HOMOGENEOUS COGNITIVE BASED BIOMETRICS FOR STATIC AUTHENTICATION

by

Omar Hamdy Mohamed

B.Sc. of Computer Engineering, Kuwait University, Kuwait 1995
M.Sc. of Computer Engineering, University of Victoria, Canada 2000

A Dissertation Submitted in Partial Fullfillment of the
Requirements for the Degree of

Doctor of Philosophy

in the Department of Electrical and Computer Engineering

HOMOGENEOUS COGNITIVE BASED BIOMETRICS FOR STATIC AUTHENTICATION

by

Omar Hamdy Mohamed

B.Sc. of Computer Engineering, Kuwait University, Kuwait 1995
M.Sc. of Computer Engineering, University of Victoria, Canada 2000

Supervisory Committee

Dr. Issa Traoré, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Alexandra Branzan Albu, Department Member
(Department of Electrical and Computer Engineering)

Dr. Mihai Sima, Department Member
(Department of Electrical and Computer Engineering)

Dr. Amy Gooch, Outside Member
(Department of Computer Science)

Supervisory Committee

Dr. Issa Traoré, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Alexandra Branzan Albu, Department Member
(Department of Electrical and Computer Engineering)

Dr. Mihai Sima, Department Member
(Department of Electrical and Computer Engineering)

Dr. Amy Gooch, Outside Member
(Department of Computer Science)

Abstract

In today's globally expanding business world, protecting the identity and transactions of online consumers is crucial for any company to reach out for new markets. This directs digital information technologies towards the adoption of stronger and more secure authentication schemes. Increasingly, such biometric-based user authentication systems have proven superiority over the traditional ones (such as username/password).

Unfortunately, despite the significant advances accomplished in developing biometric technologies, there are several barriers to their wide-scale deployment and application for Internet security. Additionally, introducing new biometrics faces similar barriers and challenges such as expensive equipment, or low-precision sensor technologies.

In this research, we propose a novel biometric system for static user authentication, that homogeneously combines mouse dynamics, visual search capability and short-term memory effect. The proposed system introduces the visual search capability, and short-term memory effect to the biometric-based security world for the first time. The use of mouse for its dynamics, and as an input sensor for the other two biometrics, means no additional hardware is required. Experimental evaluation demonstrated the system's effectiveness using variable or one-time passwords. All of these attributes qualify the proposed system to be effectively deployed as a static Web-authentication mechanism.

Extensive experimentation was done using 2740 sessions collected from 274 users. Two classification mechanisms were used to measure the performance. Using the first of these, a specially devised neural network model called *Divide* & *Select*, an *EER* of 5.7% was achieved. A computational statistics model showed a higher classification performance; a statistical classifier design called *Weighted-Sum* produced an *EER* of 2.1%.

The performance enhancement produced as a result of changing the analysis model suggests that with further analysis, performance could be enhanced to an industry standard level. Additionally, we presented a *Proof of Concept (POC)* system to show the system packaging practicality.

Table of Contents

Supervisory Committee ii

Abstract iii

Table of Contents v

List of Tables x

List of Figures xi

List of Abbreviations xii

Acknowledgment xiv

Dedication xvi

1 Introduction 1

1.1 Context 1

1.2 Problem Statement and Research Objectives 2

1.3 Research Contributions 5

1.4 Thesis Outline 6

2 Biometrics in User Authentication and Digital Forensics 8

2.1 Authentication Systems 8

2.2 Biometric Authentication 10

2.3 Physiological Biometrics 11

2.3.1 Retina 11

2.3.2 Iris 12

2.3.3 Face . . . 12
2.3.4 Fingerprint . . . 13
2.4 Behavioral Biometrics . . . 13
2.4.1 Voice . . . 14
2.4.2 Hand Signature . . . 14
2.4.3 Gait . . . 15
2.4.4 Keystroke Dynamics . . . 15
2.4.5 Mouse Dynamics . . . 16
2.5 Multi-Modal Biometric . . . 16
2.6 Properties of Biometric Systems . . . 17
2.6.1 Physiological vs. Behavioral . . . 17
2.6.2 Resistance to Forgery . . . 18
2.6.3 Suitability for Static User Authentication . . . 18
2.6.4 Homogeneous Multi-Factor Models . . . 19
2.7 Biometric Identification in Digital Forensic . . . 19
2.8 Summary . . . 21

3 Related Work and Background 22
3.1 Keystroke Dynamics . . . 22
3.2 Mouse Dynamics . . . 25
3.3 Human Factors . . . 29
3.3.1 Visual Scan and Detection . . . 30
3.3.2 Short-Term Memory . . . 33
3.4 Summary . . . 35

4 Physio-Behavioral Visual-Based Biometric 36
4.1 A New Dimension in the Biometric Field . . . 37

4.2 Biometric System Components 38
4.2.1 Visual Scan and Detection Capability 38
4.2.2 Short-Term Memory Capability 39
4.2.3 Mouse Usage Dynamics 40
4.3 System Framework 40
4.3.1 User Physio-Behavior Sensor Module 40
4.3.2 Feature Extractor Module 41
4.3.3 User Identity Module 43
4.4 Test Design 43
4.5 Data Acquisition 48
4.5.1 Preliminaries 48
4.5.2 Collected Data 49
4.5.3 Feature Analysis 50
4.5.4 Data Filtering Algorithms 54
4.6 Feature Extraction 57
4.6.1 Large Number of Possibilities 57
4.6.2 Feature Selection Criteria 60
4.6.3 Final Feature List 61
4.7 Summary 61

5 Biometric Analysis 64

5.1 Biometric Analysis Using Neural Network 64
5.1.1 Neural Network Design 65
5.1.2 Enrollment Strategy 69
5.1.3 User Authentication 75
5.1.4 User Identification 75
5.2 Biometric Analysis Using Computational Statistics 77

5.2.1 Clustering Analysis 79
5.2.2 Reference Signature 80
5.2.3 Enrollment Strategy 86
5.2.4 User Authentication 86
5.2.5 User Identification 88
5.3 Summary 88

6 Experimental Evaluation **89**
6.1 Procedure 89
6.2 Apparatus 91
6.3 Subjects and Test Sessions 91
6.4 Evaluation Based on Neural Network Model 92
6.4.1 Enrollment Details 92
6.4.2 Performance Calculation 93
6.5 Evaluation Based on Computational Statistics Model 95
6.5.1 Enrollment Details 95
6.5.2 Performance Calculation 95
6.6 Summary 96

7 Experimental Analysis and Results **97**
7.1 User Verification Results Using Neural Network Model 97
7.1.1 User Authentication 97
7.1.2 User Identification 100
7.2 User Verification Results Using Computational Statistics Model . . . 106
7.2.1 User Authentication 106
7.2.2 User Identification 112
7.3 Additional Testing 113

7.3.1 Biometric Factors Separate Evaluation 113
7.3.2 Individual Biometric Feature Impact 115
7.3.3 Random vs. Comprehensible Phrases 116
7.4 Summary 117

8 Conclusion **118**
8.1 Work Summary 118
8.2 Application 119
8.3 Future Work 121

Bibliography **123**

List of Tables

4.1 Raw Data Types 51
4.2 Sample Raw Data After Initial Processing 52
4.3 Biometric Feature Set 62
4.4 Sample Row Data After Initial Processing 63

5.1 Sample *FTE* Results 72
5.2 Individual Biometric *EER* and *Weight (w)* 85

7.1 Performance of the Three Neural Network Models 98
7.2 Performance Results Obtained for the Three Neural Network Models 99
7.3 *FTE* Impact on Evaluation Performance 99
7.4 Identification Performance for the Three Neural Network Models . . . 104
7.5 Performance Results for the Statistical Model Using Step Rule 107
7.6 Performance Results for the Statistical Model Using Triangle Rule . . 108
7.7 EER Highlighted for the Computational Statistics Models 108
7.8 Identification Performance for Computational Statistics Models . . . 112
7.9 Performance Results for Individual Biometric Factors 114
7.10 Individual Biometric Feature Performance 115
7.11 Study of Comprehendible Text Performance Impact 117

List of Figures

3.1 Human Eye Anatomy 31
3.2 Conceptual View of the Human Brain 34

4.1 Proposed System Architecture 45
4.2 Sample Shuffled Keyboard 46
4.3 Text Length Threshold 46
4.4 Embedded Human Factor Tests 47
4.5 Raw Data Phase 1 Snapshot 53
4.6 Outlier Effect on Data Calculations 55
4.7 Memory Impact on Fly Time 58
4.8 Inconsistent Recordings of Memory Impact on Fly Time 59

5.1 General Neural Network Design 68
5.2 Neural Network Training Phases 71
5.3 Divide and Fuse Training Technique 73
5.4 Divide and Select Training Technique 74
5.5 Verification Process 76
5.6 Identification Process 78
5.7 Statistical Decision Rules 82
5.8 Score Matching Fusion Method 84
5.9 Computation Statistics Procedure 87

7.1 ROC Curves for neural network Verification Results 101
7.2 User Identification Scenarios by Example 103
7.3 ROC Curve for Statistics Model Using *Simple Sum* Technique 110
7.4 ROC Curve for Statistics Model Using the *Weighted-Sum* Technique . 111

