

Fingerprint Recognition Distributed System

Mihai Hulea, Adina Aștilean, Tiberiu Leția, Radu Miron, Silviu Folea,
Technical University of Cluj-Napoca,
[Radu.Miron, Adina.Aștilean, Tiberiu.Leția, Mihai.Hulea, Silviu.Folea]@aut.utcluj.ro,

Abstract:

The paper proposes a distributed identification and verification system which integrates fingerprint recognition and Global Positioning System methods. The system is composed of a number of mobile fingerprinting terminals which communicate with data processing units using GSM/GPRS technology. The motivation for integrating GPS technology in an identification system is to add a supplementary security level by granting access based on area localization. In order to further increase the system security a cryptographic module is used to encrypt / decrypt messages passed between the distributed components. The system architecture is presented and main modules are described.

I. INTRODUCTION

Biometric recognition refers to both *physiological* and *behavioral characteristics* applied for automatic recognition of individuals. A number of different biometric identifiers are used in various applications. Among the technologies based on the various biometrics, the finger-scan is the undisputed leader, considering its market share of more than 50%. [1]

Fingerprint-based technology is mainly used for biometric recognition because it satisfies better than the other biometric technologies, the following requirements: universality, acceptability, immutability, individuality and it is easy to collect. [2]

This paper proposes a distributed identification and verification system. The system comprises of a number of mobile identification terminals and distributed data processing units wherein the recognition procedures are performed.

Each terminal includes an encryption/decryption module capable of performing transformation over the transferred data in order to implement a security level. The distributed units contain identical encryption/decryption modules.

The system is assumed to work in two modes. The mobile terminals can be used for feeding the system with the subjects' identification data which are stored in a database located at the central level. Also the system works as a verification/identification system where matching algorithms are executed at the central processing level.

The terminal includes a fingerprint sensor and also a Global Positioning System (GPS) module. The GPS module is useful when the system is used to verify someone's identity. By adding this module the system is capable of authorizing the access of individuals to resources located in a certain area. In this way the system can discriminate between the enrolled users, granting them access only to areas they are authorized to enter.

It is not necessary for these components to be physically located on the same board as the mobile terminal – and in this proposal a modular approach was considered, where the fingerprint sensor was developed as an independent unit device.

The communication between mobile terminals and the distributed servers is implemented on top of the GSM/GPRS technology.

II. STATE OF THE ART

A fingerprint is essentially a reproduction of the fingertip epidermis, produced by pressing the finger against solid surface. The fingerprint basic characteristic is a pattern of interleaved *ridges* and *valleys*.

Fingerprint ridges begin to develop on the skin of the thumbs and fingers during the third to fourth month of fetal development, being fully developed by the seventh month. W. F. Leung et al. [3] calculated that the probability of two fingerprints being alike is 1 in 1.9×10^{15} .

Other important features used to recognize a fingerprint are the *minutiae*, which refer to either the *bifurcations* or the *terminations* of the ridges. (refer to Figure 1)

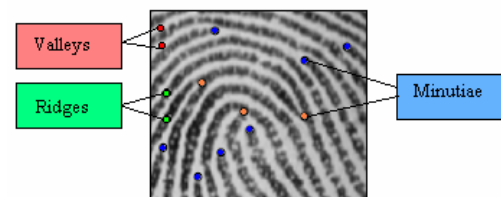


Figure 1 – Structural fingerprint characteristics

A *fingerprint-based biometric system* is used to recognize an individual, based on his unique fingerprint pattern. Depending on the application context, a fingerprint-based system might be either an *identification* system or a *verification* system. The difference between the two consists of the fact that an *identification* system conducts one-to-many comparisons in its database to match the input fingerprint with one of the pre-stored ones, thus identifying the individual. A *verification* system conducts a one-to-one comparison between the input fingerprint and a pre-stored one to verify a claimed identity. The generic term *recognition* is used where it is not necessary to differentiate between the two. [2]

In order to decrease the computational cost of fingerprint recognition, the database is structured according to the four main fingerprint topologies (the Henry classification system [4]): *arch*, *left-loop*, *right-loop* and *whorl*. These four main classes can be further divided into subclasses (e.g., arch class can be divided into plain arch and tented arch). While the Henry system has many classes, only 4, 5 or 7 of them are used in the automatic classification process, because it is quite difficult to determine a fingerprint class. According to Sarat C. Dass and Anil K. Jain [5], almost 20% of the fingerprints from NIST 4 [6] are associated to two classes.

