



An Enhanced Classifier for Authentication in Keystroke Dynamics Using Experimental Data

**مصنف مطور للتحقق من الهوية في ديناميكية الكتابة على لوحة المفاتيح
باستخدام البيانات التجريبية**

By

Abdullah Osamah AL-Rahmani

Supervisor

Dr. Mudhafar M. AL-Jarrah

This Thesis is submitted in Partial Fulfillment of the Requirements for
the Master Degree in Computer Science Department Faculty of
Information Technology
Middle East University

June 2014

اقرار تفويض

أنني عبدالله أسامة الرحمانى أفوض جامعة الشرق الأوسط للتزويد نسخ من رسالتى للمكتبات أو المؤسسات أو الهيئات أو الأفراد عند طلبها.

التوقيع: 

التاريخ: ٢٠١٤ / ٦ / ١

Authorization statement

I Abdullah Osamah Al-Rahmani , Authorize the Middle East University to supply a copy of my thesis to libraries, establishments or individuals upon their request.

Signature:



Date: 17/6/2014

قرار لجنة المناقشة

نوقشت هذه الرسالة وعنوانها " مصنف مطور للتحقق من الهوية في ديناميكية الكتابة على لوحة المفاتيح باستخدام البيانات التجريبية " وأجيزت بتاريخ ٢٠١٤ / ٦ / ١

لجنة المناقشة

التوقيع

١- د. أوليج فيكتوروف رئيس اللجنة (جامعة الشرق الأوسط)

٢- د. مظفر منير الجراح مشرفاً (جامعة الشرق الأوسط)

٣- أ.د. علي أسعد أحمد الداود عضو اللجنة الخارجي (جامعة الزيتونة)

Committee decision

This is certifying that the thesis entitled “An Enhanced Classifier for Authentication in Keystroke Dynamics Using Experimental Data” and was successfully defended and approved on June first 2014.

Examination Committee Members

Signature

Dr. Oleg Viktorov.....



Associative Professor, Department of Computer Information System
(Middle East University)

Dr. Mudhafar M. Al-Jarrah



Assistant Professor, Department of Computer Science
(Middle East University)

Prof. Ali Asad Al-Dahoud



Professor, Department of Computer Science
(Al-Zaytoonah University)

Dedication

This thesis is dedicated to all the people who never stopped believing in me.

To my great father who never stopped supporting me during my life.

To my great mother who never stopped believing on me.

To all my brothers, Zaid, Ahmed , Ali, Hamza.

To my lovely wife and my daughter Amna and my son Haider.

Acknowledgments

I would like to thank my father and my mother for their continuous support during my study. I also would like to thank my great supervisor Dr. Mudhafar Al Jarrah and Dr.Hussien AL shimary and Dr.Oleg Viktorov and Prof. Ali Asad Al-Dahoud and Dr. Mohammed Al Nuaimi for their support, encouragement, and for helping me throughout my study. And never forget my all friends.

Table of Contents

Subject	Page
List of Figures	XI
List of Table	XII
List of Abbreviations	XII
Abstract	XIV
Chapter One	
1.1 Introduction	2
1.2 Authentication Definition	3
1.3 Biometrics Based Authentication	3
1.4 Statement of the Problem	4
1.5 Goal and Objectives	4
1.6 Methodology	5
1.7 Significance of Work	5
1.8 Thesis organization	7
Chapter Two	
2.1 Introduction	9
2.2 Keystroke Dynamic Based Authentication	10
2.3 Benefits of Biometric Based Authentication	12
2.4 KSD Measurements	14

2.5	KSD Performance Measures	15
2.6	KSD Families	17
2.6.1	The Static Families	17
2.6.2	The Dynamic Families	18
2.7	Review of Previous Classifiers in KSD	20
2.7.1	Euclidean	21
2.7.2	Euclidean (Normed)	22
2.7.3	Manhattan	22
2.7.4	Manhattan (Filtered)	23
2.7.5	Manhattan (Scaled)	23
2.7.6	Mahalanobis	24
2.7.7	Mahalanobis (Normed)	24
2.7.8	Nearest-Neighbor (Mahalanobis)	24
2.7.9	Neural-Network (Standard)	25
2.7.10	Neural-Network (Auto-Assoc.)	25
2.7.11	Fuzzy-Logic	26
2.7.12	Outlier-Counting (Z-Score)	26
2.7.13	SVM (One-Class)	26
2.7.14	K-Means	27
2.7.15	Medians Vector Proximity	27
2.8	Related Study	29

Chapter Three

3.1	Introduction	44
-----	--------------	----

3.2	The CMU dataset	44
3.3	The CMU feature set	45
3.4	Using the CMU Dataset for Model Evaluation	45
3.5	The Median-Std model	46
3.6	The Proposed Median-Median Model	47
3.7	The Keystroke Dynamic System	48
3.7.1	System overview	48
3.8	The development environment	49
3.8.1	Timing	49
3.9	Training procedure	50
3.10	Testing procedure	54
3.11	Data collection system	57
3.11.1	System overview	57
3.11.2	Data entry module	58
3.11.3	Sample data	59
3.11.3.1	Interface Execution	59

Chapter Four

4.1	Introduction	65
4.2	CMU benchmark dataset analysis using the MED-STD model	65
4.3	CMU benchmark dataset analysis using the MED-MED model	67
4.4	MEU Data Collection and Analysis	69

Chapter Five

5.1	Overview	72
5.2	Conclusion	72

5.3 Recommendations for Future Work	73
Reference	74

List of Figures

Figure No.	Figure Name	Page
Figure 2-1	Keystroke Dynamics	13
Figure 2-2	Concept Map on FAR & FRR	16
Figure 3-1	Med-Training-New Algorithm	52
Figure 3-2	Med-Text-Registered Algorithm	55
Figure 3-3	Proposed Model Diagram	58
Figure 3-4	first interface in our model	59
Figure 3-5	main interface	59
Figure 3-6	training stage interface	60
Figure 3-7	testing stage interface	61
Figure 3-8	admin interface	61
Figure 3-9	change variables interface	62
Figure 3-10	password calculates interface	62
Figure 3-11	statistics calculates interface	63
Figure 3-12	statistics calculates interface	63

List of Table

Table No.	Table Name	Page
Table 2-1	Comparison of Keystroke Dynamics Classifiers / Anomaly Detectors (Killourhy, 2012)	29
Table 4-1	Error Analysis of CMU dataset Med-Std Model (31 Features Set)	66
Table 4-2	Error Analysis of CMU dataset Med-Med Model (31 Features Set)	68
Table 4-3	Error Analysis of MEU dataset Med-Med Model (31 Features Set)	70

List of Abbreviations

Abbreviation	Meaning
ATM	Automated Teller Machine
PIN	Personal Identification Number
KSD	Key Stroke Dynamics
FT	Flight Time
FRR	False Rejection Rate
FAR	False Acceptance Rate

EER	Equal Error Rate
ROC	Receiver Operating Characteristic
CMU	Carnegie Mellon University
SVM	Support Vector Machine
CKAA	Clustering Keystroke Authentication Algorithm
AAN	Artificial Neural Network
MLP	Multilayer Perceptron
BP	Back Propagation
RBF	Radial Basis Function
EER	Equal Error Rate
PSO	Particle Swarm Optimization
GA	Genetic Algorithm
ACO	Ant Colony Optimization
BPNN	Back Propagation Neural Network
FAR	False Alarm Rates
IPR	Imposter Pass Rates
GPD	Gaussian Probability Density
DSM	Direction Similarity Measure
KDA	Keystroke Dynamic-based Authentication
IDE	Integrated Development Environment

C#	See Sharp
CLR	Common Language Runtime

Abstract

The problem of cyber-attacks on information systems and networks, for various illegal purposes, has become a major threat to society and individuals. Computer hackers are using all possible means to get access to private data, or to destroy such data. It has become necessary to improve computer security through more advanced access control mechanisms. Recently the use of biometrics has been employed to strengthen access control through user authentication that is based on users' measureable features.

The work in this thesis examines the keystroke dynamics approach, as a biometric authentication scheme that does not require extra hardware. The study is focused on enhancing an anomaly detector that is based on a statistical model of classifying the typing rhythm of a person who is trying to access a computer system, whether it is a genuine user or an imposter.

An anomaly detector model is proposed, which uses the median for each typing feature element of as the point of center to measure acceptance against, and a Distance-to-Median threshold value which gives the upper and lower limits for an acceptable feature element. The proposed model is evaluated using a public benchmark dataset of 20,400 records of password typing time measurement, collected by the Biometrics Lab of Carnegie Mellon University. The proposed model achieves lowest error rates of False Acceptance and False Rejection, compared to previous results of using other models on the same dataset.

A prototype keystroke dynamics authentication tool is developed, based on the proposed model. The tool has two parts: training, to collect typing data of a user and from that data generates a user identity template, and testing module, to be used as an authentication tool that uses the data collected during training. A discussion of the results of analyzing the benchmark data, and the data collected using the proposed models are discussed, and conclusions and suggestion for future work are presented.

Keywords: *keystroke dynamics, anomaly detector, user authentication, and error rate.*

الخلاصة

أصبحت الهجمات الإلكترونية على نظم المعلومات والشبكات، لأغراض غير قانونية، تشكل خطراً كبيراً على المجتمع والأفراد. يستخدم قراصنة الحاسوب جميع الوسائل للوصول إلى البيانات الشخصية، لسرقتها أو لغرض تدميرها. لذلك بات من الضروري تحسين أمن الحاسوب من خلال آليات متقدمة لمراقبة الدخول إلى نظام الحاسوب. إستخدمت مؤخراً المقاييس الحيوية الخاصة بالأشخاص لجعل الدخول إلى نظام الحاسوب آمناً وذلك من خلال التحقق من هوية المستخدم اعتماداً على الخصائص الخاصة به والتي تكون مسجلة مسبقاً.

يتناول البحث في هذه الأطروحة ديناميكية الكتابة على لوحة المفاتيح كطريقة حيوية وقياسية لا تتطلب أجهزة إضافية للتأكد من المستخدمين. تركز الدراسة على تحسين القدرة على كشف الدخلاء اعتماداً على نموذج إحصائي لتمييز إيقاع الكتابة للشخص الذي ينوي الدخول إلى نظام الحاسوب ، كمستخدم حقيقي أو كمتطفل. النموذج المقترح للكشف عن الدخلاء يستخدم الوسيط كنقطة في المنتصف لكل خاصية كتابة (وقت ضغط كل مفتاح ووقت الانتقال بين مفاتيح)، ومقدار البعد عن الوسيط ، لتكوين الحد الأعلى والحد الأدنى لكل خاصية ، وذلك لتحديد أن تكون الخاصية مقبولة أو مرفوضة. يتم تقييم النموذج المقترح باستخدام مجموعة بيانات قياسية مكونة من ٢٠,٤٠٠ قيد ، حيث يحتوي كل قيد على قيم الوقت لكل ضغط على مفتاح وانتقال بين مفاتيح لكلمة سر موحدة أدخلت من قبل ٥١ شخص ، تم جمعها في مختبرات جامعة كارنيجي ميلون الأمريكية.

النموذج المقترح يسجل نسبة خطأ أقل ونسبة قبول عالية للمستخدمين الحقيقيين ونسبة رفض عالية للدخلاء مقارنة مع النتائج التي تم الحصول عليها في بحوث سابقة باستخدام نفس البيانات القياسية.

في هذا البحث تم تطوير نظام أولي للتحقق من ديناميكية الكتابة على لوحة المفاتيح اعتماداً على النموذج المقترح . يتكون هذا النظام من جزئين : الأول يقوم بتدريب النظام على إيقاع الكتابة للمستخدم من خلال قيام المستخدم بإدخال كلمة السر عدد من المرات ، ومن ثم تخزين إيقاع الكتابة ، والجزء الثاني يقوم بوظيفة

التحقق من هوية المستخدم الذي يدخل كلمة السر لمرة واحدة، وذلك بمقارنة بيانات قياس الوقت المدخلة مع إيقاع الكتابة المخزون في النظام.

CHAPTER ONE

Introduction

1.1 Introduction

In computerized systems, a user is required to provide username and password in order to be allowed to access resources that restricted to authorized users, such as a database, a website or an information system. To withdraw money from an Automated Teller Machine (ATM), the user needs a bankcard and a Personal Identification Number (PIN) code. At some airports fingerprints or iris scans are already used to authenticate people. In general, most secure systems need to verify that the person using the system is indeed the one he or she claims to be.

All authentication methods such as password, token, biometric features, and so on have their own advantages and disadvantages, and the environment determines which authentication method is better suited. Any authentication mechanism can be static during the entry process to the system or dynamic. Dynamic mechanisms are better than static because all static authentication methods well known to the intruder, hackers or anyone need to threat the system. To authenticate the user during the moment that the authentication mechanism is executed: any change of user after that will be unnoticeable. Such authentication systems are referred as static authentication (Liu et al., 2009).

The Key Stroke Dynamics (KSD) is a method of authenticating a user based on his typing rhythm. It is based on timing of keystrokes, the time to press a key, and the travel time between successive keys. This method has its roots in the nineteenth century, when telegraph operators on navy ships used to be able to identify their counterparts by observing their typing speed and rhythm (Swets et al., 1982).

In this thesis, the author interested in the typing rhythm profile of a person; obtained through KSD. These features have the advantage that it doesn't require a special hardware, scanners or camera.

1.2 Authentication Definition:

It means to give verity, to impart to the instrument its validity and operative effect. Until it has been signed, sealed, and delivered, it is totally invalid, and all of these several acts must concur.

"Authenticate" mean to render authentic, to give authority to or proof, attestation or formality required by law as sufficient to entitle to credit.

