Signature Verification System using Support Vector Machine

Fauziyah S., Mardiana B., Zahariah M., Hazura H.,

***Abstract*— This paper highlights the development of online signature verification system using Support Vector Machine (SVM) and VBTablet 2.0 to verify the input signature by comparing database. This may take place by signing directly on to a digitizing tablet by using stylus which is connected to the Universal Serial Bus (USB) port of computer. Owing to the fact that each individual has its own way of presenting his/her signatures on paper, there is a certain level of complexity like the way of holding the stylus, the strokes used in the signing and the amount of time/pressure put on paper which are involved in this verification system. The general on-line verification procedures are preprocessing, features extraction, detail matching and post processing. The common verification algorithm is one of the Global Feature Vector Machine called Support Vector Machine (SVM). The signature is characterized as pen-strokes consisting x-y coordinates and the data will be stored in the signature database in the form of a txt.file.**

***Index Terms*—Signature verification, Support Vector Machine**

I. Introduction

SIGNATURE verification has been developing since the 1960 year and has been receiving intensive interest since the 1980 year. In 1677 England passed an act to prevent frauds and perjuries by requiring documents to be signed by the participating parties [4]. In 1997 the first studies of both off-line and on-line signature verification algorithms were published. Nagel and Rosenfeld research especially in off-line system. Liu and Herbs more research about on-line system [4]. Until now there are a lot of methods to verify the signature verification that have been developed by human. There are ranges of ways to identify signature verification. Signature verification is generally be divided into two vast areas namely static methods or sometimes called off-line that assume no time relayed information and dynamics methods sometimes called on-line with time related information available in the form of p dimensional function of time, where p represents the number of features of the signature [5].

Signature verification is becoming better liked in the industry, and the dynamics identification of handwriting speeds and pressures has significantly improved the accuracy of this biometric type. For smaller budget, signature verification can be a cost-effective solution for analyzing and authenticating signature dynamics [5]. The increasing demand for reliable human large-scale identification in governmental and civil applications has boosted interest in testing of biometric systems. Biometrics is an emerging technology that is used to identify people by their physical and/or behavioral characteristics that inherently requires that one to be identified is physically present at the point of identification.

Signatures are one of the most popular and reliable biometric features for verifying person's identity. In this paper, a new approach for the verification of signatures is based on the Support Vector Machine (SVM). The Support Vector Machine is a new type of learning machine for pattern recognition and regression problems, which constructs its solution (decision function f) in terms of a subset of the training data, the Support Vector. Support Vector Machines (SVM) is very popular since a few years. As they provide very good results for various pattern recognition problems, they also seem to be a good choice for online signature recognition. Compared to most methods used for signature verification such as Hidden Markov Models (HMM) or Dynamic Time Warping (DTW), SVM, which are based on the principle of structural risk minimization, have various advantages such as a convex objective function with very fast training algorithms. On the other hand, SVM typically are applied to data sets containing feature vectors of fixed length and not to problems dealing with time series of variable length such as in online signature recognition [6]. SVM is mainly used in classification and regression problems. In classification it involves estimation of the decision function, f using a set of training data with the labels that will correctly classify unseen test examples. Meanwhile, for regression, it is the estimation of real-valued functions, which is carried out in analogous manner to the case of pattern recognition.

II. Pre-processing module

Data acquisition is required to acquire the signature of the user which cab be based on a variety of input tools. Size normalization is performed by scaling each character both horizontally and vertically [1]:

This work was supported in part by the University of Technical Malaysia Malacca (UTeM) under Grant PJP/2007/FKEKK(8)-S279.

Fauziyah S., is with the Faculty of Electronics Engineering and Computer Engineering (FKEKK), UTeM, Malaysia (fax: 606-5552112; e-mail: fauziyah@live.utem.edu.my).

Mardiana B., is with FKEKK, UTeM, Malaysia (e-mail: mardiana@live.utem.edu.my).

Zahariah M., is with FKEKK, UTeM, Malaysia (e-mail: zahariah@live.utem.edu.my).