The major steps involved in an automatic fingerprint recognition application are: Fingerprint Acquisition, Fingerprint Segmentation, Fingerprint Image Enhancement, Feature Extraction, Fingerprint Classification and Fingerprint Matching.

There are many methods and technologies for each step. Regarding the fingerprint matching algorithms, Maltoni D. et al. [2] group all these methods into three major classes: (i) correlation-based matching, (ii) minutiae-based matching and (iii) ridge feature-based matching.

According to [5] the fingerprint recognition approaches can be grouped into five categories: (i) approaches based on singular points [7, 8], (ii) structure-based [9, 10, 11], (iii) frequency-based [12], (iv) syntactic or grammar-based [13], and (v) approaches based on mathematical models [14].

In order to perform the recognition, in most of the cases, the fingerprints are stored on the device. This is a serious security hole because an attacker that gains access to the system can also gain access to the biometrics database. A possible solution to this problem is to store in the database only certain aspects of the fingerprint, thus making it impossible to recreate it.[15]

Hany H. Ammar et. al. [16] describe a 3-tiered architecture in which the distributed computing environment is divided into presentation, functionality, and data components. The functionality component is the backbone of the system and consists of a server that compares the fingerprint submitted by the client (presentation) with the ones pre-stored in the database.

A solution to ensure the security of the database and also to speed up the process of fingerprint recognition is a distributed mobile device system. The mobile device consists of a fingerprint scanner and a communication module that sends the captured fingerprint to a server via a communication network link. The server performs the recognition process and sends back the results. [17]

III. GENERAL SYSTEM ARCHITECTURE

A heterarchical architecture is used for a distributed fingerprint recognition system. The general system architecture is presented bellow, in Figure 2.

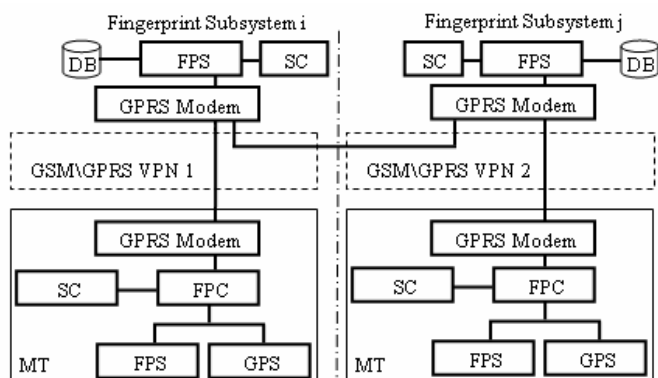


Figure 2. General software architecture

The architecture comprises a number of loosely coupled, highly autonomous Fingerprint Subsystems. Each system implements a suite of applications for fingerprint recognition. A

fingerprint subsystem is composed of a server and a number of mobile terminals.

At the upper level the connection between the fingerprint systems is implemented based on GSM/GPRS technology. Each Fingerprint Server (FPS) knows the address of all the other FPS units. At this level fingerprint validation and identification requests are exchanged. When a FPS unit cannot find a match for a fingerprint recognition request in its local database, it will forward this request to all known FPS units in the network. If a FPS unit finds a match, a reply will be sent back to the original FPS.

The architecture's lower level consists of the mobile terminals (MT). They are responsible for acquiring fingerprint data by using the fingerprint sensor (FS) and GPS data with the help of the GPS module, and for sending them over the GPRS connection link to the FPS. At FPS level data are processed and a response is sent back to the terminal client. Smartcards (SC) are used in order to provide an authentication mechanism for MT modules which connects to the FPS applications. The fingerprint and other identification and personal data are stored at the server level in local databases (DB).

In order to provide a secure connection between MT and FPS units, the use of Virtual Private Networks (VPN) technology built on top of GPRS connection is proposed.

IV. FINGERPRINT SERVER APPLICATION ARCHITECTURE

The UML component diagram presented in Figure 3 shows the main modules of the server.

