Offline Signature Verification Using Support Vector Machine

Kruthi.C, Deepika.C.Shet,
Under the guidance of Dheeraj Dwarkanath,
Dept. of Information Science and Engineering,
PES Institute of Technology, Bangalore-560085, India.
kruthi.2511@gmail.com, shetdeepika.22@gmail.com.

Abstract— This project aims at developing a support vector machine for identity verification of offline signature based on the feature values in the database. A set of signature samples are collected from individuals and these signature samples are scanned in a gray scale scanner. These scanned signature images are then subjected to a number of image enhancement operations like binarization, complementation, filtering, thinning and edge detection. From these pre-processed signatures, features such as centroid, centre of gravity, calculation of number of loops, horizontal and vertical profile and normalized area are extracted and stored in a database separately. The values from the database are fed to the support vector machine which draws a hyper plane and classifies the signature into original or forged based on a particular feature value. The developed SVM is successfully tested against 336 signature samples and the classification error rate is less than 7.16% and this is found to be convincing.

Keywords— Offline signature verification, Support Vectors, SVM, Epsilon Intensive, SMO, EDH, Kernel Perceptron, Large-Margin-Hyper plane.

I. INTRODUCTION

Signature is a person's name depicted graphically or handwritten as a form of identification in order to authorize a check or it is a mark (sign) made by an individual to execute a document and signify knowledge, approval, acceptance, or obligation. A signature is based on Biometric authentication where a user's identity is verified by means of physical trait or behavioral characteristics. There are two different categories of verification system based on the mode of signature acquisition: online for which the signature is captured during the writing process and making the dynamic information available, and offline for which the signature is acquired after the writing process and, therefore, only static information is available [1]. Signature helps us enforce security in many cases such as transactions at banks, wills, assets, government documents etc. Major documents such as cheques, property papers are generally subjected to malpractices. Hence an automated signature verification system is necessary.

The objective of the signature verification system is to discriminate between two classes of signatures: the genuine and the forgery, which is related to intra-class and inter-class variability [2]. There exists some variation among signatures of the same person which might be due to stress and this is called Intra Personal Variation. The variation caused by forging a genuine signature by another person is called Inter Personal Variation. Forgeries can be either random or skilled [3]. Random forgeries are easier to detect since they are written without the prior knowledge of how the signature might look like, while the main challenge for a signature verification system is skilled forgery where in a person is able to write a signature that looks exactly like a original signature and seems like one of those intrapersonal variations. The aim of this paper is to propose an offline signature verification system that uses support vector machine tool to classify the signatures.

A support vector machine (SVM) [4] is a tool used for classification and regression prediction and is based on machine learning theory in order to maximize predictive accuracy. The main aim of SVM is to draw a decision plane among a set of objects having different class memberships and classify them. There are two broad categories of classifiers one is linear and another is non-linear.SVM falls into the category of linear classifier. In case the data set is non-linear, SVM uses one of the four kernel functions to map the data such that they are linearly separable. A SVM generally aims at producing a large margin hyper plane, i.e. the perpendicular distance between the nearest point from the hyper plane and the hyper plane must be maximum. However in the real life scenario there exists over lapping data set and hence the SVM relies on loss functions. These loss functions ignore the errors that are present within certain range of the true value. Hard margin, L1 soft margin, L2 soft margin are the widely used epsilon intensive loss functions [5]. The proposed support vector machine uses kernel perceptron algorithm and smo algorithm to implement the hard margin and L1 soft margin separately, and also it uses linear and polynomial kernel functions to map the overlapping data.

The system proposed can be divided into two major parts: training phase and testing phase. The block diagram of the proposed system is as shown in the figure 1.