Hazura H., is with FKEKK, UTeM, Malaysia (e-mail: hazura@live.utem.edu.my).

$$x_i = \frac{x_i^o - x_{min}}{x_{max} - x_{min}} W$$

$$y_i = \frac{y_i^o - y_{min}}{y_{max} - y_{min}} H \qquad (1)$$

where (xi^o, yi^o) denotes the original point (xi, yi) is the corresponding point after the transformation.

$$x_{min} = \min i\{xi^o\},\ x_{max} = \max i(xi^o)$$

$$y_{min} = \min i\{yi^o\},\ y_{max} = \max i(yi^o) \qquad (2)$$

where W and H are the width and height of the normalized signature respectively. Re-sampling is done to make the raw data points equidistant in time using a simple linear interpolation algorithm as follows. The re-sampling step is a fraction of the total arc length L: Where d_i denotes the distance of point to point and n is the number of points. After re-sampling, the characters have a fixed number (n_1) of points per character (50 points in system) which provides a fixed size input.

$$d_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}$$

$$L = \sum_{i-1}^{n-1} d_i$$

$$\Delta S = \frac{L}{n_1} \qquad (3)$$

As a result, the total numbers of points are fixed for every character after performing re-sampling process. This will further ease the upcoming process, known as feature extraction. The purpose of the feature extraction module is enhanced the variability which helps to discriminate between classes. In this system integration of the online features are used. Online features includes: pen-up down, pen coordinates, direction θ and curvature $\in$. A binary feature '1' indicates the pen is touching the pad (pen-down) and '0' indicates the pen is not touching the pad (pen-up). The direction of a stroke is determined by a discrete approximation of the first derivative with respect to the arc length. These approximations can be calculated as shown in Fig. 1 in which the following calculations are required.

$$\cos\theta(n) = \frac{\Delta x(n)}{\Delta S(n)} \qquad (4)$$

$$\sin\theta(n) = \frac{\Delta y(n)}{\Delta S(n)} \qquad (5)$$

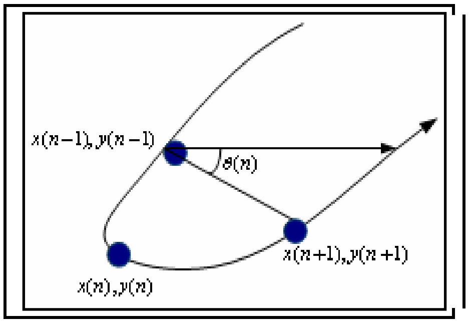


Fig. 1. Estimation of Writing Direction

$$\Delta x(n) = x(n+1) - x(n-1)$$

$$\Delta y(n) = y(n+1) - y(n-1)$$

$$\Delta S(n) = \sqrt{\Delta x(n)^2 + \Delta y(n)^2} \qquad (6)$$

The curvature of the strokes as the second derivatives $\frac{d^2x}{ds^2}$ and $\frac{d^2y}{ds^2}$ are not bounded on and the local curvature are approximated by the angle between two elementary segments. This can be shown as in Fig. 2. This angle is also encoded by its cosine and sine. Using the subtraction formulas for sine and cosine these values can be calculated as:

$$\cos\theta(n) = \cos(\theta(n+1) - \theta(n-1))$$
$$= \cos\theta(n+1)\cos\theta(n-1) + \sin\theta(n+1)\sin\theta(n-1) \qquad (7)$$

$$\sin\theta(n) = \sin(\theta(n+1) - \theta(n-1))$$
$$= \sin\theta(n+1)\cos\theta(n-1) + \cos\theta(n+1)\sin\theta(n-1) \qquad (8)$$