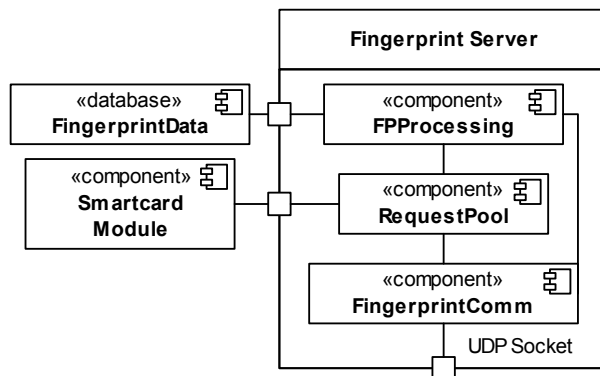


Figure 3. Server software architecture

The FPS functions are:

- To provide a communication link with the mobile clients over a Virtual Private Network built on top of the GPRS technology;
- To provide functions for enrolling, identification and validation based on GPS location and fingerprint data;
- To use smartcard module for user and client applicant authentication;
- To send/receive verification requests to/from other FPS units in the network;

A **RequestPool** mechanism is implemented at the FPS level in order to prevent congestions generated by too many requests

arriving at the same time from different clients. The Request-Pool is responsible for queuing requests. A set of concurrent worker threads executes the requests stored in the queue. When a new client connects, his request is accepted only if the number of current requests does not exceed the maximum allowed number (the queue size).

The **FingerprintComm** module provides the communication link between the mobile clients and the allocated server. The server is connected to the Internet and the communication is implemented using UDP transport protocol.

The server is connected to the **FingerprintData** database. This database stores the personal details and scanned fingerprint data. The connectivity with the database is implemented using Java Database Connectivity (JDBC) API.

FPProcessing is the core of the fingerprint server that is responsible for processing the requests received from the fingerprint clients. This server component is responsible for the following operations: (i) the fingerprint segmentation, (ii) the fingerprint image enhancement, (iii) the minutiae extraction, (iv) the fingerprint classification and (v) the fingerprint matching.

The *fingerprint segmentation* process isolates the fingerprint from the background. This is an important recognition procedure step, because by extracting the fingerprint from the picture given by the sensor, faulty fingerprint feature extraction caused by the background noise is avoided. The segmentation is done based on the local ridge frequency and orientation.

For the *image enhancement* step a Gabor filter approach was considered [18]. At this stage, after enhancing the image a transformation from a grey-scale image into a binary one is performed.

In order to *extract the minutiae*, a ridge thinning process is required, which allows for the ridge line thickness to be reduced to one pixel. After the thinning stage, determining the minutiae is done by examining the *8-neighborhood* of a ridge skeleton pixel, as shown in Figure 4.

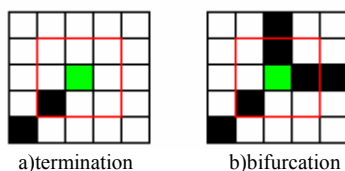


Figure 4. Minutiae extraction

The *fingerprint classification* stage is performed with the aim to reduce the number of comparisons between the input fingerprint and the templates stored in the database. If this classification is performed, the input will be compared only with the templates from the same class (e.g., arch, loop, whorl).

The system that is described in this paper uses a minutiae based matching method. The input fingerprint minutiae vector will be compared to the templates minutiae vectors in order to find a match. This technique consists of finding the alignment between the two minutiae sets that results in the maximum number of minutiae matching.

If FPS fails in matching the fingerprint in its local database then a request is sent to the other FPS modules in the network. If one of the remote FPS successfully matches the fingerprint, it will send back the response to the FPS request initiator. A FPS will forward a request to the other FPS units only if the request comes from one of the local MT modules.

V. FINGERPRINT MOBILE TERMINAL ARCHITECTURE

The Micro Edition of the Java 2 Platform (J2ME) has been used for implementing the client application in order to achieve maximum portability on different client devices. Additionally, J2ME has a good I/O abstraction layer instrumental for implementing the Bluetooth connectivity library and the communication protocol with the remote fingerprint server.

The software architecture of the fingerprint mobile terminal (MT) is presented in Figure 5.

