Embedded System for Biometric Online Signature Verification Using ARM Processor

Mr. Ch. AYYAPPA, Mr. S. MANIKANDASWAMY

Abstract—**This paper describes the implementation biometric of an embedded system for online signature verification using ARM processor. Online signature verification is one of the biometric features which can be used as a common method for identity verification. The online signature verification is the aim of difference between the original signature and forgery signature. The online signature verification is primarily focused on skilled forgery detection. The signatures are acquired using a digitizing tablet which captures both dynamic and spatial information of the writing. After pre-processing the signature, several features are extracted. The authenticity of a writer is determined by comparing an input signature to a stored reference set (template) consisting of three signatures. The similarity between an input signature and the reference set is computed using string matching and the similarity value is compared to a threshold. Several approaches for obtaining the optimal threshold value from the reference set are investigated. The best result yields a false reject rate of 2.8% and a false accept rate of 1.6%. The On-line Signature Recognition and Verification is implemented using MATLAB. This work has been tested and found suitable for its purpose.**

Index Terms—**Biometrics, embedded system, friendly ARM, handwritten signature, MATLAB.**

I. Introduction

Digital signature capture is used in a lot of applications now a days for verification. A signature which pertains to an individual is captured and treated like an image containing a pattern of pixels which can be used for verification. However no two signatures of a person are precisely the same. The important factor is to differentiate between the parts of the signature and those that vary with almost every signature. There are two types of features that validate the signature. They are Static and Dynamic. Static features are extracted that are recorded as an image whereas dynamic features are extracted from signatures that are signed in real time. Signature verification is a common Biometric technique to identify human beings for purposes of verifying their identity. Basically we can classify the Signature authentication system into two types:

Off-Line: The signature is scanned to get its digital image representation eg.pen and paper.

On-Line: Uses special hardware such as a digitizing tablet or a pressure sensitive pen to record the pen movements during signing.

Handwritten signature is the most usual method by which a person declares that they accept and take responsibility for a signed document. This method is extensively used by contemporary society and has a solid legal basis accepted by the international community as a personal authentication method. However, handwritten signature has certain disadvantages, which have hindered its widespread use as biometric modality. The main challenge currently faced by researchers is that samples taken from the same individual have a large variability in their shapes and over time. Besides, forgery signatures carried out by impostor's exhibit a small interclass variation, which makes their identification as intrusive users more difficult. However, an interesting advantage is that the acquisition process can be readily performed by electronic devices such as pen tablets, touch screens. These devices offer not only the possibility of capturing the stroke of the signature (spatial information represented by the horizontal and vertical pen position),but also other measurable characteristics such as pen pressure or pen angle versus time. Online signature verification is known as the method that includes these additional characteristics to increase the interclass variability between genuine and impostor signatures.

II. Design of the System

The biometric online signature verification algorithms include a set of functions based on different signal processing techniques. Such functions usually have a high computational cost, since they manage an important amount of data and deal with complex operations carried out in floating-point format. Usually, microprocessors clocked at moderate frequencies (50 MHz–400 MHz) are generally too slow for applications requiring intensive computations. Thus, recently several publications proposed the use of ARM processors as platform for implementing embedded biometric systems with outstanding performance in terms of execution time and cost developed a complete finger print recognition system embedded on ARM. The work presented an ARM implementation of an embedded system for online recognition. The performance offered by this system was better than that presented by a portable commercial device based on a high-performance processor.

Manuscript received March 25, 2015

Ch.Ayyappa, Department of Electronics and Communication Engineering,(Embedded System Technology), SRM University, Chennai, India, Mobile No +918122909138

S. Manikandaswamy, Department of Electronics and Communication Engineering, SRM University, Chennai, India, Mobile No +9940363970

The ARM processor is ideal for many real-time embedded applications with demanding size constraints and cost-sensitive considerations. The ARM processors Power optimized and cost-effective. Additionally, some ARM cores incorporate "Enhanced DSP" instructions. Decreased heat production and lower overheating risk.