Authentication serves three primary purposes. The first is correctly to identify those users who are authorized to access a resource such as a web page or database and to deny access to those who are not correctly identified. The second is to reassure the user that the resource is protected from access by unauthorized users. The third purpose is subtle, but its loss can result in a systematic loss of security when users attempt to ease the burden that typical authentication methods place on them by reusing passwords, writing them down, or using simple, easy to break passwords. It is the insufficient level of security provided by passwords (Al-Jarrah, 2012).

1.3 Biometrics Based Authentication:

Authentication is the features of an individual that can be used to authenticate his identity. These features can be grouped into three categories:

- Something you know such as a password, PIN and pattern.
- Something you have such as an ATM card.
- Something you are, a biometric feature on the person, physiological such as iris or fingerprint, or behavioral such as gait or typing rhythm.

The behavioral biometrics is focused on the ability to identify individuals uniquely and to associate personal attributes with an individual has been crucial to the fabric of human society. The set of attributes associated with a person constitutes their personal identity.

1.4 Statement of the Problem

The authentication issue is considered one of the main concepts in every computerized system for better performance in access control. There are different types to achieve it; one of these types is monitoring the behavioral biometric features such as the typing profile of the user for determining the user if authorized or not.

The problem in this thesis is how to enhance the KSD authentication scheme through empirical data collection and analysis of keystroke experimental data then analyzing the benchmark experimental KSD data to implement the proposed model as a prototype authentication tool.

1.5 Goal and Objective

The goal of this thesis is to enhance user authentication based on typing rhythm profile matching. The objective is to design a model for anomaly detection in typing rhythm using a statistical approach, and to implement the model as a prototype authentication KSD system for checking if a user is genuine or imposter. In this thesis, uses KSD authentication mechanism for measuring and analyzing a person's behavioral characteristics during authentication, by verifying and comparing his typing behavior with a stored template of his typing rhythm obtained during a training session.

1.6 Methodology

According to the technology revolution that occurred in our world especially in the security field that related to the computer and information technology. This field is very interesting for the researchers because of that this thesis will focus on the behavioral biometrics for providing suitable security to computer systems. The steps of the proposed model will include:

- 1- Building a web-based prototype KSD system that collects the timing data from user groups.
- 2- Collecting the keystroke data by using the prototype system on personal computer and standard keyboard.
- 3- Analyzing the collected data by using statistical measures and classifiers.
- 4- Evaluating the possible enhancements to the existing classifiers that enhance the anomaly detection performance.
- 5- Upgrading the prototypes system by adding the training data and the verification models that including the enhanced classifiers
- 6- Reviewing the achieved results, presenting some conclusions and recommendations for further development and future research.

1.7 Significance of Work

The rapid rise in hacking attacks with its serious consequences on individuals and businesses has made it necessary for several layers of security. The password has proven to be an easy target for hackers. A second authentication factor such as Google's One-Time-Password (OTP) has improved authentication to a certain point, but it is also vulnerable to be hijacked, an important factor of authentication that cannot be hacked because it is a biometric feature. The typing

rhythm of an individual is a useful second or third layer of authentication factor that can be measured without extra hardware.

KSD has received a significant amount of attention in authentication research circles because it has the possibility of providing a transparent, acceptable method of authentication that can be implemented using existing hardware. The researchers have continued to progress by performing studies on keyboard types, examining new pattern matching algorithms, and identifying new keystroke characteristics.

In any study of the KSD approach, the significance of the work is in enhancing authentication performance, i.e. the accuracy of distinguishing between genuine users and illegitimate users or imposters. To this end, the present work attempts to make a modest contribution to the enhancement authentication accuracy using KSD.

1.8 Thesis Organization

This thesis is divided into five chapters:

- Chapter One contains general concepts of this thesis that includes the authentication schemes and methods, statement of problem, goal and objective, methodology and significance of work.
- Chapter Two contains the literature survey of the fields of biometrics and KSD, and the related work.
- Chapter Three contains the proposed KSD anomaly detection model, the feature set, error metrics, and the KSD software that implements the KSD model.
- Chapter Four presents the results and discussion of using the proposed model in analyzing a benchmark KSD, and the results of using the KSD system.
- Chapter Five contains conclusions and future work.

CHAPTER TWO

Biometrics and Authentication with Literature Study

2.1 Introduction

The importance of protection is increased during this decade because any development occurred in technology field includes positive or negative factors, the factors can be appeared when the user process in any computerized system, then this process is determined if the factors is positive or negative depending on uses.

The negative factor is represented to provide the authentication process to different system, the authentication process which means the operation of verifying whether the identities of user are authentic.

There are different types of authentication process, but these types are common based on three categories, the first one is something you know (usually a password). The next category is something has (ATM, Smartcard) and final category is based on something that a person is (Fingerprint, Palm print).

The important type of authentication is the third category because it is uniquely recognizing humans based upon one or more intrinsic physical or behavioral traits, which gives a lot of methods in high-secure applications while using natural, user-friendly and fast authentication. Therefore we will use third category of behavioral traits for determined the user who access to the system if authorized or not.

In this thesis, it will use the Keystroke dynamics it is a widely biometric technique that can be easily implemented in any system without need of any external hardware. Which analyzes the way a user types by monitoring the keyboard inputs in attempt to identify them by their habitual typing patterns.

It will focus on developing the KSD authentication scheme; the features are influenced by empirical collected data from experiments. The planning in this thesis, it collect the typing data for a group users by using a prototype system.

2.2 Keystroke Dynamic Based Authentication

User authentication is a major problem in gaining access rights for computer resources. Many biometric properties of users such as iris, fingerprint and palm print are used to provide additional security. Since the KSD based user authentication approaches do not require the aid of extra special tools, keystroke analysis has been an active research topic for more than three decades. Many keystroke analysis approaches were proposed. Some of them formulated the keystroke dynamics-based user authentication into a binary classification problem. In these approaches, both the imposter's patterns and the legitimate user's patterns were used for training. This is unrealistic in practice since there are millions of potential imposters. It is not possible to obtain all the prospective imposter patterns.

KSD is a biometric authentication technique that is based on the assumption that different people type in uniquely characteristic manners. Observation of telegraph operators in the 19th century revealed personally distinctive patterns when keying messages over telegraph lines, and telegraph operators could recognize each other based on only their keying dynamics (Obaidat and Sadoun, 1997).

Conceptually closest correspondence among biometric identification systems is signature recognition. In both signature recognition and KSD the person is identified by their writing dynamics which are assumed to be unique to a large degree among different people. KSD is known with a few different names: keyboard dynamics, keystroke analysis, typing biometrics and typing rhythms (Miller, 1994).

Keystroke features can be extracted in terms of:

- Dwell Time - how long a key is pressed.
- Flight Time (FT) - how long it takes to move from one key to another.
- Difficulties of typing phrase.
- Pressure of keystroke.
- Typing rate.
- Linguistic style.
- Sound of typing.
- Frequency of word errors.

Nevertheless, not all of the above features are favorable. For example, in order to acquire keystroke pressure feature, dedicated pressure sensitive keyboard is essential, which contradicts with the main advantage of KSD biometrics. Frequency of word errors, typing rate, and difficulties of typing phrase are merely practical for text with large number of characters. Where else, there is a high concern with the noise associates with the acquisition devices used to record sound of typing.

The behavioral biometric of KSD uses the manner and rhythm in which an individual types characters on a keyboard or keypad (Araujo et al., 2005).

The keystroke rhythms of a user are measured to develop a unique biometric template of the users typing pattern for future authentication (Panasiuk and Saeed, 2010).

Raw measurements available from almost every keyboard can be recorded to determine Dwell time (the time a key pressed) and Flight time (the time between "key up" and the next "key down"). The recorded keystroke timing data is then processed through a unique neural algorithm, which determines a primary pattern for future comparison (Chang et al., 2011).

2.3 Benefits of Biometric Based Authentication

Access to computer systems is usually controlled by user accounts with usernames and passwords. Such scheme has little security if the information falls to wrong hands. Key cards or biometric systems, for example fingerprints, can be used to strengthen security, but they require quite expensive additional hardware. On the other hand KSD can be used without any additional hardware. Also, the user acceptance of a KSD biometric system is very high, since users do not necessarily even notice that such system is used. KSD is mostly applicable to verification, but not for identification. Verification system has normally required user identification details and is attempting to verify his claimed identity. Most applications of keystroke dynamics are infield of verification.

KSD was a user friendly biometric authentication technique and already there are KSD Based Human Authentication Systems using Genetic Algorithm available for online applications; web based emailing and other online services. It minimizes the impact on the user's privacy and was very simple to integrate. The keystroke pattern recognition technique could be used effectively as a safeguard to unauthorized access to computer resources and sensitive data (Gunathilake et al., 2013).

KSD also is a very cheap biometric verification method because there is no need for any additional hardware besides a normal keyboard. Using keystroke dynamics makes a username, password-based authentication procedure significantly more secure, as shown in Figure 2-1.

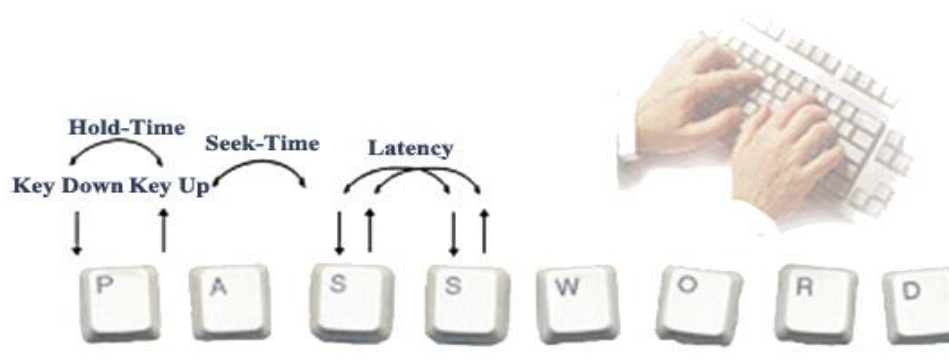


Figure 2-1 Keystroke Dynamics

2.4 KSD Measurements

KSD include several different measurements which can be detected when the user presses keys in the keyboard. Possible measurements include Latency between consecutive keystrokes, duration of the keystroke i.e. hold-time, overall typing speed, frequency of errors (user need use backspace), habit of using additional keys in the keyboard for example writing numbers with the number-pad, in what order does the user press keys when writing capital letters must use shift key or the letter key released first or the force used when hitting keys while typing (requires a special keyboard).

Statistics can be either global, i.e. combined for all keys, or they can be gathered for every key or keystroke separately. Systems do not necessarily employ all of these features. Most of the applications measure only latencies between consecutive keystrokes or durations of keystrokes.

The dynamics of a user's interaction with a keyboard input device yields quantitative information with respect to dwell time (how long a key is pressed) and time-of-flight (the time taken to enter successive keys). By collecting the dynamic aspects acquired even during the login process, one can develop a model that captures potentially unique characteristics that can be used for the identification of an individual. To facilitate the development of the model of how the user enters their details, an enrollment phase is required, when the user is asked to enter his/her login and id/password until a steady value is obtained (usually limited to 10-15 trials – but this is implementation dependent).

Once this data has been collected, a reference ‘signature’ is generated for this user. The reference signature is then used to authenticate the user account on subsequent login attempts. The user with that particular login id/password combination has their KSD extracted and then compared with the stored reference signature. If they are within a prescribed tolerance limit – the user is authenticated. If not - then the system can decide whether to lock up the workstation - or take some other suitable action (Revett, 2011).

2.5 KSD Performance Measures

The authentication performance of KSD systems used to evaluate the system's ability to classify a user as either genuine "legitimate" or imposter. KSD systems use various techniques and algorithms from machine learning for the classification task (Al-Jarrah, 2012).

- **False Rejection Rate (FRR)**, the percentage of genuine user attempts identified as impostors, which is equivalent to Error Type 1 in signal processing (Swets and Pickett, 1982). An equivalent term is False-Alarm Rate.
- **False Acceptance Rate (FAR)**, the percentage of impostor access attempts identified as genuine users, which is Error Type 2 in signal processing. An equivalent term is Miss Rate. In this work we are using False-Alarm Rate for FRR and Miss Rate for FAR, to maintain consistency with the terminology used in the CMU benchmark (Killourhy and Maxion, 2009).

- **Equal Error Rate (ERR)**, the crossover point in the ROC curves at which False-Alarm and Miss Rates are equal. The Receiver Operating Characteristic (ROC) curve, a signal detection theory tool, is a graphical plot which illustrates the authentication performance of a binary classifier system as its discrimination threshold is varied. The curve is created by plotting the fraction of Hit Rate vs. the fraction of False-Alarm Rate, as shown in Figure 2-2.

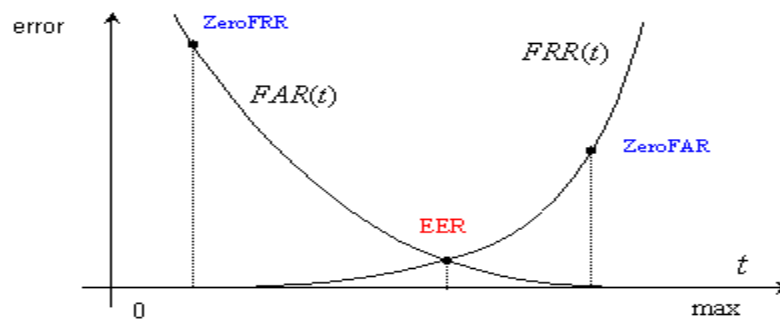


Figure 2-2 Concept Map on FAR & FRR

Although these metrics convey useful information about a biometric system's performance, they don't capture any temporal information about the occurrence of false rejections. For biometric modalities in which the possibility of false rejections is high (such as behavioral biometrics, including KSD, voice, and mouse dynamics), timing information on when the errors (such as false rejections) occur can help practitioners more thoroughly evaluate system performance.