List of Abbreviations

CC	Correct Classification
CCR	Correct Classification Rate
CV	Confidence Value
CT	Click Time
DST OCC	Mouse Traveled Distance to Key Occurrence Ratio
DST Q2S	Mouse Traveled Distance in QWERTY to Shuffled Ratio
EER	Equal Error Rate
ELRA	European Languages Resource Association
FAR	False Acceptance Rate
FM	False Match
FMR	False Match Rate
FNM	False Non-Match
FNMR	False Non-Match Rate
FRR	False Rejection Rate
FRVT	Face Recognition Vendor Test
FT	Fly Time
FT OCC	Fly Time to Key Occurrence Ratio
FT Q2S	Fly Time in QWERTY to Shuffled Ratio
FTE	Failure to Enroll
FTER	Failure to Enroll Rate
FVC	Fingerprint Verification Competition
JVM	JAVA Virtual Machine
LDA	Linear Discriminant Analysis
MCR	Misclassification Ratio
MVE	Minimum Volume Ellipsoid

NIST	National Institute of Standards and Technology
OTP	One-Time Password
PCA	Principal Components Analysis
PKI	Public Key Infrastructure
QWERTY	Named after the first six letters on a keyboard
ROC	Receiver Operating Characteristic
SS	Simple-Sum
VR	Variance Reduction
W-FOV	Wide Field of View
W-n-C	Wait and Click Time
W-n-F	Wait and Fly Time
WS	Weighted-Sum
X2Y	Mouse Traveled X-axis to Y-axis Ratio

Acknowledgment

All praise be to Allah the High, who alone has guided me to this success. I say what King Solomon said: "O my Lord! so order me that I may be grateful for Thy favours, which Thou hast bestowed on me and on my parents, and that I may work the righteousness that will please Thee: And admit me, by Thy Grace, to the ranks of Thy righteous Servants," Quran [27:19].

Next, I wish to extend my profound gratitude to my supervisor, Dr. Issa Traoré, whose invaluable ideas, stimulating discussions, and constant encouragement have guided this research to its present state. I thank him for believing in me and giving me the opportunity to do my PhD under his supervision. Through his solid research methodology, I learned the key pillars needed to be a successful researcher.

I would like to express my deep appreciation to my supervisory committee members, Dr. Alexandra Branzan Albu, Dr. Mihai Sima, and Dr. Amy Gooch for all their valuable advice and critical feedback. I also must mention that it was a privilege and an honour to have Dr. Mihaela Ulieru of the University of New Brunswick serving as my external examiner. Despite her extremely busy schedule, she spared no effort or time in evaluating the dissertation.

My sincere heartfelt gratitude is extended to my mother, who always provided me with spiritual strength through her boundless sacrifice and generosity. She leaves me speechless, because no words can ever return her favours. Her kindness is a candle that guides my life journey, and forever I will be indebted to her.

I would like to take this opportunity to thank my M.Sc. supervisor, Dr. Fayez Gebali, from whom I learned the research fundamentals. His support extended beyond academic boundaries. He will always be a good example in my life.

I wish to thank Glenlyon Norfolk Senior School, and in particular Mr. Rick Calderwood and Mr. Robert Britten, for allowing me to conduct my research's

primary experiment there. Every effort was made by the school to allow over two hundred students to take part in the studies.

My special thanks extend to my colleagues and sincere friends Ahmed Awad, Youssef Nakkabi, and Mohamed Watheq for their enormous help and advice.

Omar H. Mohamed, Victoria, BC, Canada

Dedication

To:

My mother, to you it all goes

My father's soul, oath fulfilled

My wife, I thank you

Chapter 1

Introduction

In recent decades the world has converged in the digital era. The concepts of globalization and virtual village have introduced a new culture that has pushed the world economy from being product based to being information/service based. Digital information duty has shifted toward the center of all world activities and services. As a result, digital information has become one of the most important assets to governments, to corporate entities, to financial entities, and to individuals. Protecting these valuable intangibles against unauthorized interference and use has becomes a strategic imperative at all levels [1].

1.1 Context

Securing digital information, or what is now known as eSecurity, is no longer just about risk mitigation and compliance by ensuring that intruders are "kept out." In the business world, for instance, eSecurity helps in cost reduction and revenue generation, which can only happen by letting the desirable customers in. Securing digital information gives confidence for business designers to expand. Expansion includes the functionality domain by enabling new key applications; it also involves

the delivery domain by providing new accessibility channels such as the Internet. On the other hand, compromise of digital information security could be devastating. For example, one history's biggest frauds was carried out by an ordinary bank employee who managed to overstep his authority in the bank. In January 2008, the French bank Societe Generale announced the discovery of a $7.14 billion fraud achieved by a trader employee who used his knowledge of the group's security systems to conceal his fraudulent positions.

In fact, preventing a digital crime is more critical than discovering it after it has already happened. In many cases, the effect of breaching a digital system and accessing classified information is not reversible. Therefore, security regulators and researchers in the field of information security have paid special attention to identity and access management as one of the eSecurity pillars to secure our digital world.

There is no one known approach or technology that is considered the perfect solution to that challenge. In fact, regardless of how advanced the technology is, it is only half of the solution. Effective digital security strategies involve the combination of various complementary protection mechanisms. Regardless of the security policy of the organization, effective authentication mechanisms represent an essential ingredient of the mix of deployed security technologies.

Strong authentication mechanisms are vital for any organization that conducts business online, due to the growing identify threats faced by online environments.

1.2 Problem Statement and Research Objectives

The explosion of online scams such as phishing is hurting e-business. To expand and reach out to new markets, companies are facing increasing pressures to protect the identities and transactions of online consumers by adopting stronger and more secure authentication schemes. Increasingly, biometric technology is being considered as

one of the best and strongest alternatives to traditional authentication mechanisms. Unfortunately, despite the significant advances accomplished in developing biometric technologies, there are several barriers to their wide-scale deployment and application for Internet security. The operation of most biometric systems necessitates special hardware devices (e.g., scanner, camera, or microphone) for data collection. Such devices are not always available on machines deployed on the Internet. In many cases, these devices are deployed only in geographical areas or private networks areas. Hence, the safe-zone diameter in which organizations can securely conduct business is bound by these geographical and network areas [2].

Behavioral biometrics such as mouse dynamics and keystroke dynamics are exceptions since they can be collected using standard input devices (i.e., mouse, keyboard) which are readily available in any traditional computing environment.

Successful analysis of mouse biometrics have essentially targeted dynamic or continuous authentication of users [2], [3]. Continuous authentication consists of repeating the authentication process several times throughout the login session. This necessarily requires a relatively longer time, and hence cannot be used as a viable alternative to traditional password schemes used in static authentication [2]. In static authentication, the user identity is verified only once, and typically at the beginning of the session.

The main objective of the thesis is to address the above limitations of mouse dynamic biometrics, while maintaining its input simplicity. We can define the goal to be: *To develop a new biometric system for static user authentication, by using variable or one-time passwords, and without the need for any additional hardware.* In this thesis, we propose a novel hybrid biometric system that homogeneously extracts and combines physiological and behavioral features. The proposed system introduces two human cognitive factors as biometric factors in the digital security world: *Visual Scan and Detection* and *Short-Term Memory* are two human cognitive factors that have

always been studied in the context of medicine or psychology. In this research, we managed to model and quantify these two features to become discriminative features successfully used to authenticate and identify users. Additionally, the proposed system uses mouse dynamics as a third biometric factor and also as an input mean to capture the data of the cognitive features.

A preliminary study was undertaken to survey different cognitive factors in an attempt to find ones suitable for biometrics. Most of the studied factors either showed weak biometric quality or were difficult to model and read. Through a special test design and with the use of no additional hardware except the traditional mouse, we successfully modeled and quantified the visual scan and detection and short-term memory factors. When a shuffled keyboard and a random text string were displayed, users showed different behaviors while trying to click-in the text string using the shuffled keyboard. Preliminary results from a small sample showed good results that encouraged the generalization of the experiment to a larger population consisting of 274 users.

The collected data was analyzed with different machine learning models. Specifically, various neural network architecture and computational statistics models were tested. We obtain an *EER* of 5.7% and *EER* of 2.1% with *Divide* & *Select* neural networks model and *Weighted-Sum* computational statistics model respectively.

The relatively short data-collection session time qualifies this system to be used in static user authentication. It adds a new dimension to the existing mouse dynamic usage, additional to capturing user motion behaviors. By designing new mouse dynamic features, physiological attributes such as visual scan and detection and short-term memory can now be modeled and be given quantifiable representation. The simplicity in the data input method (the standard mouse, and a light-weight virtual keyboard GUI) makes this system possible for static authentication on the Internet.

1.3 Research Contributions

Contributions of this research can be described in the following points:

New Physiological Biometric The first contribution of this research is the introduction of new physiological attributes, namely visual scan and detection and short-term memory, in the field of biometric security. Measuring these attributes was done by quantifying their effect on mouse dynamics. That homogeneous modeling allowed maintaining the simplicity and effectiveness of the system. This work was published in [4].

Static Authentication Most of the existing works on mouse dynamics have focused primarily on passive or dynamic user authentication. The second major contribution of this research is to help mitigate the gap in that area, and propose an effective mouse dynamics model for static authentication. This model capitalizes on the existing standard computing technology, and does not require any additional components. Using visual scan and detection, short-term memory and mouse dynamics homogenously in user authentication was published in [5].

Large-Scale Experiment All related experiments in the field of mouse dynamic biometrics have a participant average of around 15 users. Since this research is introducing the above contributions for the first time in the area of biometric security, we successfully enrolled a large-scale population of 274 participants to support the conclusions of the research. This significantly larger user population also contributes to validating previous results on mouse dynamics.