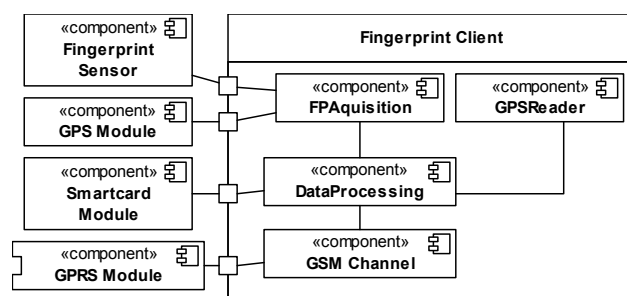


Figure 5. Fingerprint mobile client architecture.

The MT functions are:

- To capture fingerprint data and read GPS position;
- To communicate with FPS over the VPN GPRS network;
- To send enrollment\identification\validation requests to FPS;
- To provide a user interface for accessing recognition functions;
- To use smartcard module for terminal authentication;

GPSReader runs in its own thread. This module uses the Generic Connection Framework from the CLDC specification [19] to connect to a GPS receiver over a serial or a Bluetooth link. It runs in its own thread and notifies any listeners interested in the position changes using an event-based subscription pattern. It can be controlled using the start() and stop() directives, which are used from the MIDlet instance to pause the connection to the GPS receiver when the application is not in the 'Active' state. The GPS reader can be reconfigured to use a different protocol (e.g. Bluetooth, serial link) during runtime using a configuration window.

The *coordinates* are embedded in a NMEA sentence format. The National Marine Electronics Association developed the NMEA 0183 Interface Standard for data exchange between marine electronic devices. The NMEA 0183 Standard defines

electrical signal requirements, data transmission protocol, timing and specific sentence formats for a 4800 baud serial data bus. Nowadays, most GPS receivers use the NMEA interface for data exchange. The coordinates are packed in a NMEA RMC sentence format which is presented below, in Table 1. Details of the NMEA formats can be found in [20].

Table I.
The NMEA RMC format of the GPS data

Token number	Explanation
1	Time of fix in UTC Time
2	Data status
3	Latitude of fix
4	N or S
5	Longitude of fix
6	E or W
7	Speed over ground in knots
8	Track made good in degrees true
9	UT date
10	Magnetic variation degrees
11	E or W
12	Checksum

FPAquisition is responsible for reading the fingerprint from the fingerprint sensor devices and connects to the fingerprint sensor device over a Bluetooth link. The reading is a two step process. First, the user is asked to press his finger against the fingerprint sensor; the capturing process is started by sending a specific command to the sensor device. Second, the *FPAquisition* module uploads fingerprint raw image captured from sensor. The raw image is uploaded as a binary sequence of 45312 byte values. The image has 192columns and 236 rows. Data are gathered by the DataProcessing module.

The response from the sensor is received in the following format:

Binary sequence of 45312 bytes <CR> <LF>

DataProcessing module receives and parses GPS and fingerprint data and packs them in a format accepted by the fingerprint server. The data are packaged in an XML format of which structure is presented in the paragraphs below. For XML processing kXML is used. kXML is a small XML pull parser, specially designed for constrained environments such as Applets, Personal Java or MIDP devices.

A stateless XML over UDP protocol is used for the client-server interaction. The client sends an XML request to the server and the response is received also in the XML format. The server is possibly serving multiple clients and doesn't keep the state of previous requests.

The XML Schema language (XSD) [21] is used to express the rules to which the fingerprint server and client must adhere to. It specifies both the structure and the data types: the order of the tags in the XML, mandatory elements and the restrictions imposed on the data types. The messages sent between