By leveraging information about the temporal distribution of classification errors, a practitioner could make several critical inferences regarding system performance. For example, a continuous sequence of false rejections could signify that the user's template has aged and needs

an update mechanism for tracking variations in the user's features. For a system that already employs a template update policy, a similar distribution of errors could indicate that template-poisoning attacks (such as the Frog-boiling attack³) have altered the user's template. A sudden burst of false rejections could, on the other hand, suggest that the system has been subjected to perpetual short-term perturbations, such as changes in the environmental conditions (a change in the user's keyboard, for example (Serwadda et al., 2013)).

2.6 KSD Families

2.6.1 The Static Families

The user is asked to type several times the same string in order to build its model. During the authentication phase, the user is supposed to provide the same string captured during his enrollment. Such methodology is really appropriate to authenticate an individual by asking him to type its own password, before login to its computer session, and verifying if its way of typing matches the model. Changing the password implies to enroll again, because the methods are not able to work with a different password.

In static verification, the keystrokes are analyzed only at specific times e.g., during login. Static approaches provide more robust user verification than simple passwords but static methods do not provide continuous security. They cannot detect substitution of the user after the initial verification (Monrose and Rubin, 2000).

2.6.2 The Dynamic Families

Allow to authenticate individuals independently of what they are typing on the keyboard. Usually, they are required to provide a lot of typing data to create their model (directly by asking them to type some long texts, or indirectly by monitoring their computer use during a certain period).

In this solution, the user can be verified on the fly all the time he uses its computer. It can detect a changing of user during the computer usage. This is related as continuous authentication in the literature. When the users are able to model the behavior of a user, whatever the thing he or she types, it can also authenticate him/her through a challenge during the normal login process: it can ask the user to type a random phrase, or a shared secret such as a one-time password(Giot et al, 2009).

Researchers are interested in using the KSD information, which is normally discarded, to verify or even try to determine the identity of the person who is producing those keystrokes. This is often possible because some characteristics of keystroke production are as individual as Handwriting or a Signature. The techniques used to do this vary widely in power and sophistication, and range from statistical techniques to artificial neural networks.

In the simplest case can be used to rule out a possible user. For example, if the user knows that John types at 20 Words per minute, and the person at the keyboard is going at 70 words per minute, it's a pretty safe bet that it's not John. That would be a test based simply on raw speed uncorrected for errors. It's only a one-way test, as it's always possible for people to go slower than normal, but it's unusual or impossible for them to go twice their normal speed. Or, it may be that the mystery user at the keyboard and John both type at 50 words per minute; but John never

really learned the numbers, and always has to slow down an extra half-second whenever a number has to be entered. If the mystery user doesn't slow down for numbers, then, again, it's a safe bet this isn't John.

The time to get and to depress a key (seek-time), and the time the key is held-down (hold-time) may be very characteristic for a person, regardless of how fast they are going overall. Most people have specific letters that take them longer to find or get to than their average seek-time over all letters, but which letters those are may vary dramatically but consistently for different people. Right-handed people may be statistically faster in getting to keys they hit with their right hand fingers than they are with their left hand fingers. Index fingers may be characteristically faster than other finger to a degree that is consistent for a person day-to-day regardless of their overall speed that day.

In addition, sequences of letters may have characteristic properties for a person. In English, the word "the" is very common, and those three letters may be known as a rapid-fire sequence and not as just three meaningless letters hit in that order. Common endings, such as "ing", may be entered far faster than, say, the same letters in reverse order ("gni") to a degree that varies consistently by person. This consistency may hold and may reveal the person's native language's common sequences even when they are writing entirely in a different language, just as revealing as an accent might in spoken English.

Common "errors" may also be quite characteristic of a person, and there is an entire taxonomy of errors, such as this person's most common "substitutions", "reversals", "drop-outs", "double-strikes", "adjacent letter hits", "homonyms", hold-length-errors (for a shift key held down too short or too long a time). Even without knowing what language a person is working in,

by looking at the rest of the text and what letters the person goes back and replaces these errors might be detected. Again, the patterns of errors might be sufficiently different to distinguish two people.

Recently, several researchers have worked on developing models and systems for use of KSD for authentication, both static and dynamic. (Giot et al., 2009), it provides a recent review in order to compare the performance of various model source researchers have published benchmark data using pressure sensitive keyboards (Allen pressure).

2.7 Review of Previous Classifiers in KSD

Many models and algorithms for classifier in KSD systems, a review of those can be found in the work of Killourhy and Maxionat Carnegie Mellon University (CMU) (Killourhy, 2012), has provided a comprehensive benchmark dataset for keystroke dynamics research.

The historical favorite has been pattern classifiers, which are a form of statistical classifier. Commonly used statistical classifiers found in the literature to date include Mahalanobis distance, Hamming distance, Euclidean distance, etc.

More recently, studies have begun to use neural networks as a pattern classification method. Common neural network approaches include Feed Forward Multilayered Perceptron Networks (with and without back propagation), Radial Base Function Networks and Generalized Regression Networks.

The experimental results and related analyses have aimed to serve as an independent comparison reference data for the evaluation of classifiers and anomaly detection algorithms. The benchmark dataset contains 20,400 timing records where each record consists of 31 timing

measures obtained during the typing of the same password. A strong 10-character password (.abu5gathum) was used in the entire benchmark. The data reflects typing records of 51 subjects, of different backgrounds, each subject typed 400 repetitions of the password over 8 sessions of 50 repetitions each (spread over different days). The benchmark data is used in for the comparative study of the detection performance of 14 published algorithms, references (mention as below) (Killourhy and Maxion, 2009).

The work involved implementing the 15 algorithms and conducting performance measurement based on the benchmark dataset, as discussed in subsection below with the table that declared the comparison between the 15 algorithms (Killourhy, 2012), as shown in table 2-1.

2.7.1 Euclidean

In this models each password as a point in p -dimensional space, where p is the number of features in the timing vectors. The mean vector of the set of timing vectors is calculated. In the test phase, the anomaly score is calculated as the squared Euclidean distance between the test vector and the mean vector.

In this method, the template is simply the list of enrolled samples. This distance is computed between the test vector and each of the enrolled samples. The score is then the minimum computed distance.

Choosing p -norm distance is due to its low computational complexity and relatively good performance. It is desirable that a keystroke authentication system can response user input in real-time. So the distance measure should not be too expensive to compute. In p -norm distance, if the value of p is 1 then it is Manhattan distance, called as 1-norm distance as well. If p equals 2 then it is Euclidean distance, or 2-norm distance.

So a processed user pattern contains only four values, Manhattan distance for elapse time, Manhattan distance for duration, Euclidean distance for elapse time and Euclidean distance for duration. In the context of learning, a pattern can be viewed as a vector of four dimensions.

In 1997 Monroe and Rubin use the Euclidean Distance and probabilistic calculations based on the assumption that the latency times for one-digraph exhibits a Normal Distribution (Monroe and Rubin, 1997).

In 2000, they also presented an algorithm for identification, and in 2001 they presented an algorithm that uses polynomials and vector spaces to generate complex passwords from a simple one, using the keystroke pattern (Monroe and Rubin, 2000). Euclidean distance has been the default distance metric for its simplicity and geometrical intuitiveness. However, it has two major limitations:

1. It is very sensitive to scale variations in the feature variables.
2. It has no means to deal with the correlation between feature variables.

2.7.2 Euclidean (Normed)

This detector was described by (Bleha et al. 1990) who called it the “normalized minimum distance classifier.” In the training phase, the mean vector is calculated as in the standard Euclidean detector.

2.7.3 Manhattan

This resembles the Euclidean detector except that the distance measure is Manhattan (or city-block) distance. In the training phase, the mean vector of the timing vectors is calculated. In the test phase, the anomaly score is calculated as the Manhattan distance between the mean vector

and the test vector. Manhattan distance is used to find the distance between referring keystroke feature vector and the feature vector to be classified.

The Manhattan distance has the advantages of simplicity in computation and easy decomposition into contributions made by each variable. Most importantly, it is more robust to the influence of outliers compared to higher order distance metrics including Euclidean distance and Mahalanobis distance. As a result, Manhattan distance is more robust than Mahalanobis distance in the presence of outliers. The Manhattan distance also has a statistical interpretation as the Mahalanobis distance. It is in fact related to the log likelihood of the multivariate Laplace distribution with an identity covariance matrix (Zhao, 2006).

2.7.4 Manhattan (Filtered)

This detector was described by (Joyce and Gupta, 1990). It is similar to the Manhattan detector except outliers are filtered from the training data. In the training phase, the mean vector of the timing vectors is calculated, and the standard deviation for each feature is calculated also.

2.7.5 Manhattan (Scaled)

This detector was described by (Araujo et al., 2004) in the training phase, the mean vector of the timing vectors is calculated, and the mean absolute deviation of each feature is calculated as well.

2.7.6 Mahalanobis

This resembles the Euclidean and Manhattan detectors but the distance measure is more complex. Mahalanobis distance can be viewed as an extension of Euclidean distance to account for correlations between features.

Mahalanobis distance, on the other hand, takes into account the covariance of data variables to correct for the heterogeneity and non-isotropy observed in most real data. It not only weights the distance calculation according to the statistical variation of each feature component, but also decouples the interactions between features based on their covariance matrix to provide a useful distance metric for feature comparisons in pattern analysis. In statistical literature, the Mahalanobis distance is related to the log likelihood under the assumption that data follow multivariate Gaussian distribution which is a reasonable approximation for most practical data (Balagani et al., 2011).

2.7.7 Mahalanobis (Normed)

This detector was described by (Bleha et al. 1990) who called it the “normalized Bayes classifier.” In the training phase, the mean vector and covariance matrix of the training vectors are calculated.

2.7.8 Nearest-Neighbor (Mahalanobis)

This detector was described by (Cho et al., 2000). In the training phase, the detector saves the list of training vectors, and calculates the covariance matrix.

When using the Nearest Neighbor classifier with the new distance metric defined in to either ascertain a keystroke dynamics feature as originating from the genuine user when the distance to its nearest neighbor in the training data is below a threshold value, or reject it as an imposter, otherwise. The covariance matrix is computed using all the training keystroke feature vectors from the intended user (Hu et al., 2008).

2.7.9 Neural-Network (Standard)

This detector was described by (Haider et al., 2000). It incorporates a feed-forward neural-network trained with the back-propagation algorithm.

More recently, studies have begun to use neural networks as a pattern classification method. Common neural network approaches include Feed Forward Multilayered Perceptron Networks (with and without back propagation) (Karatzouni and Clarke, 2007).

Radial Base Function Networks and Generalized Regression Networks have been noted by Clarke et al. that neural networks are a superior pattern classification method, but that mobile devices lack the computing power necessary to employ a neural network in situations where the processing is done on the device itself.

2.7.10 Neural-Network (Auto-Assoc.)

This detector was described by (Cho et al., 2000) who called it an “auto-associative, multilayer perceptron.” It also incorporates a feed-forward neural-network trained with the back-

propagation algorithm, but unlike a typical neural network, the structure of the network is designed for use in an anomaly detector.

The auto-associative neural network was trained in advance not only with the owner's timing vectors but also with those of imposters. In real life situations, this is unacceptable because the owner's password has to be revealed to users at large. A longer password will simply require a neural network with more input and output units (Cho et al., 2000).

2.7.11 Fuzzy-Logic

This detector was described by (Haider et al., 2000). It incorporates a fuzzy-logic inference procedure. The key idea is that ranges of typing times are assigned to fuzzy sets (e.g., the times in the range of 210–290 milliseconds are part of a set named “very fast”).

Fuzzy clustering is a class of algorithm for Cluster analysis in which the allocation of data points to clusters is not "hard" (all-or-nothing) but "fuzzy" in the same sense as Fuzzy logic (Zhong et al., 2012).

2.7.12 Outlier-Counting (Z-Score)

This detector was described by (Haider et al., 2000) who called it the “statistical technique.” In the training phase, the detector calculates the mean and standard deviation of each timing feature. In the test phase, the detector computes the absolute z-score of each feature of the test vector.

2.7.13 SVM (One-Class)

This detector was described by (Yu and Cho, 2003). It incorporates an algorithm called a Support-Vector Machine (SVM) that projects two classes of data into a high dimensional space and finds a linear separator between the two classes.

The KSD based authentication as a one-class classification problem which learns a model for a user, rejects anomalies to the learned model as imposters, and accept inliers as the genuine user. Although the use of negative examples in training could significantly improve the accuracy of the classifier, it is unrealistic to assume prior knowledge about the keystroke features from imposters, let alone the availability of their training data (Giot et al., 2009).

2.7.14 K-Means

This detector was described by (Kang et al., 2007). It uses the k-means clustering algorithm to identify clusters in the training vectors, and then calculates whether the test vector is close to any of the clusters.

K-Means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cell (Zha et al., 2001).

2.7.15 Medians Vector Proximity

This model presented a classifier anomaly detector to compute a score for the typing of a password to determine authenticity. A measure of proximity is used in the comparison between feature set medians vector and feature set testing vector (Al-Jarrah, 2012).