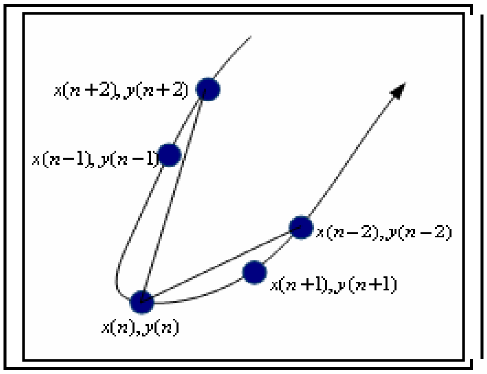


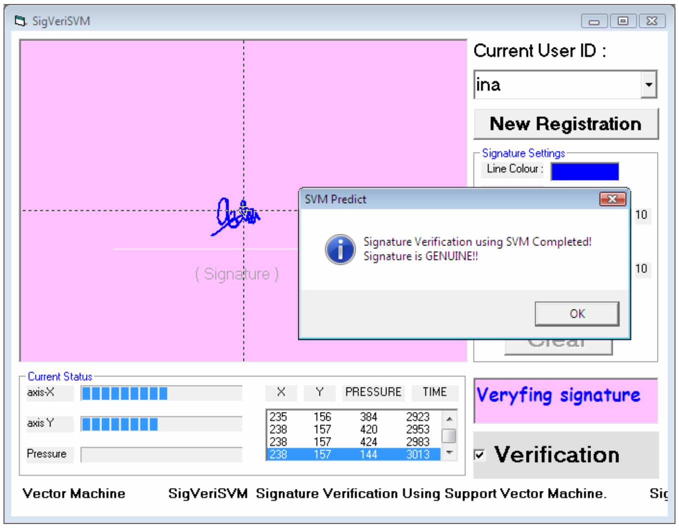
Fig. 2. Estimation of Curvature

III. Experimentation Analysis and Results

An experiment is carried out to determine the accuracy of this software, so as to prove that the objectives of this project have been achieved. This experiment involves two sections, which are software validation to prove its effectiveness in different and an analysis from a population of 10 registered users.

A. Software Validation

Software validation has been carried out to show the effectiveness of this software in verifying signatures. Before a signature can be verified as a genuine signature, there are many aspects to consider first. Thus a validation could be made to test the results obtained through this software.



Current User ID : ina

SVM Predict

Signature Verification using SVM Completed!
Signature is GENUINE!!

OK

```
235   156   384   2923
238   157   420   2953
238   157   424   2983
238   157   144   3013
```

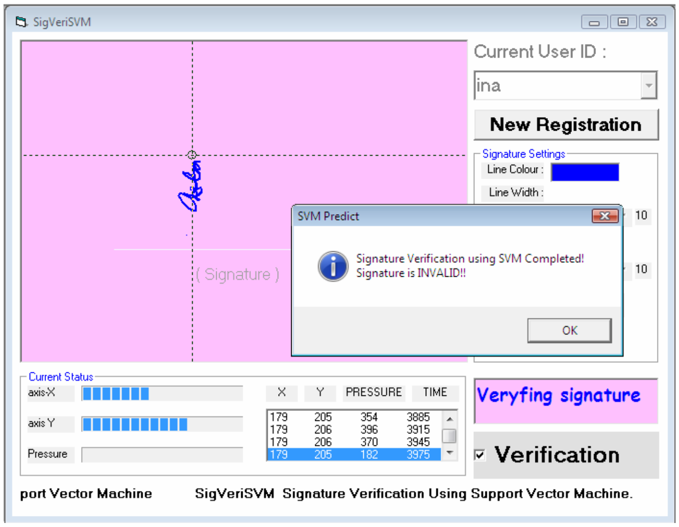
☑ Verification

Vector Machine SigVeriSVM Signature Verification Using Support Vector Machine. Sig

Fig. 3. Example of Signature Being Verified As Genuine

Basically, the validation process carried out is based on three aspects which are the orientation of the signature in coordinate x and y, pressure applied when signing the signature and the time used to sign the signature. Fig. 3 above shows a signature that matches the criteria needed in a signature to be verified as a genuine signature.

For the orientation aspect, it takes the X-coordinate and the Y-coordinate as the reference. Normally, a user's signatures will definitely have the same orientation each time he or she signed. If the user signs in the wrong orientation, in which the changes in X-coordinate and Y-coordinate are significant, this particular signature will be verified as a forgery signature. Fig. 4 below shows the different orientation of signatures, which are the root cause for being verified as forgery signature.



Current User ID : ina

SVM Predict

Signature Verification using SVM Completed!
Signature is INVALID!!