In addition to the conference papers published, this work is submitted to *ACM Transactions on Computer and Human Interaction (TOCHI)* [6]. Unlike the conference papers, this journal paper is based on the computational statistics model approach, which achieved significantly higher performance.

1.4 Thesis Outline

The remaining chapters of this thesis are structured as follows:

Chapter 2 This chapter presents a review of the literature underlying this research. It gives a survey on the biometric systems, both physiological and behavioral, that are used in the digital security authentication field. For each type, it presents challenges and limitations, allowing us to formulate the motivation for the proposed research.

Chapter 3 This chapter provides a critical review of major research on keystroke and mouse dynamics, the two biometrics that have close analogy to the research presented in this work.

Chapter 4 This chapter details the design and development of a novel hybrid biometric system that homogeneously combines physiological and behavioral characteristics. Specifically this chapter introduces the proposed biometric system. It explains how features from both physiological and behavioral biometrics are combined to provide a new homogeneous biometric system. It also explains the set goals to be achieved by this research.

Chapter 5 This chapter details the proposed biometric analysis framework; it also explains the test design and gives data acquisition details. Details on feature-extraction process and biometric analysis are also provided.

Chapter 6 This chapter provides the details of the experimental evaluation. These include the experiment procedure, and test component details such as test sessions and enrollment procedures. The chapter also provides explanation of the performance calculation used in this research.

Chapter 7 This chapter presents and discusses the performance results obtained from all the extensive testing using both the neural network and computational statistics models. Performance results which are expressed in terms of *FAR*, *FRR*, and *EER* are tabulated and illustrated graphically for better understanding. Additionally, performance of some special tests is also presented.

Chapter 8 In this chapter, all the completed work and accomplishments are summarized. It also outlines a number of ideas for related future work.

Chapter 2

Biometrics in User Authentication and Digital Forensics

In this chapter we provide a brief overview of biometric technologies. We present the different classes of biometrics available and discuss underlying properties. We also discuss two major areas of application of these technologies, namely, user authentication and digital forensics.

2.1 Authentication Systems

Different authentication schemes that are known today rely on one or more of the following four categories:

1. Something the subject knows.
2. Something the subject has.
3. Something the subject can do.
4. Something the subject is.

The first category depends solely on short information that is private to the subject. Username/Password is the most widespread implementation of this type of authentication systems. *Positive authentication* can also be classified as a member of this group. In positive authentication, a subject is asked certain short questions, such as mother maiden name, amount of tax filed, or current address, then the answers are compared with answers from a trusted source. This trusted source could be a local DB, a third party that has information about the user, or information previously provided by the user. Examples of the second category would be user's physical ID, digital certificates such as (PKI) for Internet authentication, digital signatures, ATM cards, or smart cards or tokens. The third category involves more complex methods that require users to perform an action as part of the authentication process. For example, sending the user a cellular text message (SMS) with special codes, and asking her to verify it with an agent over the phone. The last category contains what are known as the biometric features, such as fingerprints, or DNA.

Adopting one or more of these types of authentication method depends on the nature of the corporation or organization that needs it; access to a nuclear facility database is expected to be much more difficult than access to a Web-account that offers audio downloads. Also, authentication choices for phone banking are fewer than those available at the branch itself. When designing an authentication mechanism, the security team in a corporation or organization considers all these factors, as well as the corporation policies and business objectives.

Authentication systems that combine methods from different categories are called *multi-factor* authentication systems. *Multi-factor* authentication systems are known to be more difficult to compromise than *single-factor* ones. For example, accessing an ATM machine requires two authentication types that belong to different categories, the ATM card (something you have), and a PIN (something you know). However, the reliability of any authentication system depends on its proper implementation.

A poorly implemented two-factor authentication system may be less secure than properly implemented single-modal one [7].

The above four categories of authentication scheme can be divided into two types: traditional and biometric schemes. The first three are usually referred to as the traditional types of user authentication, while the last is known as biometric user authentication.

2.2 Biometric Authentication

According to ISO/IEC, biometrics are *"Automated recognition of individuals based on their behavioral and biological characteristics."* The biological aspect of a biometric measures certain physical personal characteristics, such as fingerprint or facial characteristics; the behavioral aspect on the other hand, measures data of personal habits or responses that have developed over time. As such, biometric technologies are categorized into two types, physiological and behavioral. The first biological biometric recorded in history goes back to ancient Babylonians, where hand imprints were used as authenticity proof of certain engravings and works [8].

Traditional user authentication schemes are usually based on secretness (for example, keeping PIN, DoB, or passcode secret) and security (for example, keeping ATM cards or secure tokens safe). System based on these schemes are relatively vulnerable, because the information or authentication mechanism could be guessed, shared, or stolen. In 2006, a hacker was able to spread a malware called VisualBreeze, through which he was able to collect more than 32,000 login codes including bank and mail logins [9]. Introducing biometric features into the user authentication field caused a paradigm shift in established authentication concepts. Because biometrics are not secret-based, they cannot be guessed, shared, or stolen.

Traditional user authentication requires an initial applicant registration. According

to NIST SP-800-63 [10], registration is when "*An applicant applies to a Registration Authority (RA) to become a subscriber of a Credential Service Provider (CSP) and, as a subscriber, is issued or registers a secret, called a token, and a credential that binds the token to a name and possibly other attributes that the RA has verified. The token and credential may be used in subsequent authentication events.*"

Biometric user authentication similarly requires an initial applicant enrollment, in which the applicant provides the Credential Service Provider (CSP) with their biometric data.

Verification of a user is however, conceptually different between the two types. In traditional authentication, to be granted an access a user needs to prove possession of the credentials previously given to her. In biometrics however, the user needs to prove the ability to provide biometric data matching those at enrollment. In other words, the biometric authentication does not rely on Proof of Possession (PoP) [11].

2.3 Physiological Biometrics

As defined in Section 2.2, physiological biometrics measure values of personal biological attributes. Studies such as [12–14] surveyed the different types of physiological biometrics known today. Examples of physiological biometrics include fingerprint, iris, hand geometry, retinal scan, face, ear, body thermal, DNA, and vein mapping. In the next sections, we highlight the main physiological biometrics that are commonly used in user authentication today.

2.3.1 Retina

Retina biometric refers to the features extracted from the unique vascular pattern of the human eye's retina. Due to the large number of extracted features (266 identified unique spots), retina biometric is widely regarded as the most accurate biometric

known [14].

Retina biometric technology is used for static user authentication for highly secured facilities. It requires sophisticated reading devices due to the difficulty of reading the small area of the retina. Low-intensity infrared light is used to scan and record the retina's vascular pattern for later analysis. This type of user authentication requires the physical presence of the user at the time of enrollment and at the time of authentication.

2.3.2 Iris

The iris is the pigmented ring around the eye's pupil. Iris texture has many features that are unique from one person to another. These include freckles, coronas, stripes, ridges, and furrows [15]. Since, the iris is a visible surface part of the eye, a high-precision camera can read and record the iris features. Like retina biometric, iris biometric is also considered very accurate technology for the static user authentication. In a 2006 ICE competition, the winning technology achieved a 91% correct verification rate at a 0.1% *FAR* [16]. Iris biometric requires the physical presence of the user at the time of enrollment and at the time of authentication.

2.3.3 Face

Face biometric depends on extracting facial characteristics such as nose shape, eye socket position, and distance between different facial elements. During enrollment phase, these features are extracted and used to create user template. During static authentication, different algorithms could be used to compare the claimant facial samples with the stored template [17–19]. Two common algorithms used for face recognition are Principal Components Analysis (PCA) and Linear Discriminant Analysis (LDA) [19]. Ongoing research in the field of facial recognition has introduced

a number of efficient algorithms. In the 2006 Face Recognition Vendor Test (FRVT) competition, facial recognition performance scored 90% correct verification at 0.1% *FAR* [16]. However, face biometric still faces big challenges such as illumination variance and facial expression variance effects [20]. Face biometric requires only a simple camera to capture facial data samples. That makes it possible to employ static user authentication based on face biometric at remote locations such as the Internet.

2.3.4 Fingerprint

Fingerprint impression is composed of dark lines known as ridges, and white spaces in between known as valleys. Ridges and valleys of the fingerprint are rich in their unique minutiae. Together they form loops and arches that give each fingerprint its own uniqueness. Biometric systems based on these features show high accuracy. The 2006 Fingerprint Verification Competition (FVC) reported the average lowest recorded Equal Error Rate (*EER*) to be 1.916%[1] for fingerprint recognition. The digital image of the fingerprint can be read by different technologies, from a cheap optical sensor to more sophisticated ultrasound and thermal digital imaging technologies. Fingerprint biometric is possible for static user authentication at remote locations if the proper hardware for scanning the fingerprint is available, which is not always the case.

2.4 Behavioral Biometrics

This section highlights biometric systems that depend on a measurable behavioral trait developed over time for a person. These measurements are later used for authentication and identification. Unlike the physiological biometrics, behavioral biometrics are relatively recent and are moving gradually toward maturity. The

[1] http://bias.csr.unibo.it/fvc2006/results/Light_resultsAvg.asp

common behavioral biometrics are voice, hand signature, keystroke dynamics, mouse dynamics, and gait.