The performance score for each algorithm is calculated, and the algorithms are compared in terms of performance score. The detection performance is calculated from two error rate measures: False-Alarm Rate, which represents the ratio of rejects of a genuine user, and the Miss Rate, which represents the false acceptance of an impostor (other authors use FRR for False-

Alarm Rate and FAR for Miss Rate). The CMU benchmark is chosen as the evaluation dataset in this paper for several reasons (i) it has already been used in comparative studies of the detection performance of several algorithms, (ii) it uses a strong password, and (iii) the measurement includes all typed keys, including the return key. Other studies have included pressure and force of a key-press in the determination of typing profile, which requires the use of special pressure-sensitive keyboards. This is outside the scope of our work. An important and independent reference for the evaluation of anomaly detection performance is the European Standard for Access Control Systems (EN-5013301) (CENELEC. European Standard, 2002). Alarm systems, access control systems for use in security applications], which specifies a False-Alarm Rate of less than 1.0% (i.e. one rejection per one hundred entries by a genuine user), and less than 0.001% for Miss Rate (i.e. one acceptance of an impostor in 100,000 entries). This implies that the standard considers a false acceptance of an impostor to be 1000 times more serious than rejecting a genuine user.

Table 2-1 Comparison of Keystroke Dynamics Classifiers / Anomaly Detectors(Killourhy, 2012)

S.no	ALGORITHM	EER Avg.	EER Std. Dev.
1	Medians Vector Proximity	0.080	0.060
2	Manhattan (scaled)	0.096	0.069
3	Nearest Neighbor (Mahalanobis)	0.100	0.064
4	Outlier Count (z-score)	0.102	0.077
5	SVM (one-class)	0.102	0.065
6	Mahalanobis	0.110	0.065
7	Mahalanobis (normed)	0.110	0.065
8	Manhattan (filter)	0.136	0.083
9	Manhattan	0.153	0.092
10	Neural Network (auto-assoc)	0.161	0.080
11	Euclidean	0.171	0.095
12	Euclidean (normed)	0.215	0.119
13	Fuzzy Logic	0.221	0.105
14	K Means	0.372	0.139
15	Neural Network (standard)	0.828	0.148

2.8 Related Study

➤ (Hu et al., 2008).

Keystroke dynamics exhibit people's behavioral features which are similar to hand signatures. A major problem hindering the large scale deployment of this technology is its high FAR and FRR and the identification based authentication suffers a severe scalability issue as it needs to verify the input with every training sample of every user within the whole database. In this paper, a k-nearest neighbor approach has been proposed to classify users' keystroke dynamics profiles and the proposed Clustering Keystroke Authentication Algorithm (CKAA) has

solved the scalability problem suffered by the researcher method while can achieve the same good performance in terms of FAR and FRR.

The result in this paper can further help to advance this technology towards practical applications and the experiments have been conducted which has validated the proposed scheme, keystroke dynamic authentication has its unique advantages such as inexpensive and universal available. These experiments have demonstrated the same level of FAR and FRR while as high as 66.7% improvement of the authentication speed has been achieved.

➤ **(Rybnik et al., 2009).**

This paper focused on authentication methods based on most biometrics techniques require dedicated hardware that is unhandy for remote applications. The keystroke dynamics requires only limited dedicated software and does not require any dedicated hardware. The approach they proposed is aimed at efficient user authentication with keystroke dynamics using short fixed text that may for example occur simultaneously to logging process. They have tested the suggested approach on a large group of individuals, with data gathered over Internet.

With analysis of two keystrokes features and with the use of relatively simple classification techniques the keystroke dynamics proved to be promising and effective biometrics for identification and authentication of individuals. It is necessary to stress that with use of constant text it is possible to effectively distinguish a vast majority of users with a relatively short keystrokes sequence (beginning from 9 keystrokes). It is necessary to stress the advantages: possibility to use a booming Internet as a natural transmission medium for the biometrics and the lack of need for dedicated acquisition devices. The obtained results are promising and encourage further development in this area and clearly showed the potential of keystroke biometrics for the identification and authentication of individuals over the Internet.

➤ (Giot et al., 2009).

This paper declared the authentication issue which means challenging issue in order to guarantee the security of use of collaborative systems during the access control step. Many solutions exist in the state of the art such as the use of one time passwords or smart-cards, this paper focused on biometric based solutions that do not necessitate any additional sensor.

The authors studied the ability of keystroke dynamics authentication systems which an interesting solution as it uses only the keyboard and is invisible for users to their application to collaborative systems (collaborative systems need to authenticate there users), and they can see through this study that, even with quite simple methods from the state of the art, the obtained results are almost correct with less than 5% of error but need yet to be improved. One-class support vector machines with only five vectors per users for the training seem to give better results.

In the tested systems, only well typed passwords are taken into account. This is a real problem, because with static password based authentication methods, the genuine user can correct himself his typing errors and being correctly authenticated. On the contrary, with the keystroke dynamic implementation, the system will force the user to type again the password in order to have a correct sized vector. The system and algorithms have to be modified to allow the use of backspace key to correct the password (because errors can be characteristic of the user). There is a lack of data in the analysis of the subjective evaluation. It gives better results to compute the robustness of all the algorithms. This could also give more information on the way of how to interpret this kind of curves.

➤ **(Pedernera et al., 2010).**

This paper discussed the keystroke dynamics is a set of computer techniques that has been used successfully for many years for authentication mechanisms and masqueraders detection. Classification algorithms have reportedly performed well, but there is room for improvement. This paper focused to classify intruders by analyzing their keystroke patterns and it is presented a novel approach to intruder classification using real intrusion datasets and focusing on intruder's behavior. They computed six distance measures between sessions to cluster them using both modified K-means and Subtractive Clustering algorithms. The distance measures used features that came from the relation between intruder's sessions, instead of using features from each user only.

The results showed an improvement of 70% compared to the former method that used features from single sessions. The ongoing work includes the acquisition of labeled data, leaving out some limitations of real captures also testing the adapted methods with a complete set of performance evaluation functions.

➤ **(Harun et al., 2010).**

This paper showed that keystroke is a special behavioral biometric that can be used as features for an additional and transparent layer of user authentication. This paper addressed the issue of enhancing such systems using keystroke biometrics as a translucent level of user authentication. The paper focused on using the time interval (key down-down) between keystrokes as a feature of individuals' typing patterns to recognize authentic users and reject imposters. A Multilayer Perceptron (MLP) neural network with a Back Propagation (BP) learning algorithm is used to train and validate the features. The results are compared with a

Radial Basis Function (RBF) neural network and several distance classifier method used based on Equal Error Rate (EER).

The simulation results revealed that MLP with BP network is more suitable to discriminate and classify a nonlinear keystroke database as low as 2% in EER. It also showed that MLP with the BP algorithm showed greater promising result for improving EER in order to verify the authorized user as compared to RBF network. Even though the result for the free text (Database) is high (23%) compared with others databases, however, it is still proved that MLP/BP outperforms the RBF. Moreover this work proved that MLP gives better accuracy and improvement in classifying nonlinear data compared to linear classifier. It can be concluded that distance classifiers is not good enough to classify groups of keystroke biometrics. This paper achieved 5% to 19% EER values for non-equalization data but with equalized data EER much better 4% to 10%. However these classifiers not suitable to classify or authenticate time intervals of typing biometrics because the distance of every feature is very close each other.

➤ **(Karnan and Akila, 2010).**

This paper declared the passwords technique have been the usual method for controlling access to computer systems but this approach has many inherent flaws. Keystroke dynamics are rich with individual mannerisms and traits and they can be used to extract features that can be used to identify computer user. The features are extracted from 27 users with 100 samples of each in a week period. Using the samples, the mean, standard deviation and median is calculated for duration, latency, digraph and their combinations. Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and the proposed Ant Colony Optimization (ACO) are used for feature subset selection. Back Propagation Neural Network (BPNN) is used for classification. ACO

gives better performance than PSO and GA with regard to feature reduction rate and classification accuracy. Using digraph as the feature for feature subset selection is novel and show good classification performance.

The results for this paper found that using the values of digraph and back-propagation neural network algorithms has shown excellent verification accuracy. The average classification error is reduced when the number of sample is increased. The average classification error of PSO is 0.063% and GA is 0.078%. The average accuracy of PSO is 88.9% and GA is 86.6%. ACO gives best classification accuracy and average error rate. The average classification error of ACO is 0.059% and the accuracy is 92.8%.

➤ **(Crawford, 2010).**

This paper declared the studies have progressed from examining typing patterns on desktop keyboards using statistical pattern classifiers to mobile keyboards using neural networks as a pattern classifier. The studies do not have a unifying method of comparing results, which limits comparison between the methods presented. Without the ability to compare studies within research areas, future study is limited in its ability to provide important modifications to the work in question.

This paper reviews a representative subset of the current research in keystroke dynamics, which has received a significant amount of attention in authentication research circles because it has the possibility of providing a transparent, acceptable method of authentication that can be implemented using existing hardware.

Also this paper explains the researchers have adopted commonly used comparison methods such as FAR, FRR, and EER to quantitatively show improvements in keystroke dynamics

methodology. From these studies, the authors have learned that it is unlikely that keystroke dynamics alone will be robust enough to uniquely identify users, but it shows great promise as a part of a larger multimodal biometric authentication method.

➤ **(Teh et al., 2010).**

This paper describes a novel technique to strengthen password authentication system by incorporating multiple keystrokes dynamic information under a fusion framework, using keystroke dynamics as an additional biometric security entity is reliable, fast, non-invasive and cost effective. This additional security mechanism is invisible to users as they only need to type on a keyboard as usual, instead of providing their physical biometric data (i.e. fingerprint, iris image), where some users may feel uncomfortable. The security mechanism will be responsible to capture and process the users' typing timing patterns without their knowledge. Ease of incorporation into the existing password based authentication system and no additional device requirement increase the potential of keystroke dynamics to be deployed in real world applications, such as online banking and password based security system.

They capitalized four types of latency as keystroke feature and two methods to calculate the similarity scores between the two given latency. A two layer fusion approach is proposed to enhance the overall performance of the system to achieve near 1.401% EER. They also introduced two additional modules to increase the flexibility of the proposed system. These modules aim to accommodate exceptional cases, for instance, when a legitimate user is unable to provide his or her normal typing pattern due to reasons such as hand injury.

➤ **(Sahoo and Deb 2010).**

The aim of this paper is to develop a security system for mobile devices that eliminates the vulnerability of PIN (4-digit) security and it is less resource and memory consuming than the biometrics methods (often an extra device is required in biometrics). The authors will be handling in the project work is the vulnerability of PIN security in mobile devices and providing an extra layer of security through KSD based authentication system. Also the problem of authentication error rates while using natural user keystroke pattern will be worked upon. And a better mechanism which uses an artificial keystroke rhythm with cues will be worked upon.

Here the project is made using only a limited group of users. There could be a problem in fixing an accurate threshold value, there for the FAR or FRR may vary inconsistently.

Security is a matter of concern for mobile devices. But a security method that is applicable to the general mobile users, users of diverse groups should be developed keeping in mind the limited configuration, resources and memory that can be used in mobile devices.

➤ **(Karnan et al., 2011).**

This paper is to summarize the well-known approaches used in keystroke dynamics, the concept of keystroke dynamics is not limited to the traditional keyboard but any interface in which keys must be pressed can benefit from similar techniques. Keystroke biometrics has also shown great potential as the features can be collected without the need for special hardware.

The authors attempted to provide a comprehensive survey of research on keystroke dynamics described in the last two decades. When appropriate, the relative performance of methods is reported. But, in so doing, it is cognizant that there is a lack of uniformity in how methods are

evaluated and, so, it is imprudent to explicitly declare which methods indeed have the lowest error rates.

The techniques were categorized based on the features, feature extraction methods and classification methods employed and the performance has been discussed. The research is needed to reduce both False Alarm Rates (FAR) and Imposter Pass Rates (IPR) to levels which will be acceptable to the user.

Keystroke biometrics has an advantage over most of other biometric authentication schemes, namely, user acceptance. Since users are already accustomed to authenticating themselves through usernames and passwords, most proposed keystroke biometric methods are completely transparent. There are numerous applications which can benefit from its success, and additional studies will further validate its use as an identity verifier. This is especially relevant to the popularity of keyboards as a primary input device in data processing systems.

➤ **(Giot et al., 2011).**

This paper proposed a new method based on the SVM learning satisfying industrial conditions in other words few samples per user are needed during the enrollment phase to create its template. In this method, users are authenticated through the keystroke dynamics of a shared secret (chosen by the system administrator). They used the GREYC keystroke database that is composed of a large number of users (100) for validation purposes.

Experimental results in this paper the computation time to build the template can be longer with the method (54 s against 3 s for most of the others), its performance outperforms the other methods in an industrial context (Equal Error Rate of 15.28% against 16.79% and 17.02% for the

two best methods of the state-of-the-art, on the dataset and five samples to create the template, with a better computation time than the second best method).

The benefit of supervised template update mechanisms of the biometric reference was also demonstrated. Several factors have to be tested in the keystroke dynamics domain. This often implies creating a new database especially designed for the corresponding tests (i.e., dependency on the keyboard, computer operating systems, knowledge of the password, size of the password, content of the password). These databases can be created by merging different databases from different researchers or by creating new ones with the help of GREYC-Keystroke software.

➤ **(Teh et al., 2012).**

In this paper the authors studied the performance and effect of diverse keystroke feature combinations on keystroke dynamics authentication system by using fusion approach. First of all, analyzed the influence of four keystroke features and by using fusion approaches to enhance the efficiency of a keystroke dynamic recognition system, later then transformed into similarity scores by using Gaussian Probability Density function (GPD) and Direction Similarity Measure (DSM). Next, three fusion approaches are introduced to merge the scores pairing with different combinations of fusion rules.