978-0-7695-5100-5/13 $31.00 © 2013 IEEE
DOI 10.1109/ICSIP.2014.5

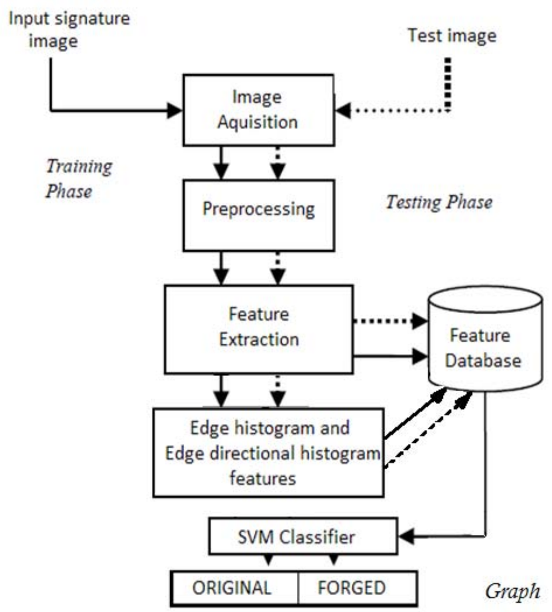

Figure 1: Block diagram of the proposed system.

The accuracy of the proposed system can be measured by applying FAR and FRR [1]:

i. FRR: Ratio of number of genuine signatures rejected to the total number of signatures submitted.
ii. FAR: Ratio of number of forged signatures accepted to the total number of signatures submitted.

The system was tested on signature database consisting of about 336 signature samples and the results are found to be convincing.

II. SIGNATURE ACQUISITION

Signatures of an individual are obtained in 3 different forms using ball, gel and sketch pens in separate templates. Each template consists of 14 fixed size blocks with in which the individual is expected to sign in the center. A total of 336 signatures were obtained from different individuals. These templates are then scanned using a gray scale scanner and the image is saved in tiff format. Each signature was then tight cropped [6] manually using Adobe Photoshop in order to concentrate only on the region containing data and separate database was maintained for each kind of signature. The tight cropped images are further considered for preprocessing.

III. PRE-PROCESSING

The scanned images cannot be directly used for any feature extraction methods. The images might contain certain noise, or might be subjected to damage because it's a old signature or the signatures under consideration may vary in size and thickness. Hence some kinds of image enhancement operations need to be performed and they are carried out in this phase. The following are the preprocessing techniques used.

A. Binarization

In the Binarization [7] method of binarizing an image by extracting lightness (brightness, density), each pixel in an image is converted into one bit and assign the value as '1' or '0' depending upon the mean value of all the pixel. If greater then mean value then its '1' otherwise its '0'.

B. Complementation

Complementation is done as the natural tendency of any one is to have data in form of 1s. In the complement of a binary image, zeros are changed to ones and ones are changed to zeros; black and white are reversed. In the output image, dark areas become lighter and light areas become darker.

C. Filtering

Filtering [6] is done in order to remove the noise that might be introduced during the scanning process. This has been done using median filter. The median filter is a digital filtering technique which is nonlinear and is often used to remove noise [6]. It considers each pixel in the image in turn and looks at its nearby neighbors to decide whether or not it is representative of its surroundings and then replaces it with the median of those values.

D. Edge thinning

Edge thinning [8] is a technique which is used to remove the outcast specious points on the edge of the signature image. The edge operator has been applied (like canny, sobel) to detect the edges and the edges are smoothed using an appropriate threshold.

E. Canny Edge detection

Edge is a boundary between two homogenous surfaces. Applying an edge detection algorithm (either using gradient or Laplacian method) to an image which shows acute variation in brightness or, which has layoffs, significantly reduces the amount of data that is to be processed and thus filters out the less relevant, non structural information of the image and preserves only the important structural properties.

The figure 2 shows the preprocessing stages of a signature sample

Figure 2: Preprocessing steps.

IV. GLOBAL FEATURE EXTRACTION

Features describe the signature image uniquely. Features can be of two kinds global and local. Global features describe the signature image as a whole while in the local feature extraction the signature image is divided into a num of units and in each unit the global features are extracted. We have included only a few important global features from the signatures mentioned below:

A. Aspect Ratio

Aspect ratio is the ratio of width to height of the signature image [8] [9].

$$\textit{Aspect ratio}=W/H \qquad (1)$$

Where W is width of the signature and H is the height of the signature.

B. Normalised Area

Normalization [6] is a process in which the range of intensity of pixels is altered. Normalization is sometimes called contrast stretching. The purpose of normalization is usually to bring the image to a common scale.