2.4.1 Voice

Human voice is generated by the orchestration between the various neck and mouth organs. Air is forced by the lungs to flow through the vocal cords in the trachea and past the glottis, lips, and tongue. Although all these are biological features of the human, voice is classified as a behavioral biometric. This is because features extracted from the voice are dominated by the developed manner and style of speaking. For example, certain letters of the voiced sounds are produced by adjusting the tension of the vocal cords to different excited or relaxed oscillations [21]. Voice biometric systems work in two modes: text-dependent and text-independent. In the latter, the user is free to pronounce any utterance for a certain period of time. In the 2006 NIST SRE competition, D.E. Sturim *et al.* [22] introduced a voice biometric algorithm in a certain system architecture that achieved *EER* performance of 4.04%. Voice biometric systems are considered one of the most convenient, because they only require the already widespread microphone as the input sensor. This allows the deployment of voice biometric systems as static user authentication for many applications including ones that require remote access.

2.4.2 Hand Signature

Hand signature biometric uses a digital pen and pen tablet to pen trajectory, as well as dynamics such as pen velocity, pressure, and azimuth. Features are then extracted from the collected data using two modes, function-oriented (such as velocity and pressure) and parametric-oriented (such as curvature radius and total signing time) [23]. Local features are the features extracted for each sampling local region

or segment, while global features are the features extracted from the whole signature space [24].

In [25], a Hidden Markov Chain based model showed an *EER* of 9.253%. Hand signature biometric is deployed for static user authentication. However, due to the sophistication of the hardware required, it is not widespread for personal remote authentications.

2.4.3 Gait

Gait could be defined as the harmonic cyclic movements of a person. For example, walking and running contain cyclic harmonic movement and hence are called gait. Recently, researchers have become interested in using someone's gait features as an authentication or identification tool.

According to [26], gait biometric could be defined as the ability to observe some of the notable human harmonic cyclic motions. Different methods have been proposed to efficiently recognize and isolate gait features. In [27] for example, human cyclic activities are segmented based on a change in the human cyclic activity, in order to accurately detect temporal boundaries of cyclic activities.

Gait biometric requires video-capturing mechanism of the human cyclic activities in order to build a template and later for authentication and identification purposes. This type of biometric is more suitable for security monitoring and dynamic user authentication.

2.4.4 Keystroke Dynamics

Research in the area of keystroke dynamic recognition started in the early 1980s. Since then a lot of progress in that area has been accomplished to effectively use keystroke

dynamics for user authentication. Research in the area of keystroke dynamics is discussed in detail in the next chapter.

2.4.5 Mouse Dynamics

Research into mouse biometrics is relatively new compared to keystroke biometrics, despite the fact that they share similar characteristics. Detailed review on previous work related to mouse dynamics is presented in detail in the next chapter.

2.5 Multi-Modal Biometric

Performance of any biometric system could degrade due to several factors [23]. For example, the universality attribute of any biometric could be debatable [13]. In retina biometrics for instance, a person with some sort of retinal atrophy cannot be authenticated using this method. Changes to physical and behavioral features due to age, habits, or skills can cause feature variance of the same user to increase. This automatically causes the biometric-based system performance to decrease. Additionally, some biometrics can have little variation among all the population, which makes it a challenge to differentiate between users.

To address this issue, recent studies have proposed two groups of solutions. Solutions in the first group focus on the methodology in which features of biometric samples are processed. Solutions in the second group propose methods of combining more than one biometric type. The overall result is what is known as Variance Reduction (VR) of the biometric data.

The first group of solutions is divided further into two subgroups, *VR* through classifiers, and *VR* through extractors [28]. *VR* through classifiers uses multiple classification techniques (i.e., group of different classifiers); each classifier from the classifier group would give a confidence value, independently from other classifiers in

that group. Later, confidence values from all classifiers are fused for a final result. *VR* through extractors, on the other hand, has separate feature extractors that target feature subsets from the biometric data. As with *VR* through classifiers, output of all the extractor units is fed to classifiers, and output confidence values are finally fused. Combining more than one biometric type is known as multi-modal biometric. An example would be combining facial, hand-scan, and fingerprint [29]. However, it is important to make a distinction between two types of multi-modal biometric systems. The first, and most, common type is the non-homogeneous multi-modal biometric systems. By non-homogeneous we mean that one can still differentiate between the sub-biometric systems used. The other type is the homogeneous multi-modal biometric systems, in which one cannot differentiate between the subcomponents of the system. An example would be using a text-dependent speech expert and a text-dependent lip expert as in [30]. In this work, we propose a homogeneous multi-modal biometric system which combines three separate modalities.

2.6 Properties of Biometric Systems

In section 2.4, we presented most of the common physiological and behavioral biometrics. In this section, we build on this outline by listing some observations about biometrics and their suitability for user authentication. These observations are the basis for our research motivation.

2.6.1 Physiological vs. Behavioral

Building on the description of each of the biometrics, we can list the following characteristics:

Accuracy Physiological biometrics are generally more accurate than the behavioral ones. That is because physical features have much less variance than behavioral

features. Mental, physical, and emotional status can all contribute to the variance of behavioral biometric data. Nevertheless, precision of data sensors and readers greatly impacts the overall biometric performance.

Cost In general, behavioral biometric systems require significantly less expensive hardware than the physiological ones. This gives an advantage to the behavioral biometrics for widespread deployment over the physiological biometrics.

Scalability Many of the presented behavioral biometric systems require no additional hardware. Keyboard and mouse are standard in each computer system, and microphones are almost as common. Physiological biometric systems do require special hardware. A simple fingerprint scanner requires additional software and hardware to install and use.

2.6.2 Resistance to Forgery

All biometrics that are used in authentication systems are vulnerable to different type of attacks [31]. However, the difficulty of cheating the system at the data-input stage differs from one biometric system to the other. For example, rubber fingers, audio recording, and photos are possible ways of fooling the reading devices of fingerprint, voice, and face biometric systems respectively. On the other hand, fooling a mouse dynamics biometric system requires hacking the terminal system to plant a spyware that can record a genuine mouse movement session, then use it for unauthorized access, which is more difficult to achieve.

2.6.3 Suitability for Static User Authentication

A good static authentication mechanism will be robust and convenient. By convenience we mean that a mechanism is easy to use, and the access decision is made quickly. Many behavioral biometrics, due to high data variability, require long

sessions to build a biometric sample that could be used successfully for authentication. Therefore, keystroke and mouse dynamic biometric systems are more suitable for seamless dynamic authentication and monitoring. Gait is another example, as it is not convenient for users to make few moves in front of a camera before they are authenticated. In this regard, physiological-based biometric systems seem to be achieving this goal. Retina, iris, and fingerprint scans, for example, require a relatively short time to read and analyze data, and are convenient to the user.

2.6.4 Homogeneous Multi-Factor Models

Like in the traditional authentication systems, using more than one biometric system in the user authentication process might seem appealing in addressing the shortcomings inherent in each system when used individually. For example, to increase the performance of a face biometric system, it is combined with a fingerprint system. In most cases, however, this approach is not as appealing as it seems. This is because most of these merges are not homogeneous; we can still differentiate the two models used in the system. Non-homogeneous systems create other complications in terms of the need for two input systems with different analysis mechanisms. Even in the rare examples of homogeneous merges, such as using a voice biometric with a vocal password, other issues are raised. In this example, the user must loudly pronounce a password that is based on secrecy and not sharing with others.

2.7 Biometric Identification in Digital Forensic

Vast advancement in technology has inevitably led to much more sophisticated crimes using complex techniques and procedures. These include money laundering using complicated electronic fund wiring processes, network virus worms intended to create as much damage as possible worldwide, industry spying, and various forms of

computer hacking.

The development of the Internet and related technologies has had the downside of introducing a lot more sophisticated computer crimes involving more or less sophisticated tools and techniques. These include various forms of computer hacking such as denial-of-service attacks, password cracking, and phishing. Intrusion avoidance techniques such as access control and firewall technologies have shown their limits in keeping hackers out and providing full protection. Intrusion detection techniques were proposed as a second line of defense to complement traditional intrusion avoidance techniques. Unfortunately, despite the huge amount of research accomplished in this field so far, intrusion detection techniques have also shown significant limits in dealing with the multiform and ever-changing threat represented by hacking activities. Not only are existing intrusion detection systems not effective in detecting new forms of attacks, they also provide little explanation on the nature and modus-operandi of detected intrusion incidents.

This is when the digital forensic role starts. Digital forensics provide means to investigate in detail intrusion incidents and the methods and techniques used by intruders. This allows the taking of appropriate measures to prevent future occurrence of such incidents, and furthermore can provide evidence that may be used to prosecute intruders in court. Forensic tools are responsible for collecting digital evidence on a certain incident. This enables full or partial damage recovery, fixes security loopholes that caused the incident, and furnishes evidence of the breach offense.

Digital forensic has three major phases: acquisition, analysis, and presentation [32]. Like in a physical crime scene, digital forensic first starts by taking a complete snapshot of the current system state. Next comes the analysis phase, when all the system state data are deeply analyzed in a way analogous to physical forensic analysis techniques. Finally, evidence discovered is presented in a convincing manner.

For a digital evidence to be accepted in a court of law, it must meet certain

requirements such as authenticity, accuracy, and reliability. Therefore, special attention is needed for forensic tool design, in order to guarantee the quality of the digital evidence produced.

Biometric technologies play an important role in collecting reliable evidence that may be used to convict alleged criminals. For instance, in the physical forensic area, fingerprints represent an essential piece of evidence in criminal investigation. In the digital world, capturing similar data is more challenging. For most traditional biometric technologies (e.g., fingerprint, iris), the data capture requires special hardware devices, and as such cannot be conducted passively. Passive data capture is essential for effective biometric identification in the digital world.

In this regards, behavioral biometrics such as mouse dynamics and keystroke dynamics are the most appropriate for biometric identification in digital forensics, because they allow passive data capture.