The result of this paper shows that the finest performance is obtained by the combination of both dwell time and flight time collectively. Dwell time (D1): which means the time interval between a key pressed until the key is released), flight time (D2): which means the time interval between a key press and the next key press) and Flight Time (D3): which means the time interval between a key release and the next key press. Negative value may occur if the next key is pressed before the previous key release. Finally, this experiment also investigates the effect of using larger dataset on recognition performance, which turns out to be rather consistent.

D1 offers the best performance among all four keystroke features if used independently, while the combination of D1 and D3 produces the optimal result in fusion mode. Hence, it is now clear why these two types of keystroke features are preferred by most of research works in keystroke dynamics domain.

➤ **(Chang et al., 2012).**

This paper discussed the authentication via mobile device; there are many authentication methods for the users. Most users used PIN-based authentication, since they do not employ a standard QWERTY keyboard for conveniently entering text-based passwords. However, PINs provide a small password space size, which is vulnerable to attacks. Many studies have employed the Keystroke Dynamic-based Authentication (KDA) system, which is based on keystroke time features to enhance the security of PIN-based authentication. Unfortunately, unlike the text-based password KDA systems in QWERTY keyboards, different keypad sizes or layouts of mobile devices affect the PIN-based KDA system utility.

This paper proposed a graphical-based password KDA system for touch screen handheld mobile devices. A user enters his or her graphical password through an identical human–computer interface and therefore the user’s keystroke features will not be affected if the user uses different devices which is easy to use in touch screen handheld mobile devices, and applies it in the proposed system.

The experiment results show: (1) EER is 12.2% in the graphical-based password KDA proposed system. Compared with related schemes in mobile devices, this effectively promotes KDA system utility; (2) EER is reduced to 6.9% when the pressure feature is used in the proposed system. The accuracy of authenticating keystroke time and pressure features is not affected by inconsistent keypads since the graphical passwords are entered via an identical size

(50 mm ×60 mm) human–computer interface for satisfying the lowest touch screen size and a GUI of this size is displayed on all mobile devices.

➤ **(Wang et al., 2012).**

This paper explains the user authentication via keystroke dynamics remains a challenging problem due to the fact that keystroke dynamics pattern cannot be maintained stable over time, and it describes a novel keystroke dynamics-based user authentication approach. The proposed approach consists of two stages, a training stage and an authentication stage.

In the training stage, a set of orthogonal bases and a common feature vector are periodically generated from keystroke features of a legitimate user's several recent successful authentications. In the authentication stage, the current keystroke feature vector is projected onto the set of orthogonal bases, and the distortion of the feature vector between its projections is obtained.

User authentication is implemented by comparing the slope correlation degree of the distortion between the common feature vector with a threshold determined periodically using the recent impostor patterns. The model in this paper can be applied to any way where password based access controls take place, for instance, it can be embedded into a Window NT or Window 2000 log-in module. Any users accessing the computer are prompted to type their password. If their typing passwords are correct, their current keystroke patterns are analyzed by the model to provide additional security.

Theoretical and experimental results in this paper show that the proposed method presents high tolerance to instability of user keystroke patterns and yields better performance in terms of FAR and FRR compared with some recent methods.

➤ **(Singh and Thakur, 2012).**

This paper explained multiple numbers of security systems are available to protect user computer or resources. among them, password based systems are the most commonly used system due to its simplicity, applicability and cost effectiveness But these types of systems have higher sensitivity to cyber-attack. Most of the advanced methods for authentication based on password security encrypt the contents of password before storing or transmitting in the physical domain. But all conventional encryption methods are having its own limitations, generally either in terms of complexity or in terms of efficiency.

This paper presented a new method based on user behavior which will attempt to identify authenticity of any user failing to login in first few attempts by analyzing the basic user behaviors/activities and finally training them through neural network is simple in designing which provides high level of security & at the same time is also cost effective because it does not need any extra hardware.

Keyboard Dynamics, being one of the cheapest forms of biometric, has great scope. It is easy to implement on the password based system or systems. This system also discriminate the users on the basis of their typing behavior as a genuine user and non-genuine user. This method have the number of application numerous irrespective of their nature. With this method two ways security is used which provides more security to password based systems and gives new direction of development to password based security system.

➤ **(Calot et al., 2013).**

This paper declared the keystroke dynamics is a biometric technique to identify users based on analyzing habitual rhythm patterns in the way they type. In order to implement this technique different algorithms to differentiate an impostor from an authorized user were suggested. One of the most precise methods is the Mahalanobis distance which requires calculating the covariance

matrix with all that this implies: time processing and track each individual keystroke event. The hypothesis of this paper was to find an algorithm as good as Mahalanobis which does not require track every single keystroke event and improve, where possible, the processing time. To make an experimental comparison between Mahalanobis distance and euclidean normalized, a distance which only requires calculate the variance, an already studied dataset was used.

The experimental work of 20 events varied 0:24%. Normalized euclidean was faster than Mahalanobis distance for 132ms but slower than euclidean for only 60ms. Versatility in normalized euclidean is also an advantage, passwords may be changed and the already-trained keys be kept in the new training. Those results lead to the conclusion that normalized euclidean distance is strong enough to be used and its advantages in data sizes and versatility are considerably important to be chosen against Mahalanobis distance and its success rate suggests that it should be employed against euclidean distance.

CHAPTER THREE

The Proposed Keystroke Dynamics Model

3.1 Introduction

This chapter presents the proposed model of anomaly detection of keystroke dynamics authentication, and the experimental keystroke dynamic system which realizes the proposed model. The aim of the proposed model is to increase the authentication power, expressed in terms of lower rejection of genuine users and higher rejection of imposters.

The performance of the proposed model is based on achieving lower error rate, the Equal-Error-Rate, which is the comparative metric of the effectiveness of an anomaly detector models. The reference dataset for the model evaluation is the CMU dataset (Killourhy, 2012).

The proposed model is an enhancement of anomaly detection model in (Al-Jarrah, 2012) which is based on the distance to median as a measure of typing anomaly.

3.2 The CMU dataset

The CMU dataset is the largest publically available keystroke dataset which has been carefully collected to reflect the typing rhythm of various people, with different typing experiences and habits, collected at different times of the day and the week. The dataset contains the typing records of 51 subjects, eight sessions per subject and 50 typing records per session. The chosen text for typing rhythm evaluation in this experiment is a 10-character hard password, which contained shift characters, numbers and special characters. The Enter (Return) key was measured too, so the actual password is of 11 characters.

3.3 The CMU feature set

The feature set in the CMU dataset consists of:

- Hold: Time duration between key press and key release of a key.

It is equal to $\text{keyup} - \text{keydown}$ of any pressed key.

The keyup and keydown are timestamp values in millisecond of the events of pressing and releasing a key.

- UD: Latency time, the time duration between the release of key_i and the pressing of key_{i+1} .
- DD: Time duration between pressing of key_i , and pressing key_{i+1} .

It is equal to $\text{UD}_{i, i+1} + \text{Hold}_i$.

Each typing record consists of 31 values, these are:

- 11 Holds, for the 10 characters and the Enter key.
- 10 UDs, for time duration between 11 successive pairs of typed keys
- 10 DDs, for time duration between 11 successive pairs of typed keys.

3.4 Using the CMU Dataset for Model Evaluation

The CMU dataset was used in (Killourhy, 2012) in the evaluation of 14 anomaly detection algorithms, using the EER metric comparison. A subset of the main dataset was selected as an imposter dataset, for each subject, it contains 5 records from the 50 other subjects, a total of 250 imposter records. Each subject's session data of 400 records are used for training, as a genuine user dataset, against the corresponding imposter dataset.

For each algorithm, the value of EER for each subject is calculated, and then the average and standard deviation of these values.

3.5 The Median-Std model

The proposed an anomaly detection model in (Al-Jarrah, 2012) which relied on the median of set timing values of a key, instead of the average, as the central point of timing values of that key. The median is not influenced by outlier values which can distort the average, taking in consideration that during typing extreme values can happen due to distraction of attention or other reasons.

The model proposes a distance threshold from the median within which a timing value is considered genuine, otherwise it is considered anomalous.

The distance threshold factor for this model is the standard deviation, for the timing values of a feature set.

According to this model, a typing record, during the testing phase, is considered genuine if a certain pass-mark of the number of genuine feature set values of that record is classified as genuine.

The result of measuring the EER values of the 51 subjects using this model, resulted in 0.080, an improvement over the previous results in (Killourhy, 2012), but it is still below the desired ultimate error rates as required by security standards as in (Killourhy and Maxion, 2009).

3.6 The Proposed Median-Median Model

The proposed enhancement of the MED-STD model is in using a different measure of Distance-to-Median (DTM), as a metric of anomaly from the normal typing behavior which is centered around the median as a point-of center. The assumption here is that the standard deviation is derived from the mean, which can be affected by extreme or outlier values. Therefore the proposed model is based on the following criteria:

- A.** The median of timing values of each typed character, obtained during the training session, is considered as the reference center-point to measure acceptance or rejection against.
- B.** The distance-to-median values, measured for each character of the password, during the training session, are used as the acceptance / rejection condition of the typed character during testing.
- C.** The distance-to-median (DTM) is calculated as a function of the median rather than the mean., as below:

$$DTM = C \times M$$

Where M is the median of timing values of the key, and C is a multiplying constant.

- D.** During the training phase, a template vector is created, which is a vector of Median and DTM values for the password.
- E.** During the testing phase, the timing value of a password character is considered acceptable if it lies within the upper and lower limits around the median of that character.

The upper limit = $M + DTM$, and the lower limit = Min .

- F.** During the testing phase, an entered password is considered genuine (acceptable) if the count of the acceptable timing values of the password characters (Score Mark) are within an acceptable Pass Mark. The pass mark is determined by the user depending on his typing rhythm, whether more rejection of imposters or less rejection of genuine users.

3.7 The Keystroke Dynamic System

3-7-1 System overview

The biometric method refers to the identification of humans by their characteristics or traits, it is divided into two classes:-

- Dynamic biometric: - this type is characteristics dynamic of humans such as behavioral which use of an individual's walking style or gait to determine the identity.
- Static biometric: - this type is characteristics static of humans such as iris, finger-print and palm-print.

In this thesis, the dynamic class is using to determine if the user authorized or not, the behavioral biometric of keystroke dynamics uses the manner and rhythm in which an individual types characters on a keyboard or keypad. The keystroke rhythms of a user are measured to develop a unique biometric template of the users typing pattern for future authentication, it is include the overall speed, variations of speed moving between specific keys, common errors and the length of time that keys are depressed.

The proposed frame work is consisting of two procedures

- 1- Training procedure.
- 2- Testing procedure.

3.8 The development environment

In this point, described the tools that used in the experimental: Windows Form Microsoft Visual Studio 2010, C#, .NET Framework and Timing.

3.8.1 Timing

A date and time format string defines the text representation of a Date Time or Date Time Offset value that results from a formatting operation. It can also define the representation of a date and time value that is required in a parsing operation in order to successfully convert the string to a date and time. A custom format string consists of one or more custom date and time format specifies. Any string that is not a standard date and time format string is interpreted as a custom date and time format string.

A standard date and time format string uses a single format specified to define the text representation of a date and time value. Any date and time format string that contains more than one character, including white space, is interpreted as a custom date and time format string.

A standard format string is simply an alias for a custom format string. The advantage of using an alias to refer to a custom format string is that, although the alias remains invariant, the custom format string itself can vary. This is important because the string representations of date and time values typically vary by culture. For the invariant culture, the short date pattern is

"MM/dd/yyyy". For the fr-FR culture, it is "dd/MM/yyyy". For the ja-JP culture, it is "yyyy/MM/dd".

In our work it used date time to return current time for each event (keyUp and KeyDown) then convert all result to milliseconds by using timespan to calculate difference between the current time and midnight, January 1, 1970 UTC.

3.9 Training procedure

The biometric dynamic depends on this procedure, when the new user used the system to the first time, it will register user-name and password. The number of registered password called training. The training stage is considering as the main stage for identification the user and provide more biometric security, the training stage in general are divided into two parts:-

- Dynamic training biometric: - it is defining as determining a specific number from the training stage; this number used for log-in process, and it is increased by one during each log-in process.
- Static training biometric: -it is defining as determining specific number from training stage and it remains fixed number for each log-in process. In this thesis, used this type by determine the fixed number of user training and the number can be equal to 30 times.

The training procedure includes Med-Training -New algorithm for registered the new user and training the password that related with him/her, after that saved this training in the database and used it for comparison with the next procedure as called result procedure.

