RBF SVM

TABLE III: Confusion Matrix for Script Identificatio with RBF Kernel Function

Method Proposed	Character Type	Gurmukhi Character	English Character	English Digit	Gurmukhi Digit
Gabor	Gurmukhi Character	99.6%	0.22%	0%	0.18%
	English Character	0%	99.41%	0.40%	0.19%
	English Digit	0%	4.66%	95.29%	0.05%
	Gurmukhi Digit	0%	2.81%	0%	97.19%
Gradient	Gurmukhi Character	99.9%	0.05%	0%	0.05%
	English Character	0%	99.56%	0.23%	0.21%
	English Digit	0%	1.77%	98.12%	0.11%
	Gurmukhi Digit	0%	1.48%	0%	98.52%

C. Error Analysis

It has been observed from experiments that script recgnition accuracy of Gurmukhi characters is 99.9% and most of the errors in recognition are for English and Gurmukhi digits which are recognized as English characters. The confusion matrix is shown in Table III. We analyzed the source of errors for the proposed methods. The errors are from some similar symbols in two different character sets. The English and Gurmukhi digit ‘0’ are similar to each other and also to English character ‘O’. The English character ‘I’ is similar to English digit ‘1’.

D. Accuracy with Fonts not present in Training Dataset

In real world applications, the robustness of an algorithim with respect to distict font and size characters is a key factor. To show the efficency of the proposed features, we performed the experiments using different fonts in training and test dataset. This is done by dividing the dataset into two parts: Dataset1 and Dataset2. These datasets have mutually exclusive font characters. Dataset1 has first nine English and first seven Punjabi fonts and Dataset2 has remaining eight English and seven Punjabi fonts.The results of two experiments are given in Table IV where in each experiment one dataset is taken as training and other as testing. It is clear that Gradient with RBF kernel function of SVM classifier has the maximum accuracy that is above 98% in both experiments.

TABLE IV: Results for Fonts not present in Training Dataset

Features Used		Linear	Polynomial	RBF
Experiment 1	Gabor Features	94.60	94.55	96.11
	Gradient Features	93.51	97.46	98.33
Experiment 2	Gabor Features	94.89	96.14	96.47
	Gradient Features	94.32	97.60	98.08

TABLE V: Results for Font Sizes not present in Training Dataset

Features Used		Linear	Polynomial	RBF
Experiment 1	Gabor Features	97.90	99.03	98.76
	Gradient Features	96.89	99.14	99.19
Experiment 2	Gabor Features	96.91	98.02	98.49
	Gradient Features	94.12	97.67	98.89

E. Accuracy with Font Sizes not present in Training Dataset

We also computed the accuracy of our system using different fontt sizes in training and test data set. Now partitioning of the whole dataset id done into Dataset3 containing font sized (10,11,12,14,16 and 18) characters and Dataset4 containing rest of the characters of font sizes (20,22,24,26 and 28). So, recognition results of two experiments, taking each time one as training and other as testing are reported in Table V.

F. Comparison with Earlier Approaches

To the best of our knowledge, this work is the first of its kind on an Indian and English script identification at character level, we can not compare the results. Howeve to get an idea about the recognition accuracy over other existing pieces of work on different scripts, some comparison is given in TableVI.

V. Conclusion and Future Perspectives

Most of the existing script identification techniques for Indian languages are based on whole document, block, line and word level. We focused our research on single character script identification and have got promising results. In this paper, we have proposed SVM based for identification of script of presegmented multifont and multisized characters. By using, Gabor and Gradient features of 19448 characters, the average accuracy obtained is 98.9% and 99.45%. To the best of our knowledge, this is the first work on English and an Indian script identification at character level, which identifies numerals from characters also. As the reported work here is only for Gurmukhi and English, so it can be tested on other Indian and non-Indian scripts also. Post-processing to correct the slight classification errors due to similar characters of different scripts is also a future direction of research.

TABLE VI: Comparison with Earlier Approaches

Method Proposed By	Script Used	Methodolgy	Accuracy
Zhang et al. [19]	Chinese,English	Structural Features and SVM	99.3%
Sanguansat et al.[21]	Thai,English	Hidden Markov Model	99.31%
Zhu et al. [20]	Chinese,English	Feature Selection and Cascade Classifier	99.25%
Proposed Method	Gurmukhi,English	Gabor Features Gradient Features and SVM	98.90% 99.45%

References

[1] D. Ghosh, T. Dube and A. P. Shivaprasad *Script Recognition A Review– IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.32, no.12, pp. 2142-2161, (Dec. 2010).