2.8 Summary

In this chapter we explored the best known types of biometric authentication systems. Survey work showed variable success in different types of system. Comparing the performance results of the different systems shows that physiological biometric systems tend to have higher performance than the behavioral ones. This is typically due to the relative stability of the physiological biometric features. Additionally, we notice that the behavioral biometric systems used in dynamic user authentication have higher performance than the ones used in static user authentication. One explanatory reason could be that these behavioral biometric systems require a relatively large amount of user data to work efficiently. In the next chapter we discuss in more detail relevant behavioral biometrics and other factors related to this research.

Chapter 3

Related Work and Background

The proposed research capitalizes on previous work in the area of mouse dynamics and keystroke dynamics, and introduces other new features.

Among all the physical and behavioral biometrics explained in Chapter 2, mouse dynamics and keystroke dynamics are notable exceptions; they require only a mouse or a keyboard for biometric data collection, and these are essential components of any standard computing system. This chapter reviews and discusses in detail previous work done in both keystroke and mouse dynamics. It also presents research done in the area of visual scan and detection and short-term memory.

3.1 Keystroke Dynamics

The idea of keystroke dynamics biometrics is to recognize the characteristics of the typing rhythm. These include the fly time between keys, dwell time (the time a keyboard key is pressed down) for each finger, and typing habits. Typing habits are mainly relations between keystrokes. For example, after both the *Shift* key and the desired shifted key are pressed, the sequence of the release of the two keys is considered a typing habit. Another example is the overlap between key strokes.

Primitive research in the area started in the early 1980s [33]. The last two decades have witnessed more mature research in the field, showing variable degrees of success in static and dynamic user authentication. Bergadano *et al.* [34] presented a shift in the traditional feature extraction from keystroke dynamics. First, they defined a sample to be the elapsed time between the first and the third key depression in a trigraph. Then, using the array degree of disorder method, similar trigraph samples were compared by measuring the distance between them. That approach was able to address two of the main issues that faced all the previous research in the area: the intrinsic variability of typing, and typing errors. The method was experimentally verified using 154 volunteers, and results were transferred over a 28.8 Kbaud modem to prove suitability for remote applications. This approach was used for identification and authentication with performance of 0.01% *FAR* and 4% *FRR*. The main criticism of their work is the long enrollment and testing text length of 683 characters. In reality, that long a string is not convenient at login time. Since their work is based on only the factor explained above, reducing the text length is expected to jeopardize the performance. Another criticism to the work is that since it depends on measuring distance of similar trigraph samples, it limits the success of using this method in free-text mode.

In 2005, Gunetti and Picardi[1] [35] addressed the free-text-mode criticism of [34]. Using the same idea, they enhanced the method to use n-graph samples instead of the trigraph samples used in [34]. This allowed using more sophisticated distance measures between samples. With these enhancements introduced, the performance obtained for free-text static authentication was 0.005% *FAR* and 5% *FRR*. Therefore, it is important to note that free-text detection targets primarily continuous authentication. The use of 537 characters for enrollment and verification

[1]Gunetti and Picardi were also in the authors list of [34].

phase, makes real-life implementation difficult.

In their approach based on long-text-input, Villani *et al.* [36] were able to achieve a user identification accuracy range between 93.3% to 99.5%. The method was designed to help in identifying people who send inappropriate text-based emails (for example blackmailing or threat emails). It requires a large number of keystroke input (650 for enrollment and 200 for testing) for each user in every session. Extracted features include press duration, transition times, means and standard deviations of measurements, and user preferences in using certain keys. The total extracted features from any data session is 239 features. The nearest-neighbor classifier was the scheme adopted for user identification. The relatively long enrollment and testing sessions are justifiable, given the scope of applications this study was targeting. However, two main criticisms are pointed out. First, enrollment and testing were done in matching conditions; users had to use the same keyboard for enrollment and testing (the same desktop keyboard or the same laptop keyboard). In practice, wrongdoers would usually use places other than their work environments for their inappropriate actions. This means they would not necessarily use the same hardware types. Second, the research scope is identifying the senders of inappropriate typed material by their typing style; in real life however, malicious users who have this intention can attempt to change their typing habits at enrollment or during their malicious attacks.

At first glance at the work of Jiang *et al.* [37], the extracted features seem no different from most of the other work in measuring dwell and release time of keys, and digraph relations as the basis for feature extraction. However, their approach is different in the use of the Hidden Markov Model and Gaussian Modeling to build user profiles. In the experiment, 58 volunteers enrolled using a 9-character password for 15 attempts. The method showed an *EER* performance of 2.54%.

Another untraditional modeling approach was proposed by Lee *et al.* [38]. Typing characteristics were represented using n-dimensional vectors. Then, genetic algorithm

was used to evolve an ellipsoidal hypothesis space for the n-dimensional vector of each user. The work also presented approaches to eliminate outlier from collected data. The experiment was based on 16 participants who enrolled as legitimate and imposter users using 10 passwords. Using the ellipsoidal hypothesis space classification, an average performance of 4.33% *FAR* and 4.36% *FRR* was achieved.

Ahmed and Traore [39] proposed an approach for continuous authentication, aimed at insider attacks. This system could be effectively used for employee surveillance, as it seamlessly monitors user keystroke signatures. The biometric features are extracted from digraph records. The main challenge that faces similar free-text-based schemes lies within the enrollment. The only way to effectively build an accurate profile is for users to enroll using all possible digraph records, which is not a realistic undertaking. In their approach, Ahmed and Traore pioneered a method to overcome this exhaustive approach, by using a sorting map. For each key, two values are assigned, to and from. These are calculated as the average time of all digraphs to/from that key. Later, keys are sorted twice based on the two assigned values, and used to train a neural network. The experiment included 53 participants, and based on that technique, the approach gave a performance of 0.0152% *FAR* and 4.82% *FRR*.

3.2 Mouse Dynamics

Mouse dynamic biometric systems are relatively new compared to other types of behavioral biometrics. Similar to the keystroke dynamic biometrics, the idea is to capture certain mouse movement characteristics. These features are then used to build the user's mouse-usage profile, which could later be used for authentication and identification purposes.

Few different approaches have been proposed to date. In an exploratory study, Hocquet *et al.* [40] asked 10 users to perform certain mouse movements (controlled

movement). Users were asked to click on a moving square as quickly as they possibly could. The mouse movements were sampled, and sample coordinates were then recorded with their time stamp. At the first stage of feature extraction, features such as speed, acceleration and angular velocity were computed. Next, more features were produced by building primitive computational relations between these feature values. These included, average and standard deviation. The final produced feature matrix was stored for later authentication. Using this technique, Hocquet scored an *EER* of 37.5%. At this high error rate, the proposed approach is not suitable for actual real-life user authentication applications.

Gamboa and Fred [41] introduced a mouse-based authentication system using a Web-based game. The system defines two groups of features, spatial and temporal, all based on the mouse actions recorded between clicks. The spatial group contains features such as the x and y coordinates, distance, time, and movement angle. The temporal domain contains features such as velocity and acceleration. Features are filtered per user to keep only the ones giving the best performance, measured as the lowest *EER*. In an experiment using 25 college students, and session length of 10-15 minutes each, an average *EER* of 14% (ranges from 5% to 49%) was scored. The presented work showed some capability of classifying users based on their mouse movement behaviors. However, the high error rate, and the lengthy game method the user is required to use every time, make it impractical to be used as a convenient user authentication technique.

Pusara and Brodley [3] hierarchically grouped the recorded mouse events in addition to the regular mouse movements. The mouse-event data set generated is composed of *Mouse Wheel*, *Clicks*, and *Non-Client (NC)* area movements (for instance, toolbars). The *Clicks* node is further branched to *Single* and *Double* click categories. Features are then generated as counters attached to each node, calculations of the mean and standard deviation values, and speed between point pairs. 42 total features were

created in this research. Analysis and user authentication was later done using a Decision Tree Classifier. Enrollment of a user required on average two hours of a supervised active session. Users were instructed to use only a Web browser throughout the session, and their mouse actions were recorded using the described grouping. A buffer of a certain window size was used as a trigger (once filled) to passively analyze the accumulated mouse actions. In order to reduce the false alarm ratio, alarms were fired after the number of suspicious mouse behaviors exceeded a certain threshold. Pusara and Brodley in their work were able to achieve a *FAR* of 0.43%, and *FRR* of 1.75%. Unlike the previous two methods, users in this experiment were given no movement instructions (free movement); however, collected data were analyzed using an application-specific template. In other words, this system provides a passive authentication of some user group A on a specific application B. Unless this work is untied, and becomes application-independent, it will be unrealistic to use this method in real-life scenarios.

Ahmed and Traore's work [2] developed in 2003 in the ISOT research lab made a breakthrough in the use of mouse dynamics in passive authentication. Their patent-pending approach is movement-free based, and application-independent. The scheme scored a *FAR* of 2.4649% and *FRR* of 2.4614%. New classification of mouse actions was introduced. Any mouse action is classified as either *Mouse-Move*, *Drag-and-Drop*, *Point-and-Click* or *Silence*. Mouse movement was recorded in terms of the traveled distance, elapsed time, and line direction angle. From the recorded mouse actions and movements, five features were generated, namely *Movement Speed*, *Movement Direction*, *Action Type*, *Traveled Distance*, and *Elapsed Time*. In the experimental evaluation, a group of 22 users were enrolled. Participants used the computer for a normal daily use, while the system was passively reading and recording all their actions. Collected data were later analyzed, and a profile was created for each user. The system is composed of a client, which was installed on remote machines, and a

centralized server connected to the client through the Internet. During evaluation, users again simulated a real-life use of the computer, and the system passively monitored and reported users classified as intruders based on their mouse dynamics. As a result the above-mentioned performance was achieved.