➤ **Med-Training-New Algorithm**

The Med-Training-New algorithm provides a tool for the collection of training data about a user's typing behavior, which will be used to build a template of user's typing profile. The algorithm generates two measurements that are based on keydown (keypress) and keyup (key release) :-

- $\text{Hold} = \text{Keyup1} - \text{Keydown1}.$
- $\text{Latency} = \text{Keydown2} - \text{Keyup1}$

This algorithm includes the steps shown in Figure 3-1

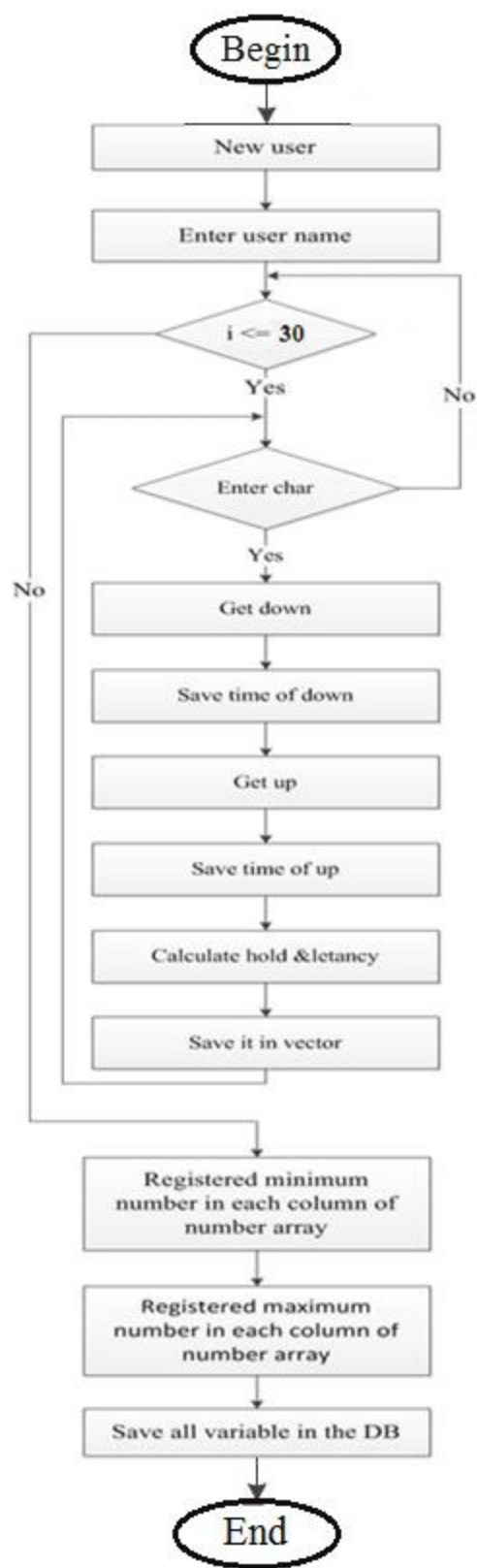


Figure 3-1 Med-Training –New Algorithm

Algorithm: Med- Training -New algorithm

// Input: user-name and password

Output: save profile template vectors in a database //

$N \leftarrow \text{user-name}$

For $i=1$ to 30// 30 is the number of training repetitions

While (enter char==true) Then

$M[i] \leftarrow \text{char}$

$Z1 \leftarrow \text{Get Down (char)}$

$DB \leftarrow \text{save time (Z1)}$

$Z2 \leftarrow \text{Get Up (char)}$

$DB \leftarrow \text{save time (Z2)}$

$H \leftarrow \text{keyup1-keydown1}$

$L \leftarrow \text{keydown2-keyup1}$

$DB \leftarrow \text{save (H, L)}$

END While

END For

$\text{min} \leftarrow \text{minimum number in each column (DB)}$

$\text{max} \leftarrow \text{maximum number in each column (DB)}$

med ← Median (DB)

LL ← min // LL is lower limit or lower threshold of a feature

DTM ← med * 0.7 // DTM is Distance to Median

UL ← med + DTM // UL is upper limit or upper threshold of a feature

Return (all variables)

3.10 Testing Procedure

The testing procedure reads a typed password, with its KSD timing data, and compares the test data (test vector) with the two stored template vectors (LL and UL). For a password of 10 characters, there are 30 feature set elements:

- 10 Hold (H)
- 10 UD (up-down or latency)
- 10 DD (down-down) which represents H + UD.

The test vector contains the 30 feature set elements.

The test vector elements will be compared with the upper and lower limits of the user's template that was stored in the database during training. A feature elements that lies between the upper and lower limits are given the score of 1 (acceptable), otherwise 0 (rejected).

A user's test typing is considered as genuine if the numbers of acceptable feature elements are within a pre-defined Pass-Mark, for example 70% of all features, which is 21 feature elements. The Pass-Mark can be adjusted to change emphasis whether to reduce the number of genuine users rejected as imposters or vice-versa.

Any user that is already registered in the system will enter his user-name and password, this procedures applied in the Med-Test-Registered algorithm.

➤ Med-Test-Registered Algorithm

The Med-Test-Registered algorithm used to determine the user is authorized or not, this algorithm includes sequential stages, as shown in Figure 3-2

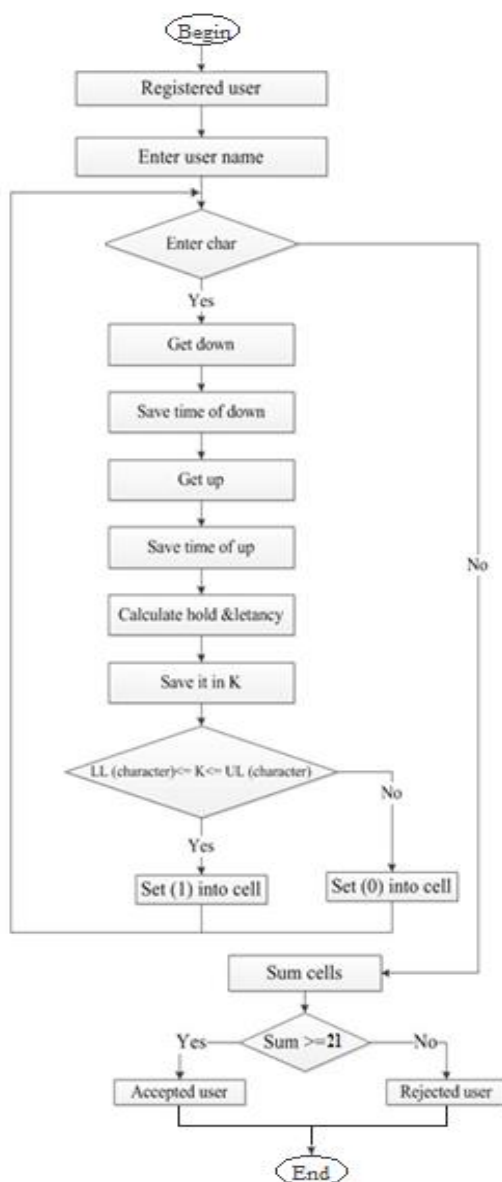


Figure 3-2 Med-Test-Registered Algorithm

Algorithm: Med-Test-Registered algorithm

// Input: user-name and password

Output: identify user //

$N \leftarrow \text{user-name}$

$i \leftarrow 0$

While (enter char==true) Then

$Z1 \leftarrow \text{Get Down (char)}$

$K \leftarrow \text{save time (Z1)}$

$Z2 \leftarrow \text{Get Up (char)}$

$K \leftarrow \text{save time (Z2)}$

$H \leftarrow \text{keyup1-keydown1}$

$L \leftarrow \text{keydown2-keyup1}$

$K \leftarrow \text{save (H, L)}$

If (LL (char) <= K => UL (char)) Then

$X[i] \leftarrow 1$

ELSE If

$X[i] \leftarrow 0$

END If

$i \leftarrow i+1$

END While

$R \leftarrow \text{sum}(X)$

If $(X \geq 21)$ Then

Accepted user

Else if

Rejected user

Return (identify user)

3.11 Data Collection System

The proposed system focused on developing the KSD authentication scheme through empirical collected data by using a prototype system.

3.11.1 System overview

Computer systems and network are being used in almost every aspect of our daily life. As a result, we need to deal with the User authentication which means a method for identifying the user and verifying that the user is allowed or not to access some important resource. In this thesis, we focused on the Keystroke Dynamics is considered one of the most important methods of authentication, the behavioral of this method uses the manner and rhythm in which an individual types characters on a keyboard or keypad.

3.11.2 Data entry model

The keystroke approach needs to collect samples for experiment it, then determines each sample is authorized or not. In this thesis, collected random samples and make each sample to test the model, provided that each sample using the standard keyboard, Figure 3-3 shows the main steps in the proposed model diagram.

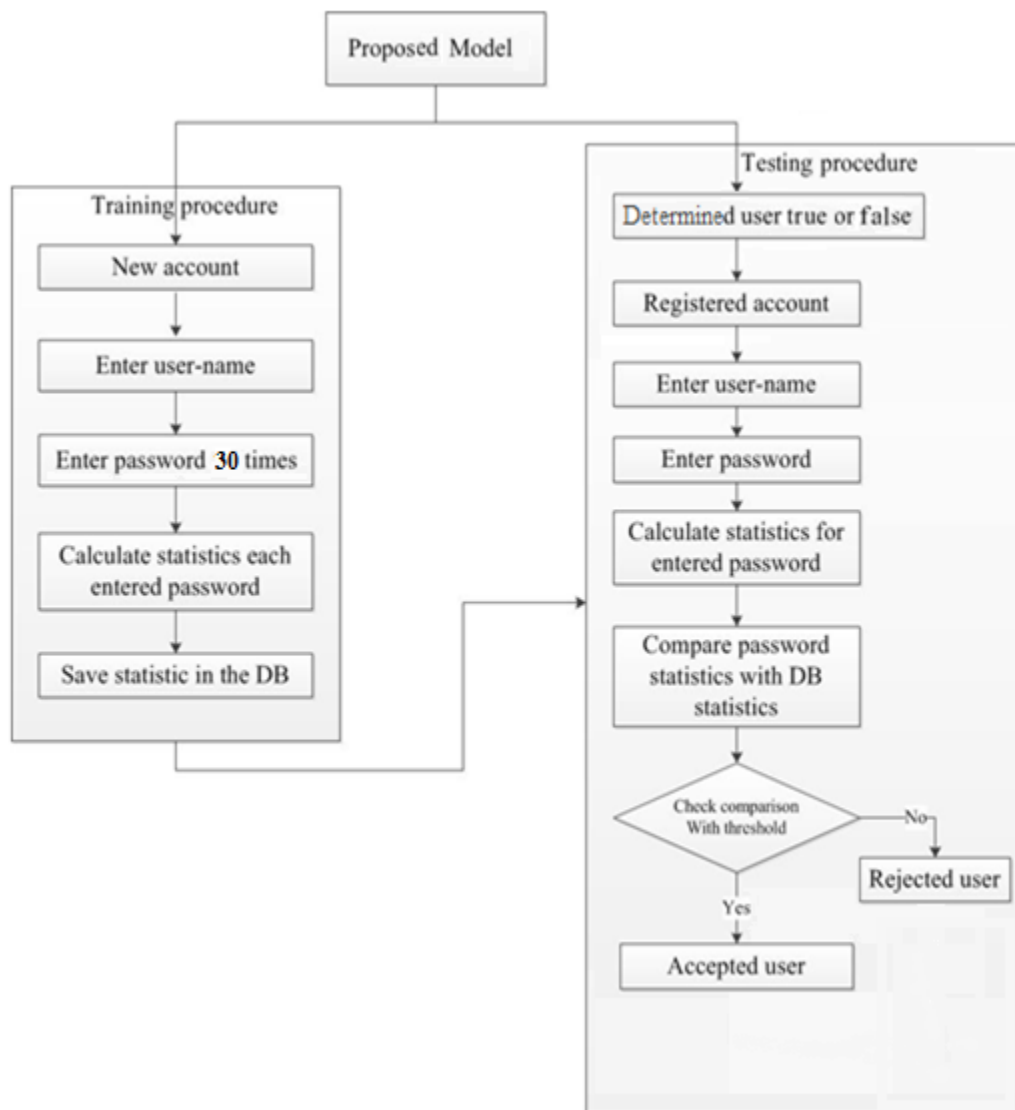


Figure 3-3 Proposed Model Diagram

3.11.3 Sample data

In this thesis, it collects a set of random samples of students for experience of the system and the execution was as follows

3.11.3.1 Interface Execution

The first interface appears in the program implementation is the input interface, as it allows students enter their name with University ID, as shown in Figure 3-4.

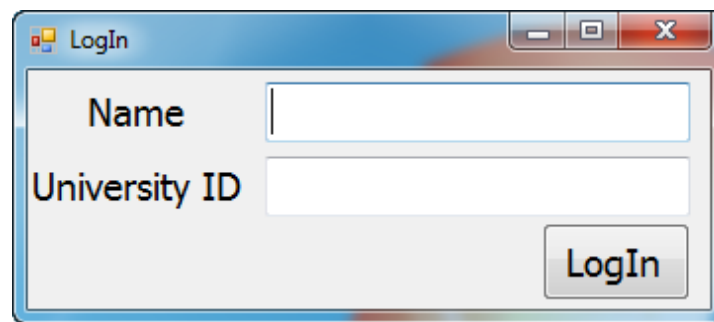


Figure 3-4 first interface in our model

After insert student information, the main interface of the program will appear which consists of two parts: the first part is the method which considered the training stage, and the second part is the test which considered the testing stage, as shown in Figure 3-5.