[2] R.Rani, R. Dhir *A Survey: Recognition of Scripts in Bi-Lingual/Multi-Lingual Indian Documents— National Journal of PIMT Journal of Research* vol. 2,no. 1 pp. 55-60 (Mar- Aug, 2009).

[3] G.D.Joshi, S.Garg and J.Sivaswamy *Script Identification from Indian Documents — Proc. IAPR Intl Workshop Document analysis Systems*, pp. 255-267 (Feb. 2006).

[4] S.Chaudhury and R. Sheth *Trainable Script Identification Strategies for Indian Languages— Proc. Intl Conf. Document Analysis and Recognition* pp. 657-660, (Sept. 1999).

[5] U.Pal and B.B. Chaudhari *Script Line Separation from Indian Multi-Script Documents — Proc. Intl Conf. Document Analysis and Recognition*, pp. 406-409. (Sept. 1999).

[6] U.Pal and B.B. Chaudhuri *Identification of Different Script Lines from Multi-Script Documents— Image and Vision Computing* vol.20, nos. 13, pp. 945-954, (Dec. 2002).

[7] U.Pal, S.Sinha and B.B.Chaudhri *Multi-Script Line Identification from Indian Documents —- Proc. Intl Conf. Document Analysis and Recognition*, pp.880-884, (Aug. 2003).

[8] S.Sinha, U.Pal and B.B.Chaudhri *Word-Wise Script Identification from Indian Documents — Proc. IAPR Intl Workshop Document Analysis Systems* pp. 310-321, (Sept. 2004).

[9] S.Chanda, R.K.Roy and U.Pal *English, Devanagari and Urdu Text Identification— Proc. Intl Conf. Cognition and Recognition*, pp. 538-545, (Dec. 2005).

[10] S.Chanda, R.K.Roy and U.Pal *English and Tamil Text Identification— Proc. Natl Conf. Recent Trends In Information Systems*, pp. 184-187, (July 2006).

[11] M.C.Padma and P.Nagabhushan *Identification and Separation of Text Words of Kannada, Hindi and English Languages through Discriminating Features— Proc. Natl Conf. Document Analysis and Recognition*, pp. 252-260, (July 2003).

[12] P.B. Pati , A.G.Ramakrishnan *Word level multi-script Identification— Pattern Recognition Letters*, pp. 1218-1229,(2008).

[13] B.V.Dhandra, Mallikarjun Hangarge *On Separation of English Numerals from Multilingual Document Images— Journal of multimedia* , vol.2, no. 6, pp. 26-33 (November, 2007).

[14] R. Rani, R. Dhir and G.S.Lehal *Comparative Analysis of Gabor and Discriminating Feature Extraction Techniques for Script Identification — -Proc. of ICISIL, Communications in Computer and Information Science 139* Chandan Singh *et al. Eds.* Springer, (March 2011)

[15] R. Rani, R. Dhir and G.S. Lehal *Performance analysis of feature extractors and classifiers for script recognition of English and Gurumukhi words– Proc. of the Workshop on Document Analysis and Recognition (DAR 2012)* Mumbai, Publisher ACM, USA. pp. 30-36. (2012)

[16] Support Vector Machines (SVM), StatSoft website [Online]. Available at: *http://www.statsoft.com/textbook/support-vector-machines/*

[17] Support Vector Machine, Wikipedia website [Online]. Available at: *http://en.wikipedia.org/wiki/Support-vector-machine*

[18] M. Jonathan, M. Cheriet, and R. Sabourin *One Against One or One Against All: Which One is Better for Handwriting Recognition with SVMs?– Tenth International Workshop on Frontiers in Handwriting Recognition* (2006).

[19] Y. Zheng, C. Liu, X. Ding. *Single character type identification— In Proc. of the SPIE Document Recognition and Retrieval* IX. Vol.4670, pp.49-56, (2002).

[20] Y. Zhu, J. Sun, A. Minagawa, Y. Hotta, S. Naoi*Separate Chinese Character and English Character by Cascade Classifier and Feature Selection, In Proc. of 10th Int'l Conf. ICDAR,*, pp. 1191-1195, (2009).

[21] P. Sanguansat, P. Yanwit, et al, *Language-based handprinted character Recognition: a novel method using spatial and temporal informative features – In Proc. of IEEE 13th Workshop on Neural Networks for Signal Processing*, pp.527-536, (2003).