C. Horizontal and Vertical profiles

Horizontal and vertical projection is computed from both the binary and the skeletonised images. Number of black pixels (indicated by 1's in the Binarized and complemented image) is counted from the horizontal projections and vertical projections respectively **[6][9].**

D. Vertical centroid

The vertical centre of gravity is used to indicate the location and strength of the signature baseline and the answer was obtained from the horizontal projection [9] Ph as:

$$\sum_i \boldsymbol{i} * \boldsymbol{Ph}[\boldsymbol{i}] / \sum_i \boldsymbol{Ph}[\boldsymbol{i}]. \qquad (2)$$

E. Slant angle

A gray scale image of the signature sample whose angle to be found is considered first, the connected components in the signature is found by filling in the holes and then the color pixels in the signature samples are categorized into back ground and fore ground. The Slant angle [6] is then found using the m*n matrix of the changed gray scale image using the equation:

$$\textit{Angle}=\textit{atan}\ (m/n) \qquad (3)$$

F. Edge histogram

The edge histogram [10] and edge directional histogram [10] are mainly meant to extract the texture feature of the signatures. Texture is also a thing to be considered if the signature obtained is a old one and is subjected to some kind of degradation. It s calculated by obtaining the gradient of pixels that is the maximum rate of change of co-ordinates, (x, y), in the five directions with a threshold value of 100 as given equation below where, G_x and G_y are gradient vectors in x and y direction.

$$\boldsymbol{\theta} = \boldsymbol{arctan}(\boldsymbol{Gx}/\boldsymbol{Gy}) \qquad (4)$$

The edges of each input signature image are grouped into five classes, vertical, horizontal, 45˚- diagonal, 135˚ - diagonal and isotropic (non-diagonal) based on strengths of directional edges. When the scanned gray scale image is given as input, we obtain output values in five different directions based on threshold as explained. The below given figure indicates the edge filters in five different directions.

$$\text{a.}\begin{bmatrix}1 & -1\\1 & -1\end{bmatrix} \quad \text{b.}\begin{bmatrix}1 & 1\\-1 & -1\end{bmatrix} \quad \text{c.}\begin{bmatrix}\sqrt{2} & 0\\0 & -\sqrt{2}\end{bmatrix}$$

$$\text{d.}\begin{bmatrix}0 & \sqrt{2}\\-\sqrt{2} & 0\end{bmatrix} \quad \text{e.}\begin{bmatrix}2 & -2\\-2 & 2\end{bmatrix}$$

Figure 3: (a) Horizontal edge filter (b) Vertical edge filter (c) Diagonal(45˚) edge filter (d) Diagonal (135˚) edge filter (e) Non diagonal (isotropic) edge filter.

G. Edge direction histogram(EDH)

The histogram for each of the images represents the frequencies of occurrences of five classes of edges in the corresponding images for texture extraction. The edge direction histogram uses the sobel operator which helps in capturing the spatial distribution of edges in four directions (0˚,-45˚, 45˚and 90˚) with filter mask shown below with sobel's operator in X and Y direction.

$$X = \begin{bmatrix}1 & 2 & 1\\0 & 0 & 0\\-1 & -2 & -1\end{bmatrix} \qquad Y = \begin{bmatrix}1 & 0 & -1\\2 & 0 & -2\\1 & 0 & -1\end{bmatrix}$$

The sobel operator calculates the opposite of gradient (directional change in intensity/colour) of image at each point to extract information.The edge histogram has eight bins corresponding to the Sobel filters to count the number of edge pixels in eight directions. Edge histogram is normalized with respect to the image size.When the scanned gray scale image is given as input; we obtain output values in four different directions based on threshold as explained.

V. DEVELOPING SVM

A support vector machine (SVM) is a machine learning task of inferring a function called classifier, from supervised (labeled) training data. They are a specific class of algorithms which are characterized by usage of kernels and optimize it with an algorithm that is very fast in the linear case, acting on the margin on number of support vectors. A support vector provides several computational advantages by presenting the solution for classification of signature by providing simple hypothesis using random test points. SVM contains some main features like maximum margin classifier: a decision strategy which separates the training data with the maximal margin and a nonlinear function that controls the input parameters to find a linear separating hyper plane which do not depend upon high dimensional feature space [11].