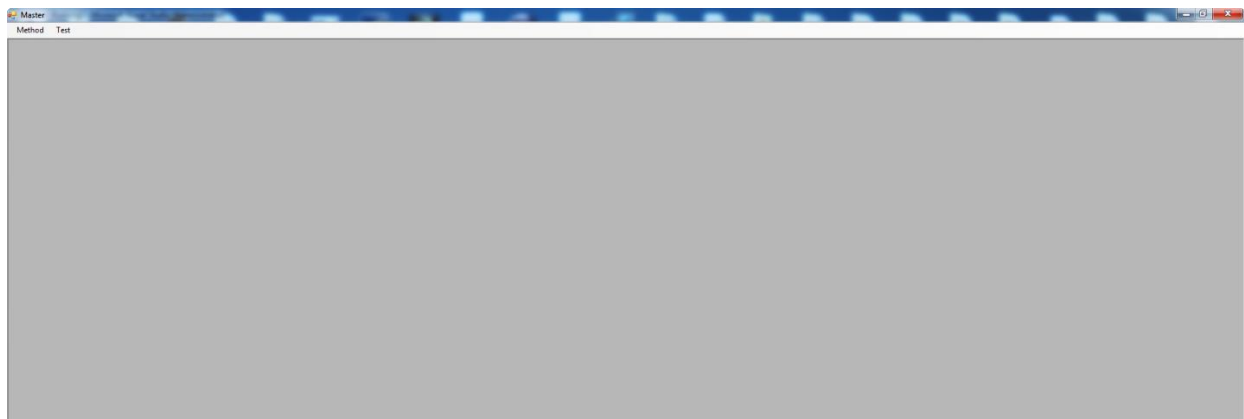


Figure 3-5 main interface

When pressed on the method choice, the interface appears that allows students to enter own password and the application determines a counter to count the number of times and insert password such as (30 times), then save all passwords with Statistics in the database, as shown in Figure 3-6

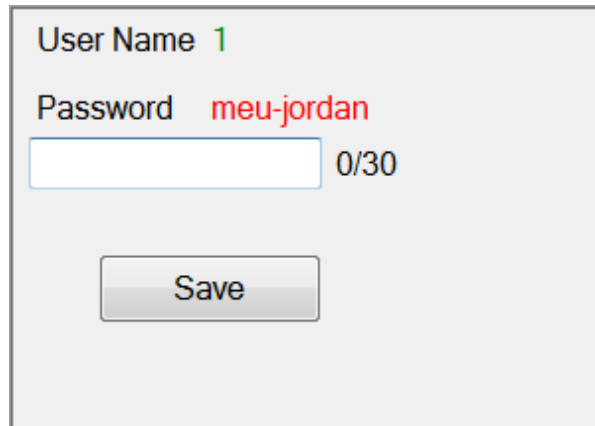


Figure 3-6 training stage interface

When pressed on the method choice, we will enter into the second phase of the system, the student enter own password and the supervisor of the system is identifying the student is True user or False user. The depending on identifying is determined FRR and FAR.

If the user is true user and the application may preventing it from accessing to the system because misidentification user, in other words, a person is true with error style, this is FRR

If the user is false user and the application may allowed it from accessing to the system, in other words, a person is false with similar true style, this is FAR, as shown in Figure 3-7.

User Name 1

Password meu-jordan

☒ True User ☐ False User

Test

Figure 3-7 testing stage interface

The admin interface includes, as shown in Figure 3-8

- Users: - the admin chooses the desired user name for the system application.
- Export: - it appears users report.
- Excel, PDF, and HTML: - the admin chooses the form for statistics.
- Calculate: - calculates the Statistics that is clarified in the above algorithms.
- Admin button: -the admin pressed on this button if he wants any change in the variables of the application.

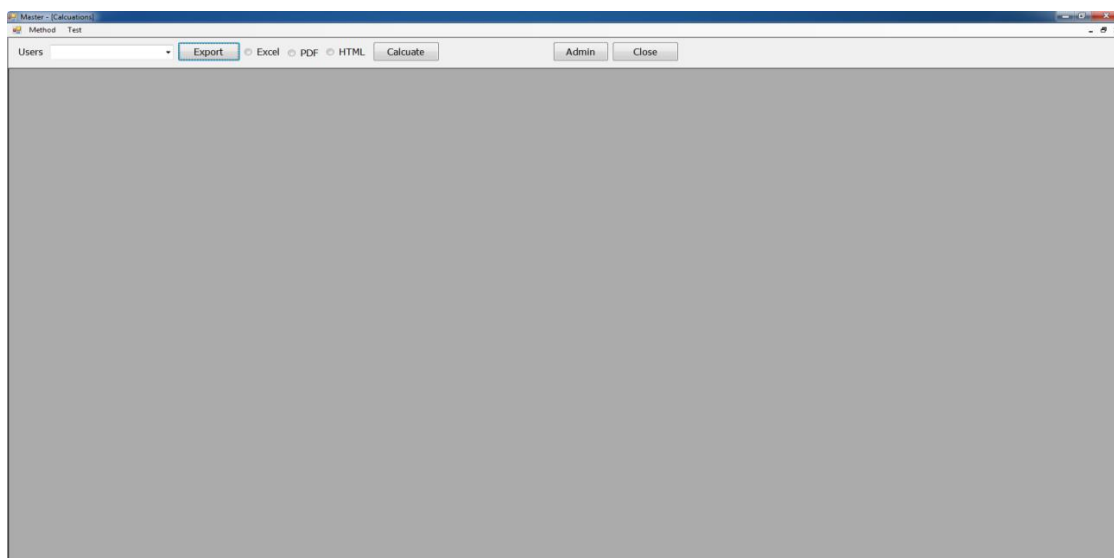


Figure 3-8 admin interface

When pressed on the admin button in previous interface, the admin determined two values that appeared in the interface, as shown in Figure 3-9

- Distance value: it is the value which multiplication in the median.
- Pass mark: it is the threshold for determining the user is acceptable or rejected.

The screenshot shows a window titled 'Master - [AdminPage]' with a menu bar containing 'Method' and 'Test'. The main area contains two input fields: 'DistanceVal' with the value '0.7' and 'Pass Mark' with the value '16'. Below these fields is an 'Update' button.

Figure 3-9 change variables interface

When the user enters his/ her password, the system calculates the Hold, down and up for each character in the password, as shown in Figure 3-10

username	Original_ID	UD_m0	Hold_m0	DD_m0	UD_e1	Hold_e1	DD_e1	UD_u2	Hold_u2	DD_u2	UD_u3	Hold_u3	DD_u3	UD_u4	Hold_u4	DD_u4	UD_u5
1	14	0	340020	340020	0	1280074	1280074	1360077	1440083	2800160	2240128	1680096	3920224	3600206	560032	4160238	4
1	14	13440769	1280073	14720842	960055	1360078	2320133	2480142	1520087	4000229	2880165	1360077	4240242	3840220	800046	4640266	4
1	14	28711642	1120064	29831706	320019	960054	1280073	2800161	1360077	4160238	2480142	1440083	3920225	2880164	320018	3200182	4
1	14	16720956	1520087	18241043	160009	1120064	1280073	1600092	1120064	2720156	3120178	1520087	4640265	4960284	240014	5200298	6
1	14	25681469	320018	26001487	27041547	1360078	28401625	1440082	1440082	2880164	1920110	1600092	3520202	4080233	560032	4640265	4
1	14	8160466	1280074	9440540	240013	1680096	1920109	1840106	1200068	3040174	1920110	1510087	3430197	4090233	550031	4640264	4
1	14	8880508	80005	8960513	10080576	1440083	11520659	1920110	1200068	3120178	2160124	1360077	3520201	3120179	640036	3760215	3
1	14	12720727	480028	13200755	13840791	1450083	15290874	1990114	1360078	3350192	1920110	1440082	3360192	4800275	720041	5520316	5
1	14	17521002	1440082	18961084	2880165	1280073	4160238	2240129	1280073	3520202	2160123	1600092	3760215	2640151	800046	3440197	3
1	14	11040632	1040059	12080691	560032	1360078	1920110	2400138	1280073	3680211	1840105	1440082	3280187	3360193	550032	3910225	4
1	14	9040517	1200069	10240586	560032	1520087	2080119	800046	1280073	2080119	2080119	1440082	3520201	2240128	640036	2880164	3
*																	

Figure 3-10 password calculates interface

The system calculates the statistics of the password which are described in the (Med-Proxy-New) algorithm, as shown in Figure 3-11

username	Original_ID	Calculation_Type	UD_m0	Hold_m0	DD_m0	UD_e1	Hold_e1	DD_e1	UD_u2	Hold_u2	DD_u2	UD_-3	Hold_-3	DD_-3
1	14	Standard deviation	7768873.46772039	487876.831542061	7880855.22416689	8217403.12489856	183624.551236826	8248097.78671148	550008.443074759	114998.653441991	579535.766007914	398499.16849807	97892.6686760168	###
1	14	Median	12720727	1120064	13200755	560032	1360078	2080119	1920110	1280073	3120178	2160123	1440083	#
1	14	Min	0	80005	340020	0	960054	1280073	800046	1120064	2080119	1840105	1360077	#
1	14	Max	28711642	1520087	29831706	27041547	1680096	28401625	2800161	1520087	4160238	3120178	1680096	#
1	14	Distance	8904508.9	784044.8	9240528.5	392022.4	952054.6	1456083.3	1344077	896051.1	2184124.6	1512086.1	1008058.1	#
1	14	Lower Limit	0	80005	340020	0	960054	1280073	800046	1120064	2080119	1840105	1360077	#
1	14	UDper Limit	21625235.9	1520087	22441283.5	952054.4	1680096	3536202.3	2800161	1520087	4160238	3120178	1680096	#
*														

Figure 3-11 statistics calculates interface

The system then compares the results with the threshold to determine the value in each cell is it one or zero, which are described in the (Med-Proxy-Registered) algorithm, as shown in

Figure 3-12

username	Original_ID	UD_m0	Hold_m0	DD_m0	UD_e1	Hold_e1	DD_e1	UD_u2	Hold_u2	DD_u2	UD_-3	Hold_-3	DD_-3	UD_j4	Hold_j4	DD_j4	UD_c
1	17	-1	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0
1	17	-1	-1	-1	0	-1	0	-1	-1	-1	0	-1	0	0	0	0	0
*																	

Figure 3-12 statistics calculates interface

CHAPTER FOUR

The Discussion of Results

4.1 Introduction

The empirical analysis carried out in this thesis focuses on two aspects:

- Examining the results of applying the proposed statistical med-med model as an anomaly detector in keystroke dynamics, using the CMU benchmark dataset.
- Explore the results of using the implemented KSD system to collect training and testing data, to test the authentication capability in distinguishing between genuine users and imposters.

4.2 CMU benchmark dataset analysis using the MED-STD model

Table 4-1 shows a detailed measurement of the error metrics FAR, FRR, EER for each of the 51 subjects in the CMU dataset. The purpose of showing this table is to confirm the EER value as reported in (Al-Jarrah, 2012). In addition, the shown details can be used in the comparison with results of the proposed med-med for the same subjects.

Table 4-1 Error Analysis of CMU Dataset Med-Std Model (31 Features Set)

Subject	FAR	FRR	EER	Pass-Mark
2	0.204	0.185	0.195	19
3	0.104	0.055	0.080	18
4	0.092	0.115	0.104	21
5	0.036	0.025	0.031	21
7	0.056	0.075	0.066	19
8	0.044	0.060	0.052	21
10	0.016	0.020	0.018	22
11	0.020	0.050	0.035	20
12	0.020	0.010	0.015	21
13	0.028	0.025	0.027	19
15	0.084	0.060	0.072	22
16	0.260	0.185	0.223	18
17	0.016	0.020	0.018	18
18	0.056	0.065	0.061	23
19	0.028	0.030	0.029	21
20	0.132	0.100	0.116	22
21	0.088	0.080	0.084	22
22	0.024	0.035	0.030	19
24	0.040	0.055	0.048	23
25	0.072	0.055	0.064	21
26	0.040	0.050	0.045	21
27	0.072	0.040	0.056	21
28	0.056	0.065	0.061	18
29	0.048	0.040	0.044	20
30	0.104	0.090	0.097	23
31	0.264	0.195	0.230	23
32	0.196	0.165	0.181	20
33	0.220	0.175	0.198	22
34	0.156	0.130	0.143	21
35	0.164	0.120	0.142	22
36	0.008	0.015	0.012	19
37	0.108	0.105	0.107	20
38	0.072	0.055	0.064	25
39	0.072	0.040	0.056	21
40	0.136	0.185	0.161	23
41	0.048	0.030	0.039	22
42	0.020	0.005	0.013	22
43	0.040	0.025	0.033	20

Table 4-1 ... Continued

44	0.124	0.125	0.125	23
46	0.096	0.155	0.126	25
47	0.212	0.280	0.246	23
48	0.048	0.050	0.049	22
49	0.072	0.055	0.064	26
50	0.072	0.065	0.069	22
51	0.096	0.100	0.098	20
52	0.008	0.010	0.009	20
53	0.012	0.020	0.016	20
54	0.092	0.050	0.071	22
55	0.012	0.150	0.081	17
56	0.052	0.045	0.049	21
57	0.076	0.045	0.061	21
AVG	0.083	0.078	0.080	21.08
STD	0.066	0.061	0.061	1.87

4-3 CMU benchmark dataset analysis using the MED-MED model

Table 4-2 shows the error metrics of FAR, FRR and ERR, as well as the pass-mark for the each of the 51 subjects in the CMU dataset.

In comparison with the MED-STD analysis for the same dataset, the following observations can be made:

1. The EER metric has an average of 0.07, compared to 0.08 in the MED-STD.
2. The EER metric has a lower standard deviation compared to its average, which indicates less dispersion.

3. The Pass-Mark has an average of 25, compared to 21 in the MED-STD, which can be interpreted that the Med-Med model is using more features in order to classify a subject as imposter or genuine.