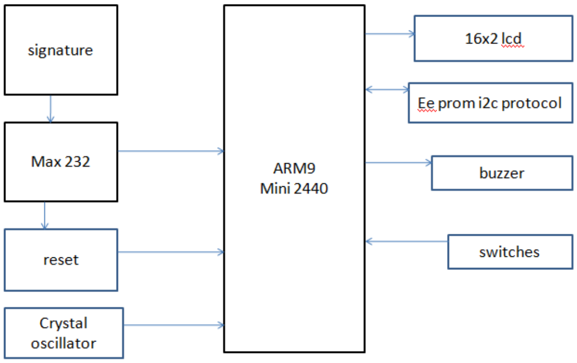
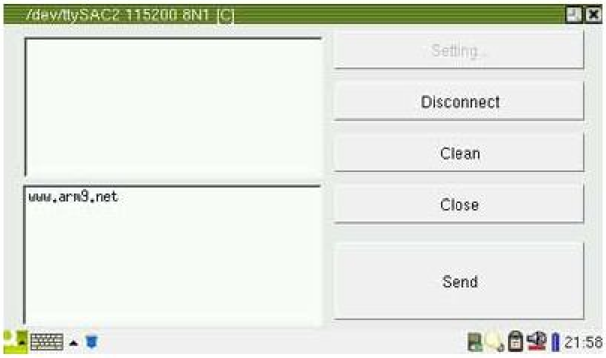


Fig 1: Block Diagram of the Overall System

The online signature verification is used to friendly ARM9 mini2440 controller. The mini2440 is a single board computer based on s3c2440 microprocessor. Using MATLAB the proposed system is MATLAB is the high-level language and interactive environment used by millions of engineers and scientists worldwide. It lets you explore and visualize ideas and collaborate across disciplines including signal and image processing, communications, control systems, and computational finance. In our project used MATLAB to verify the signature and to process in the ARM architecture and to show the result via serial port. Click on the “Connect” button to open “SAC0”, type some characters in the edit area, click on the “Send” button and it will send data to the connected serial port device. Click on “Disconnect” to disconnect the connection. Click on “Setting…” to enter the parameter setting interface which lists some basic serial port parameters. Communication Port is choosing “SAC0” or the USB to Serial “USB1”. Speed is bits per second. Data is data bits, 8 or 7.


/dev/ttySAC2 115200 8N1 [C]

www.arm9.net

Fig 2: Serial port communication between ARM processor and pc

III. Online signature algorithm

The training data is pre-processed and the features are extracted. This data is then saved in a database together with a unique identifier that is used to retrieve the signatures during matching. In addition, a threshold on the matching score is derived from the training data. For verification, a test signature along with the claimed writer identity is input to the system. The same pre-processing and feature extraction methods are applied. The signature is then compared to each of the reference signatures which are retrieved from the database based on the writer identifier. The resulting difference values are combined and, based on the individual threshold for the writer; the signature is accepted as genuine or rejected as a forgery.

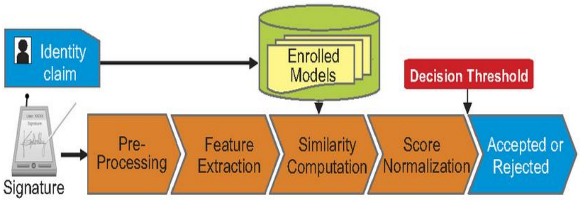


Fig 3: Steps in online signature verification

A. Acquisition

The acquisition of a signature is performed by a specialized input device. In our experiments a commercial pen tablet was used to obtain signature images for testing purposes. It is desirable that these data satisfy the regulations adopted in international standard organizations. This international standard has been developed to enable the interoperability among products and developments to facilitate their joint integration. It specifies two data inter change formats. The first one, known as full format, is well suited for our application, as it stores the raw data in the form of a multi dimensional time series vector with a precision of 2bytes.These named compact format, is oriented to smart-cards or other tokens with limitations in storage and communication capabilities.

B. Pre-processing

The input signal from a digitizing tablet or digitizing pen can be very jagged. The physical space provided for writing the signature may vary between different applications and the pen used can affect the smoothness and the size of the signature. A commonly used method to smooth the signature is based on a Gaussian filter. In order to compare the spatial features of the signature, time dependencies have to be eliminated from the representation. This is achieved by re-sampling the signature uniformly with equidistant spacing. Certain points in the signature, such as start and end points of a stroke and points of trajectory change carry important information. These points, referred to as critical points, are extracted before pre-processing and their positions are retained throughout the re-sampling and smoothing process. Temporal features must be extracted before re-sampling, and then propagated to the re-sampled points by interpolation. Solutions for pre-processing steps: i) Size Normalization ii) Position Normalization iii) Smoothing iv) Re-sampling v) Ligature.

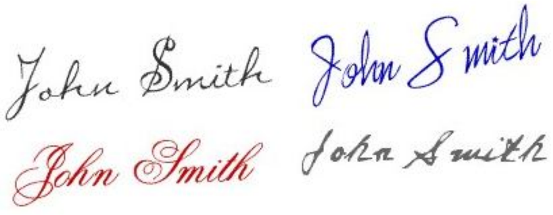


Fig 4: Pre-processing of online signatures.