This type of classification approach depends on certain activation values which define the loss function that ignores errors, located within certain range of the true values and the function used is often called – epsilon intensive – loss function. Using this loss function for classification ensures the existence of the global minimum of the objective function and also the optimization of reliable generalization bounds**.** The variables used in the loss function are used to indicate the weights of the errors for parameter estimation on the training points and the weights are zero for all points that lie inside the defined band.

A. Dataset creation

The dataset to be passed for the SVM is derived from the preprocessed signatures in the database. From each of the preprocessed signature stored in the database all the above mentioned global features are extracted and the values tabulated. For each kind of signature we have taken four skilled forgeries and these signatures are also preprocessed similar to the original signature and the feature values are tabulated. From these tabulated values data set is created. A dataset indicates the set of points to be marked on the output graph from the SVM and also the labels indicating to which group these points must belong to. The dataset created is in .mat format. Below is a table of data set we have created. It contains X, a (18 * 2) matrix indicating the points and Y, a row matrix (18 * 1), indicating the labels. The first row in X is the x–coordinates and the second row is the y-coordinates. The matrix Y contains the labels corresponding to the 18 points. The first 14 are the points corresponding to original samples and has the label 1 and the next 4 samples 14 to 18 correspond to the forged ones and have the label 2.

Table 1: Sample data set created.

Variable Name	Value	Min	Max
X	<2*18 double>	0.9213	14
Y	<1*18 double>	1	2

B. Kernel functions

When the data does not have a good linear separator for classification problems it makes use of certain functions called the kernel functions [12], i.e. a support vector is presenting the solution for classification of signature by providing simple hypothesis using random test points that maps our data to a higher dimensional space, where the pairs of features for each sample will be linearly separable by a large margin. However, we notice that most of the learning algorithms only access data through performing dot-products (the way to interpret algorithms like Perceptron). So the idea of kernel leads us to perform our mapping in such a way that we have an efficient way of computing dot-products. A Kernel is a function K(x, y) such that for some mapping φ, $K(x, y) = \varphi(x) \cdot \varphi(y)$.

C. Loss functions

a. Kernel Perceptron Algorithm:

The pseudo code of kernel version of perceptron algorithm is as follows:

argument: $X := \{x_1, \ldots, x_m\} \subset X$ (data)

$Y := \{y_1, \ldots, y_m\} \subset \{\pm 1\}$ (labels)

function $f = \text{Perceptron}(X, Y, \eta)$
    initialize $f = 0$
    repeat
        Pick (x_i, y_i) from data
        if $y_i f(x_i) \le 0$ then
            $f(\cdot) \leftarrow f(\cdot) + y_i k(x_i, \cdot) + y_i$
    until $y_i f(x_i) > 0$ for all i
end

The figure 4 indicates the output of SVM developed using Kernel perceptron.

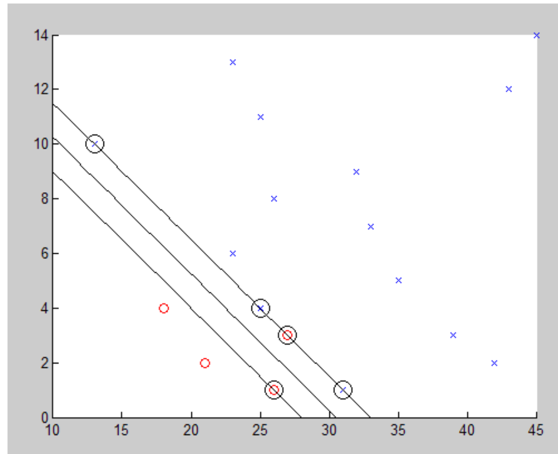

Figure 4: Output of SVM using Kernel perceptron.

b. Platt's Sequential Minimal Optimization(SMO) Algorithm:

SMO is an iterative algorithm which is used for solving the global optimization problem efficiently that can arise as binary classification problem during the training of SVM. SMO provides an analytical solution for solving the complicated problem by dividing it into a number of smallest possible easily solvable sub-problems. the Lagrange multipliers α_i, are subjected to linear equality constraint and therefore each of the smallest sub problem contains two such multipliers. Thus for any two multipliers α_1 and α_2, the constraints are reduced to $0 \le \alpha_1, \alpha_2 \le C$, $y_1 \alpha_1 + y_2 \alpha_2 = K$ and thus the problem can be solved using an analytic approach.