Table 4-2

Error Analysis of CMU Dataset Med-Med Model (31 Features Set)

Subject	FAR	FRR	EER	Pass-Mark
2	0.112	0.105	0.109	26
3	0.060	0.055	0.058	24
4	0.048	0.024	0.036	26
5	0.076	0.055	0.066	26
7	0.072	0.075	0.074	25
8	0.048	0.050	0.049	25
10	0.012	0.010	0.011	25
11	0.032	0.075	0.054	24
12	0.032	0.030	0.031	24
13	0.016	0.015	0.016	24
15	0.044	0.045	0.045	25
16	0.236	0.234	0.235	24
17	0.032	0.032	0.032	25
18	0.096	0.065	0.081	25
19	0.020	0.025	0.023	26
20	0.084	0.105	0.095	22
21	0.076	0.090	0.083	25
22	0.016	0.030	0.023	24
24	0.056	0.020	0.038	25
25	0.056	0.080	0.068	26
26	0.044	0.045	0.045	25
27	0.052	0.065	0.059	25
28	0.068	0.070	0.069	25
29	0.086	0.070	0.078	25
30	0.088	0.075	0.082	27
31	0.148	0.230	0.189	27
32	0.152	0.125	0.139	23
33	0.076	0.125	0.101	25
34	0.100	0.120	0.110	24
35	0.140	0.175	0.158	25

Table 4-2 ... Continued

36	0.016	0.015	0.016	24
37	0.104	0.070	0.087	25
38	0.040	0.025	0.033	26
39	0.064	0.060	0.062	25
40	0.152	0.170	0.161	25
41	0.052	0.070	0.061	25
42	0.028	0.015	0.022	26
43	0.008	0.100	0.054	25
44	0.052	0.035	0.044	26
46	0.072	0.080	0.076	27
47	0.244	0.180	0.212	26
48	0.052	0.045	0.049	26
49	0.064	0.075	0.070	26
50	0.068	0.105	0.087	26
51	0.048	0.070	0.059	27
52	0.064	0.040	0.052	24
53	0.016	0.025	0.021	24
54	0.028	0.065	0.047	26
55	0.012	0.010	0.011	22
56	0.048	0.065	0.057	25
57	0.076	0.090	0.083	24
AVG	0.068	0.073	0.071	25.04
STD	0.050	0.052	0.049	1.11

4.4 MEU Data Collection and Analysis

Table 4-3 shows the error metrics of the 6 subjects experimental dataset obtained by using the implemented KSD system, which is based on the Med-Med model.

For this experiment, the “Enter” key measurements were not included the feature set, as many login procedures do not include an Enter key, instead they use a separate submit button to enter username and passwords, so 28 feature set elements were used for the chosen password “meu-jordan”. The obtained results show lower EER error rate in comparison to that obtained using the CMU dataset for the same model, which can be a result of two factors:

- a. Limited number of subjects for comparison between genuine and imposter data.
- b. Lower number of repetitions during training and testing, 30 compared to 200 in case of CMU.

Table 4-3 Error Analysis of MEU dataset Med-Med Model

(30 Features Set)

Subject	FAR	FRR	EER	Pass mark
1	0.025	0.000	0.013	24
2	0.000	0.000	0.000	18
3	0.100	0.067	0.084	22
4	0.100	0.075	0.088	24
5	0.000	0.000	0.000	22
6	0.133	0.100	0.117	23
AVG	0.060	0.040	0.050	22.17
STD	0.058	0.046	0.052	2.23

CHAPTER FIVE

Conclusions and Future Work

5.1 Overview

Keystroke Dynamics as an identity authentication solution is quickly emerging as a viable, low cost, non-intrusive alternative to traditional biometric technologies. As with all technology, KSD is not without its challenges. Higher error rates is of particular concern, especially when there is insufficient training to acquire the typing profile, or when the use of different types of keyboards can result in a change in typing behavior.

In this thesis, we have conducted an empirical study focused on improving the keystroke dynamics hit rate and reducing the error rate, on two experimental data:

- a) A public benchmark dataset, developed and verified by CMU (Killourhy, 2012).
- b) A small dataset collected at MEU, as part of this work.

5.2 Conclusions

The reported results have shown an improved performance in the anomaly detection of the proposed med-med model, compared to previous work using the same CMU dataset. The error rate (EER) is 0.070, a reduction of 27% compared to the top performing model in the CMU study, and a reduction of 12.5% compared to the med-std model (Al-Jarrah, 2012).

At the error rate of 0.07 (7%), the Hit Rate is 93%, which indicates that even though the proposed model has a higher anomaly detection performance, it does not deliver the required detection power expected in access control standards (CENELEC. European Standard, 2002). However, the proposed model can serve as a secondary authentication factor in a multi-factor authentication tool.

The obtained results from the MEU experiment showed lower EER error rate and higher hit rate, compared to the results using the CMU dataset for the same Med-Med model, for the obvious reason that the MEU dataset is much smaller than the CMU dataset, and the MEU experiment used 30 repetitions for training, compared to 200 in case of CMU. The choice of 30 repetitions for training is based on practical reasons related to user intolerance to extended repetitions, in a real KSD authentication tool.

Therefore to confirm the proposed anomaly detection model as a secondary factor in a practical authentication tool, further experimental work is needed where the KSD system is used dynamically as an authentication tool rather than just a data collection system.

5.3 Recommendations for Future Work

Based on the reported work, the following suggestions can be put forward for future research work.

- a) Further experimental data is needed using the proposed model, in different experimental environments, to enhance the MEU dataset and make it another reference dataset.
- b) Implementing the proposed model within an experimental multi-factor authentication tool.
- c) Investigating the authentication performance of the proposed model on a touch tablet or an iPad.
- d) Developing the proposed model into an App authentication tool for smart phones security.

Reference

Al-Jarrah, M. M. (2012). An Anomaly Detector for Keystroke Dynamics Based on Medians Vector Proximity. *Journal of Emerging Trends in Computing and Information Sciences*. Volume. 1, Issue. 3.

Araujo, L. C., Sucupira Jr, L. H., Lizarraga, M. G., Ling, L. L., and Yabu-uti, J. B. (2004). User authentication through typing biometrics features. In *Biometric Authentication*, pp. 694-700. Springer Berlin Heidelberg.

Balagani, K. S., Phoha, V. V., Ray, A., Phoha, S. (2011). On the discriminability of keystroke feature vectors used in fixed text keystroke authentication. *Pattern Recognition Letters*. pp. 1070-1080.

Bleha, S., Slivinsky, C., and Hussien, B. (1990). Computer-access security systems using keystroke dynamics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Volume. 12, Issue. 12, pp. 1217-1222.

Calot, E., Rodríguez, J. M. (2013). Improving versatility in keystroke dynamic systems. In *XVIII Congreso Argentino de Ciencias de la Computación*. Volume.1, Issue.5.

CENELEC. European Standard. (2002). Alarm systems, access control systems for use in security applications.

http://www.cenelec.eu/dyn/www/f?p=104:110:284844008749401:::FSP_ORG_ID,FSP_LANG_ID,FSP_PROJECT:73,25,13155

Chang, T. Y., Tsai, C. J., Lin, J. H. (2012). A graphical-based password keystroke dynamic authentication system for touch screen handheld mobile devices. *Journal of Systems and Software*. pp. 1157-1165.

Chang, T. Y., Tsai, C. J., Yang, Y. J., Cheng, P. C. (2011). User Authentication Using Rhythm Click Characteristics for Non-Keyboard Devices. International Conference on Asia Agriculture and Animal (IPCBEE). pp. 167-171.

Cho, S., Han, C., Han, D. H., and Kim, H. I. (2000). Web-based keystroke dynamics identity verification using neural network. Journal of organizational computing and electronic commerce, Volume 10, Issue. 4, pp. 295-307.

Crawford, H. (2010). Keystroke dynamics: Characteristics and opportunities. In Privacy Security and Trust (PST). IEEE Eighth Annual International Conference. pp. 205-212.

Giot, R., El-Abed, M., Hemery, B., Rosenberger, C. (2011). Unconstrained keystroke dynamics authentication with shared secret. Computers & Security. pp 427-445.

Giot, R., El-Abed, M., Rosenberger, C. (2009). Greyc keystroke: a benchmark for keystroke dynamics biometric systems. In Biometrics: Theory, Applications, and Systems. IEEE 3rd International Conference. pp. 1-6.

Gunathilake, N. M., Padikaraarachchi, A. P. B., Koralagoda, S. P., Jayasundara, M. G., Paliyawadana, P. A. I. M., Manawadu, C. D., Rajapaksha, U. U. S. (2013). Enhancing the Security of Online Banking Systems via Keystroke Dynamics. IEEE Computer Science and Education International Conference. pp. 561-566.

Haider, S., Abbas, A., and Zaidi, A. K. (2000). A multi-technique approach for user identification through keystroke dynamics. In Systems, Man, and Cybernetics, 2000 IEEE International Conference, Volume. 2, pp. 1336-1341.

Harun, N., Woo, W. L., Dlay, S. S. (2010). Performance of keystroke biometrics authentication system using artificial neural network (ANN) and distance classifier method. In Computer and Communication Engineering (ICCCE), IEEE International Conference pp. 1-6.

Hu, J., Gingrich, D., Sentosa, A. (2008). A k-nearest neighbor approach for user authentication through biometric keystroke dynamics. In Communications IEEE International Conference. pp. 1556-1560.

Joyce, R., and Gupta, G. (1990). Identity authentication based on keystroke latencies. Communications of the ACM, Volume 33, Issue. 2, pp. 168-176.

Kang, P., Hwang, S. S., and Cho, S. (2007). Continual retraining of keystroke dynamics based authenticator. In Advances in Biometrics, pp. 1203-1211. Springer Berlin Heidelberg.

Karatzouni, S., Clarke, N. (2007). Keystroke analysis for thumb-based keyboards on mobile devices. In New approaches for security, privacy and trust in complex environments. pp. 253-263. Springer US.

Karnan, M., Akila, M. (2010). Personal authentication based on keystroke dynamics using soft computing techniques. In Communication Software and Networks. ICCSN'10. IEEE Second International Conference. pp. 334-338.

Karnan, M., Akila, M., Krishnaraj, N. (2011). Biometric personal authentication using keystroke dynamics: A review. Applied Soft Computing. pp. 1565-1573.

Killourhy, K. S. (2012). A Scientific Understanding of Keystroke Dynamics (No. CMU-CS-12-100).

Killourhy, K. S., and Maxion, R. A. (2009). Comparing anomaly-detection algorithms for keystroke dynamics. In Dependable Systems & Networks, DSN'09. IEEE/IFIP International Conference . pp. 125-134.

Liu J., Yu R. F., Lung C. H., Tang H. (2009). Optimal combined intrusion detection and biometric based continuous authentication in high security mobile ad hoc networks. IEEE Journal of Wireless Communications. pp. 806-815.

Miller, B. (1994). Vital signs of identity. IEEE Spectrum. pp. 22–30.

Monrose, F., Rubin, A. (1997). Authentication via keystroke dynamics. In Proceedings of the 4th ACM conference on Computer and communications security. pp. 48-56.

Monrose, F., Rubin, A. D. (2000). Keystroke dynamics as a biometric for authentication. Future Generation Computer Systems. pp. 351-359.

Obaidat, M. S., Sadoun, B. (1997). Verification of computer users using keystroke dynamics. IEEE Transactions on Systems, Man and Cybernetics. pp. 261–269.

Panasiuk, P, Saeed, K. 2010. A modified algorithm for user identification by his typing on the keyboard. Springer Advances in Intelligent and Soft Computing. pp.113-120.

Pedernera, G. Z., Sznur, S., Ovando, G. S., Garcia, S., Meschino, G. (2010). Revisiting clustering methods to their application on keystroke dynamics for intruder classification. In Biometric Measurements and Systems for Security and Medical Applications (BIOMS). IEEE Workshop pp. 36-40.

Revett, K. (2011). User authentication via keystroke dynamics: An artificial immune system based approach. ICIT the 5th International Conference on Information Technology. Volume. 7, Issue.1.

Rybnik, M., Panasiuk, P., Saeed, K. (2009). User authentication with keystroke dynamics using fixed text. In Biometrics and Kansei Engineering IEEE International Conference. pp. 70-75.

Sahoo, P., Deb, P. (2010). Secure Strokes—A Security Mechanism for Authentication in Mobile Devices using User's Behavioral Pattern of Keystrokes Dynamics with Visual Cues. pp.68-80.

Serwadda, A., Wang, Z., Koch, P., Govindarajan, S., Pokala, R., Goodkind, A., and Balagani, K. (2013). Scan-Based Evaluation of Continuous Keystroke Authentication Systems. IT Professional. pp. 20-23.

Singh, P. I., Thakur, G. S. M. (2012). Enhanced password based security system based on user behavior using neural networks. International Journal of Information Engineering and Electronic Business (IJIEEB). Volume.4, Issue.1.

Swets, J. A., Pickett, R. M. 1982. Evaluation of diagnostic systems: Methods from signal detection theory. Academic Press, New York.

Teh, P. S., Teoh, A. B. J., Tee, C., and Ong, T. S. (2010). Keystroke dynamics in password authentication enhancement. Expert Systems with Applications, Volume. 37, Issue.12, pp.8618-8627.

Teh, P. S., Yue, S., Teoh, A. B. (2012). Feature Fusion Approach on Keystroke Dynamics Efficiency Enhancement. *International Journal of Cyber-Security and Digital Forensics (IJCSDF)*. pp. 20-31.

Wang, X., Guo, F., Ma, J. F. (2012). User authentication via keystroke dynamics based on difference subspace and slope correlation degree. *Digital Signal Processing*. pp. 707-712.

Yu, E., and Cho, S. (2003). GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification. In *Neural Networks, 2003. Proceedings of the International Joint Conference on*, Vol. 3, pp. 2253-2257.

Zha, H., He, X., Ding, C., Gu, M., Simon, H. D. (2001). Spectral relaxation for k-means clustering. In *Advances in neural information processing systems*. pp. 1057-1064.

Zhao, Y. (2006). Learning user keystroke patterns for authentication. In *Proceeding of World Academy of Science, Engineering and Technology*. pp. 65-70.

Zhong, Y., Deng, Y., Jain, A. K. (2012). Keystroke dynamics for user authentication. In *Computer Vision and Pattern Recognition Workshops. IEEE Computer Society Conference*. pp. 117-123.