fingerprint client and fingerprint server must adhere to the rules described in the following XML Schema:

```
<xsd:schema xmlns:xsd=http://www.w3.org/2001/XMLSchema>
<xsd:element name="fingerprint">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="imagedata"/>
      <xsd:element name="gpsdata"/>
      <xsd:element name="personaldata"/>
    </xsd:sequence>
    <xsd:attribute name="commandType" type="xsd:int"/>
    <xsd:attribute name="commandID" type="xsd:int"/>
    <xsd:attribute name="authenticationToken" type="xsd:long"/>
    <xsd:attribute name="terminalID" type="xsd:long"/>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```

The following types of commands are exchanged between server and client: authentication request, authentication response, identification request, identification response, validation request and validation response.

GSMChannel module implements the communication mechanism between fingerprint sensor and server. A GSM module is used at the hardware level.

Wireless communication network based on GSM/GPRS/UTMS is used as the communication backbone between the fingerprinting mobile devices and the server. The General Packet Radio Service (GPRS) has quicker session setup, permanent connection, lower cost, higher data rate performance [22]. Furthermore, the data transmission is IP based, so the transceiver is unnecessary in the management center, which is connected to the Internet.

The main option for the application programmer when configuring an IP-stack is the use of either TCP or UDP. The communication was implemented using UDP protocol which provides several advantages over the TCP protocol: an increased transmission speed, low latency, smaller packages and no need to maintain a connection between client and server.

A cyclic redundancy check (CRC) code is used to detect data alteration during the transmission. In the case of data alteration detection or if no response is received back for a command in a predefined time limit, the initiator will retry to resend the initial command and then will fail with an error code.

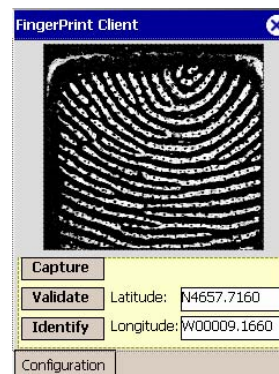


Figure 7. FP Client main window.

The Figure 7 presents the main window of the Java MIDLET mobile terminal application (MT) constructed based on the presented architecture. The application has been installed on an ASUS P535 with Intel XScale 520 MHz processor with GPS\GSM embedded modules. The device's Bluetooth technology is used for communicating with the fingerprint sensor module.

VI. FINGERPRINT SENSOR DESCRIPTION

As part of this research a prototype fingerprint unit has been developed. This is implemented as a separate dedicate board which communicates with mobile terminals using Bluetooth technology. From an architectural point of view this module is a part of the mobile terminal subsystem (MT) and it is responsible for the fingerprint data acquisition. The hardware diagram of the developed sensor unit is presented in Figure 8.

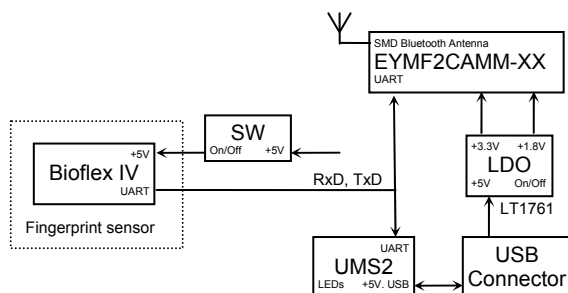


Figure 8 – Fingerprint Sensor block diagram

The BioFlex IV is the core of the fingerprint device and it consists of a powerful processor and a capacitive fingerprint sensor. The BioFlex IV can perform fingerprint verification but in the proposed solution this function of the sensor is deactivated and it is used only for capturing fingerprint biometrical data.

The characteristics of the BioFlex IV sensor are:

- Compact size, including a CPU which runs at 400 MHz, internal memory and VIP100 standard;
- The design permits easy integration of biometric security functions in any product;
- High C/P value, provide 1:1, 1:N fingerprint matching;
- Semiconductor chip sensor, anti-stain, 3-D image capture, 508 DPI resolution, 236 x 192 pixels numbers;
- Standalone system, with low power consumption: in idle mode 165 mA, in capture mode 250 mA, in enrollment mode 280 mA and in sleep mode only 50 mA.

The hybrid integrated circuit EYMF2Camm-XX is a Bluetooth module manufacture by Taiyo Yuden Ltd. The module includes a Serial Port Profile (SPP) for a simple cable replacement feature. The target applications of this module are PDAs, mobile phones, etc. The Bluetooth interface supports only point-to-point connections. Two LEDs, red and green indicate the state of the Bluetooth module: discovery, connected or command mode. The configuration of the module can be set from the application: the role – master or slave, the baud rate and the operation mode – automatic or command. [23]