1. First step of algorithm is to find a Lagrange multiplier α_1 that violates the Karush–Kuhn–Tucker (KKT) conditions for the optimization problem.
2. Then it picks a second multiplier α_2 and optimizes the pair (α_1, α_2).
3. This is iterated until convergence occurs. The problem can be solved when all the Lagrange multipliers satisfy the user-defined tolerance (i.e., KKT conditions).
4. Although this training algorithm is primitive, it is guaranteed to converge; trial and error methods are used to increase the rate of convergence by choosing the appropriate pair of multipliers.

In SVM context a Lagrange Multiplier allows us to simplify the constraints as it is considered as the ratio of gradient of Objective Function to the gradient of Constraint Function at the solution of the problem. When the problem is expressed with Lagrangian multipliers (α_i) the only constraints are non-negative α_i and the solution which satisfies the KKT conditions are an optimal solution. In SVM equations this means $\Sigma\ \alpha_i\ y_i = 0$ should be satisfied. Therefore only support vectors contribute to the constraints on the margin.

The figure 5 indicates the output of the SVM using SMO algorithm.

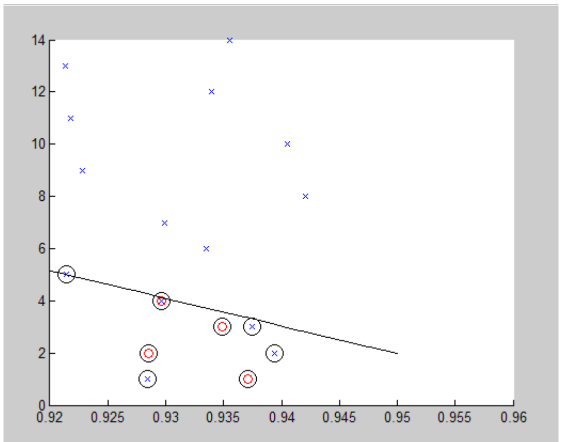

Figure 5: Output of SVM using SMO.

VI. RESULT ANALYSIS

The support vector machine was successfully developed with both SMO algorithm and Kernel Perceptron. . The developed SVM was tested against both linear and polynomial kernel and confusion matrix was drawn. The values obtained in the process are tabulated as shown below.

Table 2: error rate values using SMO algorithm.

	SMO ERROR RATES							
FEATURES:	DEEPIKA				KRUTHI			
	BALL		GEL		BALL		GEL	
	linear	poly	linear	poly	linear	poly	linear	poly
ANGLE	0.1667	0.1667	0.2222	0.2222	0.0556	0.0556	0.1667	0.1667
AREA	0	0	0	0	0.4444	0.4444	0.2222	0
ASPECT RATIO	0	0.2222	0	0	0	0	0	0
CENTROID	0.2222	0.1667	0	0	0.1667	0.2222	0.1667	0.2222
LOOPS	0	0	0.0556	0.0556	0	0	0	0
NORM AR	0	0.1667	0	0	0.1111	0.1111	0	0
VARIANCE	0.1111	0	0.4444	0.2222	0.4444	0.4444	0.2222	0.3333

Confusion matrix consists of four variables namely:

1. True positive: The percentage of total number of genuine signatures taken as genuine.
2. True negative: The percentage of total number of genuine signatures taken as forged.
3. False positive: The percentage of forged signatures taken as genuine.
4. False negative: the percentage of forged signatures taken as forged.

The below is the confession matrix results obtained for the two algorithms:

Table 3: Comparison of confusion matrix values.

ALGORITHM	TRUE POSITIVE (%)	TRUE NEGETIVE (%)	FALSE POSITIVE (%)	FALSE NEGATIVE (%)
SMO	71.19	6.57	15.08	7.16
KERNEL PERCEPTRON	73.36	4.82	15.67	6.15

Table 4: Comparison of error rates.

ALGORITHM	FAR (%)	FRR (%)
SMO	7.16	6.57
KERNEL PERCEPTRON	6.15	4.82

VII. CONCLUSION

In this project we are trying to validate whether a signature sample is forged or not using support vector machine. We have acquired the signature samples of different individuals, pre-processed using techniques like binarization, complementation, thinning, filtering and edge detection. Further from these pre-processed signatures features such as aspect ratio, centroid, number of loops, area and slant angle are extracted. These feature set are separately passed through the support vector machine developed using SMO and kernel perceptron, which are tested against both linear and polynomial kernel. The accuracy of the whole system is found to be 72.275 %.