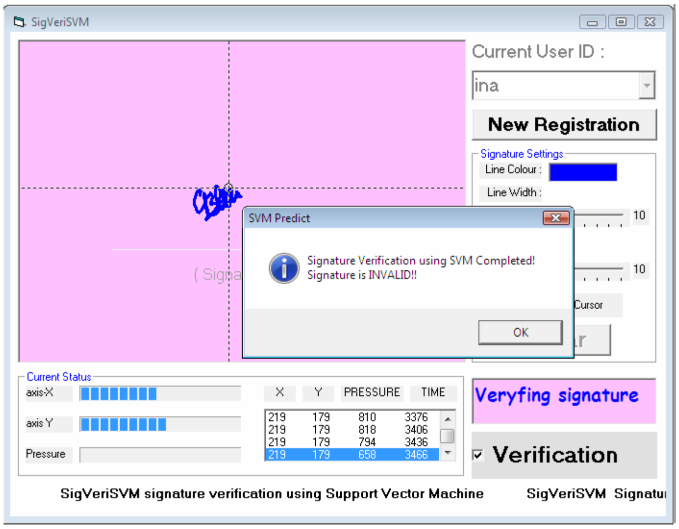
OK

```
179   205   354   3885
179   206   396   3915
179   206   370   3945
179   205   182   3975
```

☑ Verification

port Vector Machine SigVeriSVM Signature Verification Using Support Vector Machine.

Fig. 4. Example of Signature in Wrong Orientation



Current User ID : ina

SVM Predict

Signature Verification using SVM Completed!
Signature is INVALID!!

OK

```
219   179   810   3376
219   179   818   3406
219   179   794   3436
219   179   658   3466
```

☑ Verification

SigVeriSVM signature verification using Support Vector Machine SigVeriSVM Signatu

Fig. 5. Example of Signature with Different Applied Pressure

As for the pressure aspect, a signature will be verified as forgery signature if the pressure applied from the pen towards the digitizing tablet is significantly different from the genuine signatures that have been trained. The pressure is measured through values where a higher value shows that pressure is higher pressure and vice versa. Fig. 5 above illustrates different pressure being applied when signing a signature that causes it to be verified as forgery

B. FRR and FAR Analysis

There are two quantities in characterizing the performance of the signature verification algorithm. False Rejection Fate (FRR) is defined as the percentage of genuine signatures being wrongly verified by this software as forgery signatures. False Acceptance Rate (FAR) is defined as the percentage of forgery signatures being wrongly verified by the software as genuine signatures. Table 1 below shows the results obtained throughout the whole experiment to analyze the False Rejection Rate and False Acceptance Rate to prove the accuracy of the software developed to verify online signatures.

TABLE 1
Results Obtained Through FRR and FAR Analysis

		Genuine Signature		Skilled Forgery Signature		Unskilled Forgery Signature	
No	UserID	#Tested	#Rejected	#Tested	#Accepted	#Tested	#Accepted
1	Ina	10	0	5	0	5	0
2	Ishak	10	1	5	0	5	0
3	Azean	10	1	5	0	5	0
4	Azlan	10	2	5	0	5	1
5	Ismail	10	0	5	2	5	0
6	Fira	10	0	5	0	5	0
7	Prema	10	2	5	0	5	0
8	Cyndi	10	1	5	2	5	0
9	Rena	10	1	5	1	5	0
10	Ija	10	1	5	0	5	0
TOTAL		100	9	50	5	50	1

The results show that the system is effective in recognizing online signature with a high accuracy in real time situation. In the future, it is possible to improve accuracy by using new feature extraction techniques and recognition methods. Also, the system can be potentially improved for the online recognition system genuine signatures. The FRR and FAR have to be as low as possible for achieve the accuracy of signature verification at 100%. The low rate of FRR and FAR have shown that this software has the capability to verify signatures within the range of acceptable accuracy.

IV. Discussions

Obtained throughout the validation of software, it basically proves that this software has the capability to disregard genuine signatures that do not meet the specific requirement. Even the slightest mistake in the genuine signatures can be verified a forgery signature. This tight security that feature of this software will indirectly contribute towards the increased False Rejection Rate but manage to lower down the False Acceptance Rate where forgery signatures can be hardly verified as genuine signatures. Additionally, it also causes inconvenience towards to users whom are not consistent in their own signatures. The investment of this tight security into the software is more worthwhile compared to having low security where the FRR will be lower down but there is an increased rate in FAR. Lower security of this signature verification software will cause more forgery cases to happen around the world.