The UMS2 is a module based on FT8U232BM integrated circuit manufactured by FTDI Ltd., which provides an easy to use USB connectivity to PC without any need of additional knowledge about USB itself. [24]

The UMS2 is a USB to UART interface with full hardware data flow control capability. In addition, there is an EEPROM memory which may be used to store identifier (ID) for applications. This interface is used for debugging the application and for supplying the development board in debug mode.

The fingerprint module can be powered directly from an USB port or from an external power supply. Three voltage levels are necessary: +5.0V, +3.3V and +1.8V, designed with three low dropout voltage regulators integrated circuit: LT1761-xx. The USB port can generate the +5.0V voltage level. The external power supply needs a battery holder with four AA batteries (1.5V).

The power management for BioFlex IV module includes a deep sleep mode which is automatically started after 30 sec if no commands are input. Deep sleep mode can be terminated only by resetting the power. As a result, a switch circuit to control the power module was added on to the board. [25]

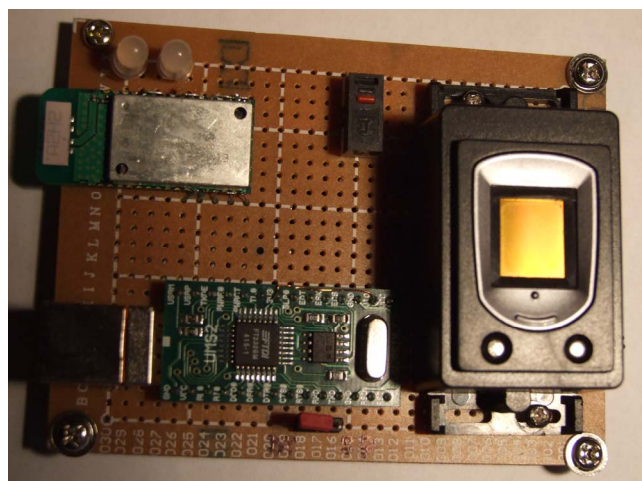


Figure 9 – Fingerprint sensor module prototype board.

The fingerprint sensor module board is presented in Figure 9. The FP sensor is on the right side of the board, the Bluetooth module is on the top-left corner and the USB module is on the bottom-left corner.

VII. SECURITY CONSIDERATIONS

Incorporate security into a distributed system is a complex task and attention should be paid to three aspects: establishing secure channels between distributed entities, access control and management of keys and authorizations.

Traffic data confidentiality is ensured by using VPN technology over GSM/GPRS connection. GPRS defines an encryption protocol based on an A5 algorithm, an authentication protocol based on an A3 algorithm, and a cryptographic key management protocol based on an A8 algorithm. Details about A3, A5 and A8 algorithms can be found in [26].

In a GPRS network, a subscriber is identified by an unique international mobile subscriber identity (IMSI) stored in the subscriber identity module (SIM) along with the handset (the phone). The SIM module also has a secret key (K_i) associated with the IMSI. On the network side, IMSI and its K_i are stored in an authentication center (AuC). The subscriber authentication is carried out in a challenge-and-response fashion, whereby a random number as a challenge is generated and sent to the mobile station by a serving GPRS service node (SGSN). The mobile station uses its K_i to produce a code as the response according to the A3 algorithm. The encryption key (K_c) is derived from the same random number and K_i by the A8 algorithm. On the mobile station, this is performed by the SIM module. Data traffic is encrypted using K_c by applying the A5 algorithm, a stream cipher.

Smart card authentication is proposed to be used in order to prevent unauthorized use of the FP system. A smart card reader device is installed at the MT and FPS levels. The system uses Java Card compatible smart cards. Performing the authentication is accomplished by the mobile terminal by obtaining the PIN from the user and by using appropriate APDUs to establish an authentication session with the smart card: create a secure channel to the card, issue a challenge, and verify the response.

VIII. CONCLUSIONS

The paper describes a distributed fingerprint based recognition system. The system has a multilayer heterarchical architecture. The fingerprint servers are interconnected over the internet. The mobile fingerprint terminals are connected to the server by using the data transmission technology provided by GSM networks.

The system can be used in three modes: enrollment, verification and identification. The enrollment mode allows an authorized mobile device user to register a new subject into the system.