The kernel perceptron with FRR and FAR of 6.15% and 4.82% respectively is found to be a better algorithm than the SMO algorithm with FRR and FAR of 7.16% and 6.57%.

REFERENCES

[1] Offline Signature Verification Using Classifier Combination of HOG and LBP Features Mustafa Berkay Yilmaz, Yanikoglu, Caglar Tirkaz Faculty of Engineering and Natural Sciences Sabanci University Istanbul, Turkey 34956,Alisher Kholmatov TUBITAK UEKAE Kocaeli, Turkey 41470.

[2] Offline Geometric Parameters for Automatic Signature Verification Using Fixed-Point Arithmetic, Miguel A. Ferrer, Jesu´s B. Alonso, and Carlos M. Travieso,2006.

[3] International Journal of Modern Engineering Research (IJMER) www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1171-1175 ISSN: 2249-6645,Offline Signature

Verification Using Neural Network, S.T. Kolhe, S. E. Pawar (Dept. of Computer Engg, AVCOE, Sangamner, India) (Dept. of I.T. Engg, AVCOE, Sangamner, India).

[4] A comparison of SVM and HMM classifiers in the off-line signature verification, Edson J.R. Justino, Fla´vio Bortolozzi , Robert Sabourin ba PUCPR—Pontifı´cia University ade Cato´ lica do Parana´ , Rua Imaculada Conceic¸a´ o, 1155, Curitiba, CEP 80215-901, PR, Brazil, E ´ TS—E´ cole de Technologie Supe´rieure, 1100, rue Notre-Dame Ouest, Montre´ al, Que´bec, Canada H3C 1K3, Received 18 October 2004.

[5] Statistical Learning Theory Lecture: 6 , Non-separable (soft) SVMs, Lecturer: Peter Bartlett ,Scribe: Joseph Austerweil , (Spring 2008)

[6] A new signature verification technique based on a two-stage neural network classifier H. Baltzakisa, N. Papamarkosb, A Department of Computer Science, University of Crete, P.O. Box 2208, 71409 Heraklion, Crete, Greece, Electric Circuits Analysis Laboratory, Department of Electrical & Computer Engineering, Democritu University of Thrace, 67100 Xanthi, Greece. Received 1 November 1998; received in revised form 1 June 2000; accepted 1 September 2000.

[7] Offline Signature Verification Using Local Radon Transform and Support Vector Machines, Vahid Kiani, Reza Pourreza & Hamid Reza Pourreza International Journal of Image Processing (IJIP) Volume(3), Issue(5) 184,july 2004.

[8] Off-line Signature Verification and Recognition by Support Vector Machine, Emre Özgündüz,Tülin Şentürk and M. Elif Karslıgil Computer Engineering Department, Yıldız Technical University Yıldız , Istanbul, Turkey.

[9] Offline Signature Identification by Fusion of Multiple Classifiers using Statistical Learning Theory Dakshina Ranjan Kisku1, Phalguni Gupta2, Jamuna Kanta Sing3 1.Department of Computer Science and Engineering, Dr. B. C. Roy Engineering College, Durgapur-713206, India, 2.Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, Kanpur-208016, India, 3.Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India.

[10] A Combined Color, Texture and Edge Features Based Approach for Identification and Classification of Indian Medicinal Plants, Basavaraj S. Anami, Principal K.L.E.Institute of Technology, Hubli-580030, Karnataka, India, Suvarna S. Nandyal, Research Scholar, PDA College of Engineering, Gulbarga-585102, India, A. Govardhan, Principal JNTUH College of Engineering, KarimNagar-505501, AP, India.

[11] Offline Geometric Parameters for Automatic Signature Verification Using Fixed-Point Arithmetic, Miguel A. Ferrer, Jesu´s B. Alonso, and Carlos M. Travieso, August, 2001.

[12] Kernel neuron and its training Algorithm, Jian hua X U, Xuegong ZHANG and Yanda LI, Department of Automation, Tsinguha university Beijing.