Apart from the main experiment, another separate experiment was conducted to study the effect of the environment variables. In the main experiment, users were working in free-movement environment, where they had the choice of applications they wanted to work on, and tests were conducted on different machines. To verify and isolate the effect of the environment variable, a small experiment was conducted. Seven users selected randomly from the main experiment participants performed the same guided mouse movements using a specially developed GUI. Participants used the same machine for their experiment. As a result, many of the features that were designed for help in free-style mouse movement were factored out. Using the remaining features to dynamically authenticate users gave a performance of 2.245% *FAR* and 0.898 % *FRR*. This indicated that the proposed mouse dynamics can operate effectively in both controlled and free-mouse-movement environment.

In their survey on using mouse dynamics in user authentication, Revett *et al.* [42] presented a preliminary method to authenticate users using graphics. At login time, thumbnail images oriented in a circle were displayed. Using the mouse, the user had to click 5 of the displayed images (password) in the correct sequence (similar to the rotary lock). Biometric features were extracted from the time between successive mouse clicks and total time. Experimental evaluation with 6 users showed an *FAR* and *FRR* of approximately 2-5%. However, the lack of experiment details such as enrollment method, data extraction, testing procedures, in addition to the small number of subjects make us conclude that verifying the validity of this idea requires more fundamental work.

Sayed and Traore [43] proposed a static user authentication system based on mouse

gesture dynamics. In this controlled authentication system, users are requested to repeatedly replicate presented gestures using the mouse. Twelve extracted features were used to enroll and authenticate users. Features included vertical and horizontal velocities, tangential acceleration, curvature rate of change, and slope angles. Using a modular neural network classifier, the system achieved a performance scope of FAR 1.55% and FRR of 2% with a test session length of 40.7 seconds.

Syukri *et al.* [44] combined the principal features in hand signature and mouse dynamics to present a mouse-based signature system for user authentication. An experimental study to examine different features produced five good features: number of signature points, coordinates of points, signature writing time, velocity, and acceleration. The classification model was based on geometric average mean. Using this classifier and data of the five features collected from 21 subjects, the system achieved an FAR of 8% and FRR of 9%.

3.3 Human Factors

We can define a human factor as a cognitive or physical property or behavior of humans. The study of human factors is a multidisciplinary science that involves psychology, medicine, operations research, and industrial engineering. From the early times, these factors have been studied by psychologists or by physiologists. Psychologists studied human factors in an attempt to understand how humans relate to the world around them. Physiology research was toward enhancing human health. In the 18th century, research was done on using what is known about human capabilities and limitations to design products, processes, systems and work environments [45]. The study of human factors aims at enhancing performance, increasing safety and user satisfaction [46]. In the following sections, we discuss in more detail two cognitive factors this research is based on: visual scan and detection

and short-term memory.

3.3.1 Visual Scan and Detection

Visual scan is a frequent process in our daily life. We search for road signs, we visually search for a piece of information of interest on a website, or we search for certain food brand in a grocery store. Visual scan and detection could be defined as the task of discriminating a target of interest from surrounding distracters [47].

The eye's fovea has the highest concentration of photoreceptor cells (cones) (Figure 3.1). Therefore, when we perform a visual search, object images that fall on the fovea are the ones that are examined, as they can be identified in details [48]. Special quick and short eye movements called saccades and fixation help changing the eye location to locate stimuli of potential interest during the visual search [48].

Visual detection is the process of discriminating a target from the surrounding noise [49]. Most of the research in the area has attempted to quantify and model this cognitive factor. In this section we discuss three experiments related to our proposed work.

Neisser [50] conducted an experiment that required subjects to search a letter list of variable length to detect a certain target letter. This is known as *structured visual detection*. Visual stimuli were presented as an organized array of items in the visual field, and subjects were required to search the stimuli according to a number of serial search rules. In his research, Neisser was able to establish a linear relationship between the serial position of the target in the letter list and the time needed to find it [51]. This finding suggested that any sequential (structured) visual search scanning is modeled as a linear self-terminating scanning process.

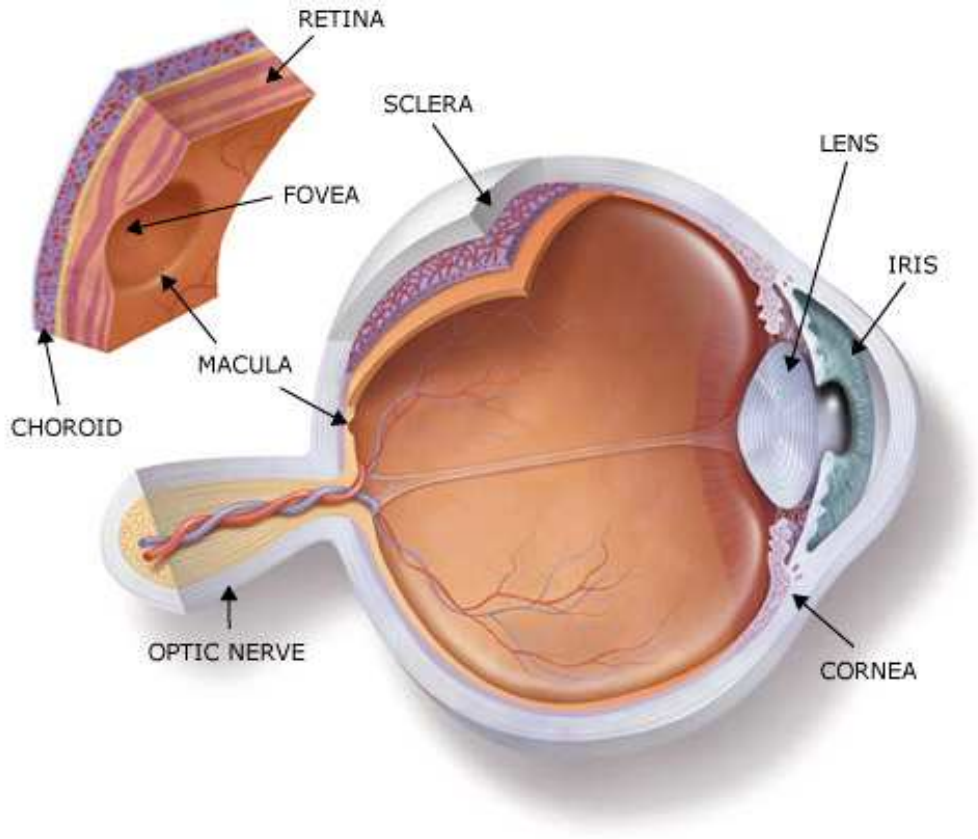


Figure 3.1: Human Eye Anatomy. The fovea has the highest concentration of photoreceptor cells. Source: http://www.worldelderland.com/photos/color27.png

Drury, on the other hand, performed an experiment to study unstructured visual search [52]. Visual stimuli were scattered randomly in the visual search display, and subjects were requested to detect a certain target with no search rules. The research focused on studying the relation between the search time allowed and the detection probability. The research showed that detection probability increased as a function of time at a non-linear rate.

Estes and Taylor [53] developed a procedure known as *detection method* that is also based on the unstructured visual search. At the beginning of the experiment, subjects were assigned two target letters (for example R and V). In each test trial, an array of letters including one of the two target letters was displayed tachistoscopically. Subjects were requested to tell which of the two target letters was shown.

X	D	Y	Z
A	G	K	L
E	R	F	M
W	T	O	S

The experiment measured both accuracy and latency of response. In this experiment, each letter in the stimulus had to be analyzed well enough to distinguish it from the two targets. Results showed that the array size affected the performance. For example, for arrays of 8, 12, or 16 letters, the percentage of correct detections were 0.78, 0.72, and 0.67 respectively.

There are similarities in the fundamentals between the work presented above and our research. The work above used time as a factor to measure the visual search criterion in humans. Additionally, the above work studied the effect of the visual noise increase on the overall search performance. Our research, as will be detailed in the later chapters, uses similar fundamentals in more details, but from discriminative perspective.

3.3.2 Short-Term Memory

Short-term memory is the human sensory input storage. In the visual system, short-term memory receives visual snapshots received by the visual cortex. Input from the visual cortex is briefly stored there to give a chance to the perceptual part of the brain to go through all the temporarily stored visual snapshots, filter out all that are not important or not worth attention, and pass to the working memory visual information that is of high importance. Figure 3.2 provides a conceptual illustration of this process. Short-term memory is characterized by two properties: size and decay. For visual information, the short-term memory is estimated to have a size of about 17 letters, and an average decay time of 200 ms [54]. Different studies have researched the retrieval capacity and capability of short-term memory [55]. It was found that information retrieval capability from short-term and working memory is different from one person to another.

In this section we present sample research on the short-term memory factor that is relevant to our research scope. Many research works study the relation between the time articles are temporarily stored in short-term memory and the rate of presentation of these articles. Posner [56] suggested that recall rate from the short-term memory is improved as a result of increasing the presentation rate. In one experiment, Posner [57] requested 20 subjects divided into two groups to listen to a series of digits at a fast rate or at a slow rate, based on the group they were in. After that, subjects were requested to repeat back the digits they heard, and use the word blank for the digits they no longer remembered. The percentage of the correct series was finally calculated. Another supplementary experiment asked subjects to recall digits in certain order. Results showed that when recall was not requested in any order, there was no significant difference between the fast and slow groups. However, a substantial difference was concluded when an ordered recall was requested.