C. Feature extraction

All strokes are combined into one long stroke during pre-processing. The original number of strokes is recorded and used as a global feature. From the x- and y-coordinates of the pre-processed image, a number of local features are extracted which are divided into two categories, spatial and temporal features. Spatial features are static features that are extracted from the shape of the signature. Local features are using the ordering (timing) of the signature. When the input data to an algorithm is too large to be processed and it is suspected to be very redundant (e.g. the same measurement in both feet and meters, or the repetitiveness of images presented as pixels), then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. Feature extraction involves reducing the amount of resources required to describe a large set of data. Absolute and relative speeds are defined as distance per unit time. Tablet PC captures the position of the pen 100 times per second Distance is measured in pixels. Only distance between points is necessary to define the speed. Speed is normalized by dividing the local speed at each sample point by the average writing speed of the signature Overall speed may vary but the relative speeds should be more stable.

D. Verification

In the verification process a test signature must be compared to all the signatures in the reference set (template database). Three basic methods to combine the individual dissimilarity values (between the input and one of the templates) into one value are investigated: (i) the minimum of all the dissimilarity values, (ii) the average of all the dissimilarity values and (iii) the maximum of all the dissimilarity values. After the dissimilarity value is computed, a decision regarding whether the signature is authentic or a forgery must be made. For this, the result of the matching will be compared to a threshold. If the dissimilarity value is above that threshold, the signature is rejected, otherwise it is accepted. The threshold can be chosen to be identical for all the writers or set individually for each writer.

A common threshold has the advantage that all the enrolment data from all the writers can be used to find an optimal threshold. The dissimilarities between all the signatures of all the writers who are enrolled into the system are computed and a threshold value is selected based on the minimum error criterion.

Writer-dependent threshold to adapt the verification process to the properties of a single writer, writer-dependent thresholds should be used. In principle, a writer-dependent threshold can be derived only from that writer's enrolment data. However, to reliably estimate the writer-dependent threshold, more enrolment data than usually available are necessary. To circumvent this, one starts with a common threshold and then modifies it for each writer by adding a writer specific component. Three choices to calculate the writer specific component from the reference set are investigated: (i) the minimum distance between all the references, (ii) the average distance between all the references and (iii) the maximum distance between all the references.

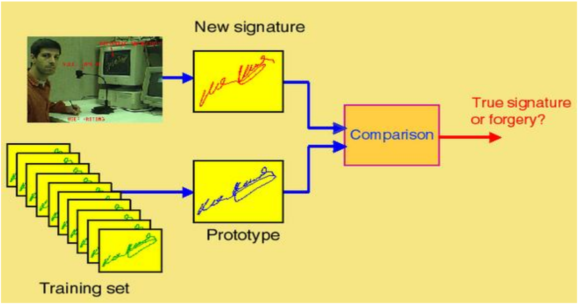


Fig 5: Verification steps in online signature verification

IV. Simulation Results

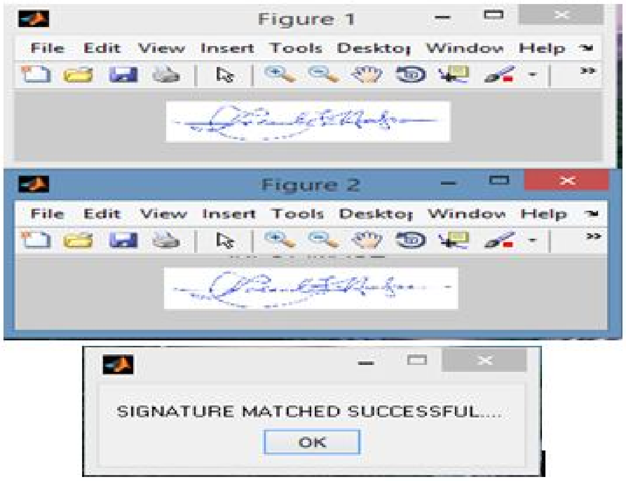


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 2
File Edit View Insert Tools Desktop Window Help
OK

SIGNATURE MATCHED SUCCESSFUL....

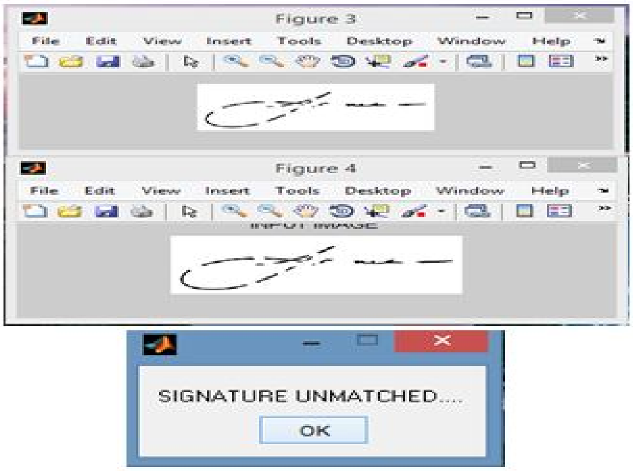


Figure 3
File Edit View Insert Tools Desktop Window Help
Figure 4
File Edit View Insert Tools Desktop Window Help
INPUT IMAGE
OK

SIGNATURE UNMATCHED....