Law enforcement personnel would be greatly aided by the use of such a system. Officers' time could be saved by making it possible to identify a subject in the field and not having to make a trip to the police station. False arrests could also be avoided by positively identifying a subject in the field. For the identification use, the system can also be deployed to airports, ports of entry and other government agencies.

Regarding the verification scenario, the use of the GPS module can help granting different access privileges to restricted areas or resources to enrolled subjects.

The system could be used as an authentication method for internet banking, having the advantage that the fingerprint based verification is much safer than an username/password system.

REFERENCES

- [1] "Biometrics Market Report 2003-2007". International Biometrics Group, 2002, http://www.biometricgroup.com/reports/public/market_report.html
- [2] Maltoni D., D. Maio, A.K. Jain, S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, New York, 2003, p 3-7
- [3] W. F. Leung, S. H. Leung, W. H. Lau, And A. Luk, *Fingerprint Recognition Using Neural Network*, Neural Networks For Signal Processing – Proceedings Of The 1991 IEEE Workshop
- [4] E. R. Henry. *Classification and Uses of Fingerprints*. London: Routledge, 1900.
- [5] Sarat C. Dass, Anil K. Jain, *Fingerprint Classification Using Orientation Field Flow Curves*, ICVGIP 2004, p 650-655
- [6] C. I. Watson and C. L. Wilson. *Nist special database 4, fingerprint database*. National Institute of Standards and Technology, March 1992.
- [7] L. Hong and A. K. Jain. *Classification of fingerprint images*. Proceedings of the 11th Scandinavian Conference on Image Analysis, Kangerlussuaq, Greenland, June 7-11, 1999.
- [8] K. Karu and A. K. Jain. *Fingerprint classification*. *Pattern Recognition*, 29(3):389–404, 1996.
- [9] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. *Fingerprint classification by directional image partitioning*. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35:1209–1223, 2002.
- [10] J. Chang and K. Fan. *A new model for fingerprint classification by ridge distribution sequences*. *Pattern Recognition*, 35:1209–1223, 2002.
- [11] M. M. S. Chong, T. H. Ngee, L. Jun, and R. K. L. Gay. *Geometric framework for fingerprint image classification*. *Pattern Recognition*, 30(9):1475–1488, 1997.
- [12] A. K. Jain, S. Prabhakar, and L. Hong. *A multichannel approach to fingerprint classification*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4): 348–359, 1999.
- [13] B. Moayer and K. S. Fu. *A syntactic approach to fingerprint pattern recognition*. *Pattern Recognition*, 7(1):1–23, 1975.
- [14] S. Minut and A. K. Jain. *Hierarchical kernel fitting for fingerprint classification and alignment*. *Proc. of International Conference on Pattern Recognition*, Quebec City, August 11-15, 2002.
- [15] Draper, S.C.; Khisti, A.; Martinian, E.; Vetro, A.; Yedidia, J.S., *Using Distributed Source Coding to Secure Fingerprint Biometrics*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ISSN: 1520-6149, Vol. 2, pp. II-129–II-132, April 2007 (IEEE Xplore).
- [16] Hany H. Ammar, Xuhui Zhen, Diaa Nassar, Yingzi Jin, *Web-Based Test Bed for Fingerprint Image Comparison*. The World Internet and Electronic Cities Conference 2001, May 1-3, 2001 KISH ISLAND, IRAN , Received Best Paper Award.
- [17] Tomas Bulka, Chaitanya Agarwal, and Pavel Pozdnyakov, *SECURE MOBILE FINGERPRINT IDENTIFICATION SYSTEM*, Santa Clara, California 2005. URL: <http://www.scu.edu/engineering/cse/ugrad/upload/COEN-2005-PROJECT-01.pdf>
- [18] Hong, L., Wan, Y., Jain, A.K. (1998), *Fingerprint Image Enhancement: Algorithms and Performance Evaluation*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, 777–789.
- [19] Java CLDC specifications <http://java.sun.com/products/cldc/>
- [20] NMEA, "NMEA 0183 Interface Standard, v3.01.", *National Marine Electronics Association*, 2002.
- [21] Fallside, D. C. and Walmsley, P. (2004). XML Schema, W3C Recommendation 28 October 2004. Technical report, W3C
- [22] J. Tisal. "The GSM network: GPRS evolution", *John Wiley & Sons Ltd*, 2001
- [23] Taiyo Yuden, *Bluetooth Module EYMF2CMM -XX(Embedded SPP)*, Data Report, 1-Jun-2003 Ver.1.1
- [24] FTDI LTD., *ASIX, UMS2 – USB to UART interface module*, Manums2
- [25] Aimgene Technology Co., Ltd, *Aimgene BioFlex IV, Fingerprint Recognition Embedded System*, Version 1.03, Release Note 2005/09/02
- [26] GSM Security Algorithms, <http://www.gsmworld.com/using/algorithms/index.shtml>