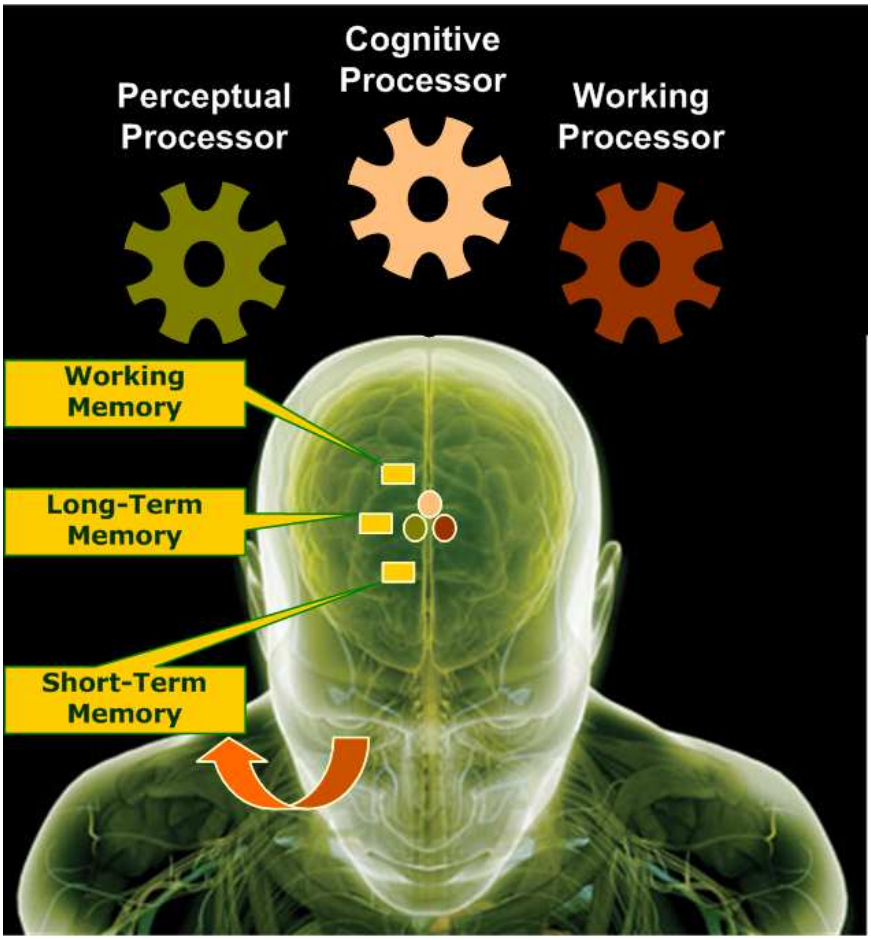


Figure 3.2: Conceptual View of the Human Brain. Short-term memory receives and temporarily stores visual sensory input. Cognitive centers and other memories act upon the received visual input

The general purpose of the above studies was successful in giving tangible understanding of the short-term memory characteristics. In our research, we attempted to prove a hypothesis related to these characteristics. In the later chapters we show that the short-term memory characteristics could be also unique from one individual to the other, and can be used as biometric factors.

3.4 Summary

In this chapter we presented previous work that has similarity with either our proposed system components (i.e., mouse and keyboard) or concepts (i.e., visual scan and detection and short-term memory). From the discussion on mouse dynamics, we note a higher success rate in using mouse dynamics biometrics in continuous authentications than in static user authentication. Attempts at using mouse dynamics biometrics in static user authentication are undermined by their high error rates, or the inconveniently long login sessions required.

We also observe a notable success in understanding, modeling, and quantifying human factors, especially the cognitive ones discussed. However, all the surveyed research studied these factors as collective human phenomena. In the next chapter we present the novelty of our research by combining cognitive and mouse dynamics factors in a biometric system for static user authentication.

Chapter 4

Physio-Behavioral Visual-Based Biometric

Biometrics in authentication use *something that we are* as a way to verify or identify one another. A physical or behavioral attribute needs to have certain characteristics in order to qualify as a good candidate for authentication (Section 4.6.2). Nevertheless, not all good candidates are practical to use. Many other integrant components affect building any new biometric-based authentication system. Drivers of some of our behavior differences are still mysteries, and hence cannot be modeled. An example of this is how visual distortions correspond to behavioral errors. Some other behaviors are possible to quantify only with high-caliber equipments; for example, measuring brain activities in response to different stimuli. As a result, these biometrics become locked in specialized research labs, and are not practical for deployment in security systems. Data-sensor precision is another factor that shapes the effectiveness of a biometric system. For instance, although there are 266 distinct features in the iris, not all of them can be effectively scanned through the reading sensors. Therefore, introducing a new biometric requires success in all of these areas, and not only in modeling a certain physiological or behavioral attribute.

In this chapter we introduce two human cognitive factors - visual scan and detection and short-term memory - as behavioral biometrics that satisfy the above criteria.

Both factors are explained in detail. Additionally, in this chapter we provide the design of a test framework system based on these two cognitive factors in addition to mouse dynamics. The rest of this chapter explains data gathering and feature extraction details.

4.1 A New Dimension in the Biometric Field

Studies in areas related to human perception and cognition are usually directed toward studying collective human behaviors and phenomena for psychology research [48, 58–61]. In other research, these attributes are studied to engineer better gadgets and systems for human use and interactions [46, 62]. The novelty in our research is that for the first time, two such attributes are used to discriminate between humans for user authentication purposes. The rationale of that hypothesis is derived primarily from the fact that all literature on human factors which provided numerical studies always estimated an error margin to the results. This error margin is usually due to measurement accuracy and/or human differences. That inspired us to assume that making the error margin the focal point could yield the discovery of a new biometric system based on these human factors.

Visual scan and detection, and the short-term memory are the two new human attributes this research introduces to the world of biometric-based security systems. Additionally, mouse dynamics is added as a third biometric component; this component has dual use in this research: as a behavioral biometric, and as a tool to measure the other physiological biometrics. This means we are introducing a homogeneous biometric system, one in which separate sensors are not needed to read the data for each biometric. Additionally, the proposed system maintains its simplicity in only using the mouse, given the complexity of the new introduced

physiological biometrics.

Our research philosophy is that under abnormal circumstances, biometric differences are more distinct. Normally, when a person is in a serene and un-stimulated state, the "firing" of neurons in the locus coeruleus is minimal, and hence people reactions in many aspects are difficult to clearly differentiate. However, the presence of a stressor causes the neural system to direct the body to continue performing well under pressure (stress response). Stress response is better differentiated from one person to the other. In the next section, we present the components of the proposed biometric system.

4.2 Biometric System Components

The proposed system homogeneously collects three different biometrics, namely the visual scan and detection capability, the short-term memory capability, and the mouse biometrics. Later in this chapter, features related to these attributes are discussed in further detail.

4.2.1 Visual Scan and Detection Capability

In Section 3.3.1, we explained the principles of the visual scan and detection cognitive factor. In a bottom-up model, the average time estimate needed to search for an object among N objects in a spatial field can be expressed by the following [46]:

$$T = \frac{N \times I}{2} \tag{4.1}$$

Where I is the average time one user takes to inspect an object during the visual search. The idea is to properly design an experiment that can measure the average inspection time I of an individual, and use that as biometric data. Two main design factors should be considered to isolate I from other influences:

1. Visual search has to be guided by the bottom-up model [62], in which cognitive factors, which are the top-down model, do not influence the search. For example, searching for a name in an unsorted list will require a sequential search across the entire list until the desired name is found (bottom-up); however, searching for the same name in a sorted list is influenced by the cognitive skills to perform a more efficient search (top-down).

2. Experiment should maximize the visual distortion in order to better differentiate between subjects' visual search capabilities.

In this research, we propose to use a virtual shuffled keyboard (Figure 4.2), where subjects are requested to visually scan the keyboard image on the screen to locate different random characters. The shuffling of the keyboard cancels any top-down effect, because the cognitive centers cannot in this case make any useful suggestions on where to look next. Every test session presents a new random keyboard with a new random text. As a result, average search time in locating a letter on the shuffled keyboard represents the subject's visual search capability to a high degree of accuracy.

Additionally, the choice of a shuffled keyboard over any other visually distracting stimulus has an additional advantage in maximizing the distraction factor. Objects of a shuffled keyboard (keys in this case) are very similar, except for the letter impression on each key. This makes it very difficult for the subject to develop other discriminating algorithms other than performing a sequential search, and inspecting the letter on each key.

4.2.2 Short-Term Memory Capability

As presented in Section 3.3.2, Posner's [57] experiments support our hypothesis that short-term memory has discriminative characteristics that could be used in the

proposed biometric system. To introduce these characteristics of short-term memory information retrieval as a biometric, the experiment is modified to measure that capability. The random text associated with each shuffled keyboard is customized to include some characters that are repeated to study that effect. Figure 4.4 illustrates an example of such character repetition.

4.2.3 Mouse Usage Dynamics

As was shown in Section 3.2, mouse dynamics have been successfully used as behavioral biometrics. In this research we capitalize on that, by introducing additional mouse dynamics features for better user verification accuracy. Additionally, scan and detection and short-term memory effects are also recorded through the mouse actions.