Fig 6: Simulation results for online signature verification.

This proposed method has been implemented and evaluated with one session is signature matched and another session is signature unmatched. These signatures were collected database using ARM processor via serial port. Forgeries are classified into random or zero-effort forgeries and skilled forgeries. For a random forgery, the forger has either no knowledge about the original signature or does not try to imitate the shape of the signature. Since only a limited number of forgeries exist in our database.

V. Conclusion and Feature Work

A system for on-line signature verification has been implemented. The best results are biometric algorithm for signature verification based on the new system should be an on-line system. Shape is an integral part of signature verification; it is a biometric that is most easily imitated by a forger. Both global & local features should be used. Different methods have been tried with varying results, about 99% at the best. Great deal of speed improvement to be done. Signature segmentation into individual strokes needs attention. Multi-expert system to integrate different methods. The analysis on proper setting of thresholds and use of user-specific thresholds. Sensors have developed to a fair point of saturation. Study on multi-lingual signatures is unfocused.

The best error rates for a common threshold are 3.3% false rejects and 2.7% false accepts. Writer-dependent thresholds are computed from the reference signatures. All the reference signatures are matched with each other. The best feature set for writer-dependent thresholds consists of the absolute speed between critical points as the speed feature. Using the minimum dissimilarity value plus a user dependent offset results in 2.8% false rejects and 1.6% false accepts. Finally, more signatures must be collected and over a longer period of time. The current test database, consisting of signatures from 102 writers, is at most representative of an application appropriate for a small organization. Larger organizations or applications that use signature verification for their clients will have a much larger signature database and the scalability of our system needs to be investigated.

Acknowledgment

The author, Mr. Ch. Ayyappa wish to thank Mrs. A. Ruhan Bevi., Assistant Professor (Sr. Grade), Department of Electronics and Communication Engineering, Faculty of Engineering and Technology, SRM University, Tamil Nadu, India., for her helpful guidance and valuable comments.

References

[1] O. Miguel-Hurt ado, "Online Signature Verification Algorithms and Development of Signature International Standards" Ph.D. dissertation, Universidad Carlos III de Madrid, Madrid, Spain, 2011.

[2] O. Miguel-Hurt ado, L. Mengibar-Pozo, and A. Pacut, "A new algorithm for signature verification system based on DTW and GMM" in Proc. 42nd. Annu. IEEE Int. Carnahan Conf. Security Technol., Oct. 2008.

[3] S.D. Connell, Online handwriting recognition using multiple pattern class models, Ph.D. Thesis, MSU-CSE-00-27, Department of Computer Science, Michigan State University, and May 2000.

[4] D. Impedovo and G. Pirlo, "Automatic signature verification: The state of the art," IEEE Trans. Syst., Man, Cyber. —Part C: Appl. Rev., vol. 38, no. 5, pp. 609–635, Sep. 2008.

[5] Ma, M. M., W. S. Wijesoma, and E. Sung. 2000. An Automatic On-line Signature Verification System based on Three Models. Proceedings of Conference on Electrical and Computer Engineering. Halifax (NS). 890-894.

[6] R. Plamondon and S.N. Srihari, "Online and Offline Handwriting Recognition: A Comprehensive Survey", IEEE Tran. On Pattern Analysis and Machine Intelligence, vol.22 no.1, pp.63-84, Jan.2000.

[7] Shafiei, M. M. and H. R. Rabiee. 2003. A New On-line Signature Verification Algorithm Using Variable Length Segmentation and Hidden Markov Models. Proceedings of the 7th International Conference on Document Analysis and Recognition. 443-446.

[8] http://www.signotec.com.

[9] Jain, A., Griess, F., and Connel1, S. "Online Signature Recognition", Pattern Recognition, vol.35,2002, pp 29632972

[10] http://www.securedsigning.com.

Mr. Ch. AYYAPPA Master of technology in Embedded system Technology, Student, SRM University, Chennai, India.

Mr. S. MANIKANDASWAMY Assistant professor in Department of Electronics and Communication Engineering at SRM University, Chennai, India.