V. Conclusions

Basically this project deals with the development of online signature verification using Support Vector Machine in verifying signatures that is either genuine or a forgery signature. In verifying signatures, the signatures will be processed through this software and this software would have the ability to tell us to accept or not for this currently input signature. Thus the ability of this software is in verifying signatures of anyone who already has a database in this software stored in the signature database. In order to have high quality, this software has to be accurate and reliable in verifying signatures. Therefore, the error rate used to measure the accuracy and reliability of this software has to be as low as possible and within an acceptable range. Thus, the FAR and FRR has to be low and if possible 0%, meaning it has a perfect accuracy. With its many applications of this software into our daily life especially on the financial transactions, online signatures verification is a biometrics technology with great potential. The application of automatic online signature verification will become increasingly accepted in the real world.

Acknowledgment

First of all I would to thanks to our sponsor UTeM, secondly to our Dean, Prof. Madya Muhammad Syahrir Johal and my-ex Dean Dato' Prof. Dr Mohd Nor Husain and Deputy Dean (Research) ENGR. Imran Ibrahim, my Head of Department, Mr. Sani Irwan Md. Salim, for their time, attention and contribution during this research project.

References

[1] Abdul Fadlil, Marzuki Khalid, Rubiyah Yusuf (2005). *Online Handwritten Character Recognition Based On Online-OfflineFeatures Using BP Neural Network* Centre of Artificial Intelligence.

[2] Andrew W. Moore (2001), *Support Vector Machine*. Carnegie Mellon University.

[3] Anil K.Jain, Salil Prabhakar ans Sharath Pankati, *Biometric Recognition: Security and Privacy Concern*, IEEE Transaction, 1994

[4] A.K.Jain,F.D.Friess, and S.D. Connel (2002). *Online Signature Verification*. Pattern Recognition, Vol 35 , pp. 2963-2972.

[5] A.Pacut, A Czajka,"Recognition of Human Signatures",pp. 1560-1564, 2001

[6] Bin Li, Kuan –Quan Wang, and David Zhang, *On-Line Signature Verification For E-finance and E-Commerce Security System*, IEEE Transaction, 2004

[7] Byron Anderson and Yi Gu, *Real Time Dynamic Signature Verification*, Goggles Transaction, 2004

[8] Cemil OZ, Fikret Ercal Demir. *Signature Recognition and Verification With ANN*. Sakaraya University

[9] Claus Bahlman, Bernard Haasdonk and Hans Burkhardt (2002). *Online Handwriting Recognition wit Support ector Machine-A Kernel Approach*. Albert Ludwings- University Freiburg, Germany.

[10] E.J.R Justino, F. Bortolozzi, R.Sabourin , "*Off-line Signature verification using HMM for random simple and skilled forgeries*", Proc. 6th Intl. Conf. On Document Analysis and Recognition, 2001, pp. 450-453

[11] Rosielawati Binti Zawawi, *Thesis The Electronic Lock Using Signature Recognition By neural Network*, 2005

[12] Yong Haur Tay. Offline Handwritten Signature Recognition Using Hidden Markov Model (HMM) Centre of Artificial Intelligence

Fauziyah Salehuddin Department of Computer, FKEKK, UTeM, Malaysia (fauziyah@live.utem.edu.my). Fauziyah received a B.S degree from Universiti Teknologi Mara (UiTM) in 2001 and M.S degree from Universiti Kebangsaan Malaysia (UKM), Bangi Malaysia in the field of Microelectronics.

Mardiana Bidin Department of Computer, FKEKK, UTeM, Malaysia (mardiana@live.utem.edu.my). Mardiana received a B.S degree and M.S degree from Universiti Kebangsaan Malaysia (UKM), Bangi Malaysia in the field of Microelectronics.

Zahariah Manap Department of Comunication Engineering, FEKK, UTeM,Malaysia (zahariah@live.utem.edu.my). Zahariah received a B.S degree and M.S degree from Universiti Kebangsaan Malaysia (UKM), Bangi Malaysia in the field of Wireless.

Hazura Haroon Department of Communication Engineering, FKEKK, UTeM, Malaysia (hazura@live.utem.edu.my). Hazura received a B.S degree and M.S degree from Universiti Teknologi Malaysia (UTM), Johor Bahru Malaysia in the field of Microwave