4.3 System Framework

The proposed architecture provides an integrated solution suite with tools needed at the different research stages. Our system main components are presented in Figure 4.1. It includes the following main components:

4.3.1 User Physio-Behavior Sensor Module

This module provides a configurable test environment. It can design and deploy different tests which target some or all of the human factors identified in section 4.2. The module offers two test modes. The first of these is the standard mode which is designed to collect the desired human factors in one test session. The second is the repetitive mode, which is designed to progressively collect data over a long period of time for the same user, with a primary focus on human factor related to memory skills. This module is composed of the following sub-components:

VISUAL DISPLAYS This sub-component generates all the visual displays. These include the QWERTY and shuffled keyboards, the login page, the explanatory screen, and the wait screen. It interfaces with the TEST GENERATOR sub-component to get the keyboard codes, and then translates them into visual keyboards. It also interfaces with the MOUSE TRACKER sub-component to sense and record all the mouse movements.

TEST GENERATOR This sub-component is the factory for building the keyboard codes and the random phrases that go with them. It first designs a keyboard code. Based on the generated code, it then constructs a random phrase that observes the types of test that need to be implanted within it.

SHUFFLER ENGINE This sub-component combines a collection of shuffling mechanisms that are used in the process of generating the shuffled keyboard codes.

MOUSE TRACKER This sub-component is instantiated by the VISUAL DISPLAYS sub-component to start recording user activities. It dispatches several sensor threads. Each sensor is responsible for detecting and recording a specific mouse activity, and they all synchronize recording the data into the initial logs. Minimal computing is performed in this sub-component.

4.3.2 Feature Extractor Module

This module processes user raw data collected from the sensor module, and produces a user feature vector. This passes through different data-processing stages. Using a customized knowledge-based system, data is cleansed from human behavior irregularities, and input and system errors. Primary quantitative features are then extracted from the clean data set. Using a customized outlier detection and removal

engine, the primary features are iteratively filtered to maintain homogeneous data set for each primary feature. In the final stage, other compound features are calculated as mathematical relations between the primary features. This module is composed of the following sub-components:

USER LOG READER This sub-component reads all the user recorded data from the testing sessions, and loads them into the memory. Since the logs do not follow a consistent sequence in recording the data, this sub-component has the logic rule necessary to handle this log randomness.

FEATURE EXTRACTION This sub-component processes the user raw data loaded, and extracts the identified features using the built-in rules.

PROFILE SETTING This sub-component uses the extracted feature to build a user profile that is later used for training and verification. This includes generating complex features by relating two or more of the extracted features.

DATA SERVICE This sub-component is composed of a set of utilities that provide a set of tools needed by other components in processing data. These include sort and search algorithms, mathematical functions, and tabulating and storing processed data.

PATTERN ANALYSIS This sub-component is used by the FEATURE EXTRACTION sub-component in analyzing data and examining patterns.

DATA FILTER This sub-component cleanses data at each and every stage of this phase. It removes bad data due to system malfunctions. It also provides a custom statistics tool responsible for detecting and removing outlier data. It also includes an API interface to the R-project statistical software for the more advanced outlier removal method.

4.3.3 User Identity Module

This module builds a user profile, verifies user identity, and authenticates users. Neural Network and computational statistics models are used independently in this module to ensure reliability.

4.4 Test Design

The experiment idea is to expose each subject to a sequence of specially designed interactive keyboard illustrations, as shown in Figure 4.2. In addition to the keyboard, a special phrase constructed of random characters is also displayed, and the subject is requested to "click in" the displayed phrase keys. The subject's response to distracting (shuffled) and non-distracting (QWERTY) keyboards is recorded through the mouse movements for later analysis and feature extraction.

In order to run a successful experiment, the test was designed using the following foundation factors:

- Exposure duration has been found to have strong impact on the effective value attributed to the visual stimulus [63]. Many studies such as [64] and [65] have found an opposite relationship between exposure duration and attention; the more the subject is exposed to the test, the less his attention gets. Therefore, the test-session duration could be at some threshold a confounding factor. In this experiment, the primary factor affecting the test duration is the number of characters in the random text. Therefore, a careful calculation of the random text length was essential to eliminate the exposure long-duration effect. A special experiment was designed for that purpose: three volunteer subjects were requested to do three test sessions at three different times. Each test session had a random text composed of 50 characters. Later, two essential features, *Speed*

and *Fly Time*, were extracted and compared against the random text length, as illustrated in Figure 4.3. The results show that users are showing consistent behavior when exposed to a string up to about 25 characters in length. Users' recorded behaviors start changing significantly and become inconsistent when the string length is between 25 and 30 characters. Based on this observation, and to be on the safe side, the random text length of this research experiment was chosen to be 25 characters.

- Short-term memory factor test is implanted in the experiment by selectively duplicating some of the letters in the random text string, as illustrated in Figure 4.4. Duplicated keys' measurements are later studied to extract the short-term memory factor effect.

- During visual scan, the corners of the displayed keyboard image fall near the outer edges of the retina periphery. Image corners are usually regarded as points of curvature discontinuity edges along the contours of detected boundaries [66]. Retina periphery is only capable of transmitting low resolution information to the brain [67] [68], and that is not sufficient to detect a desired key. Therefore, in order to search the corners for a desired letter, a subject has to make relatively long saccadic movements for the corners to be foveated. The experiment measures the subject's foveated primate vision efficiency at the corners by placing some of the random text letters at the corners, as illustrated in Figure 4.4.

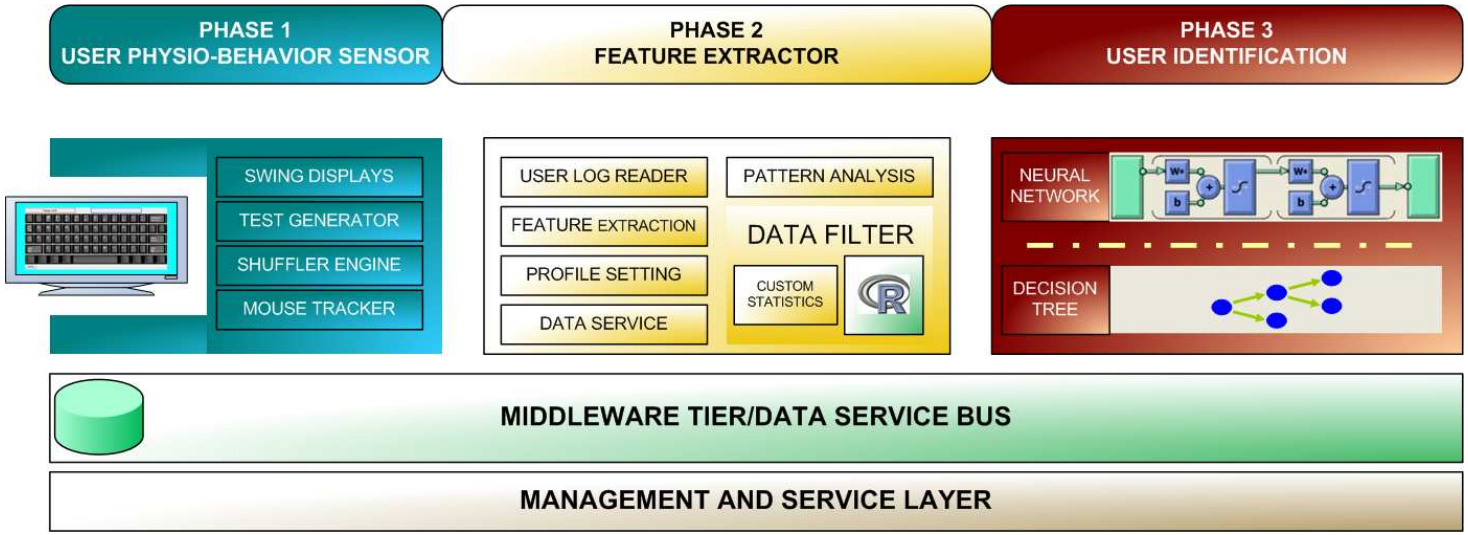


Figure 4.1: Physio-Behavior System Architecture

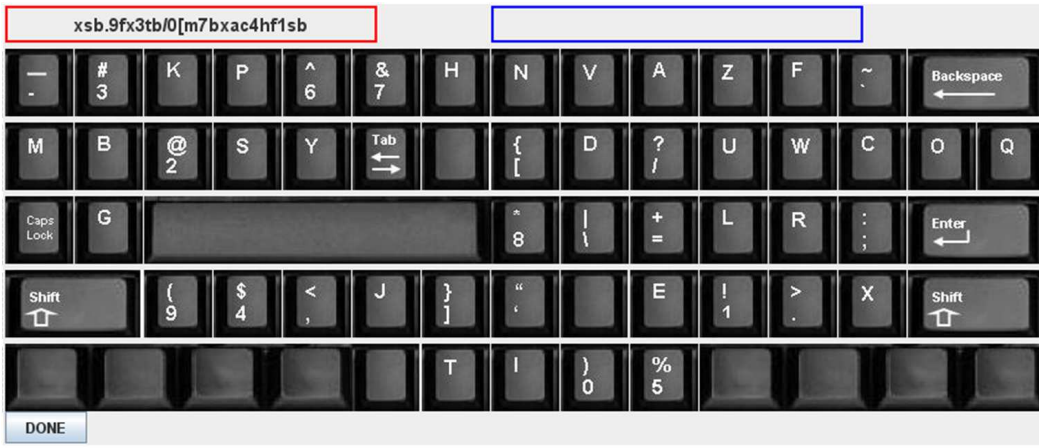
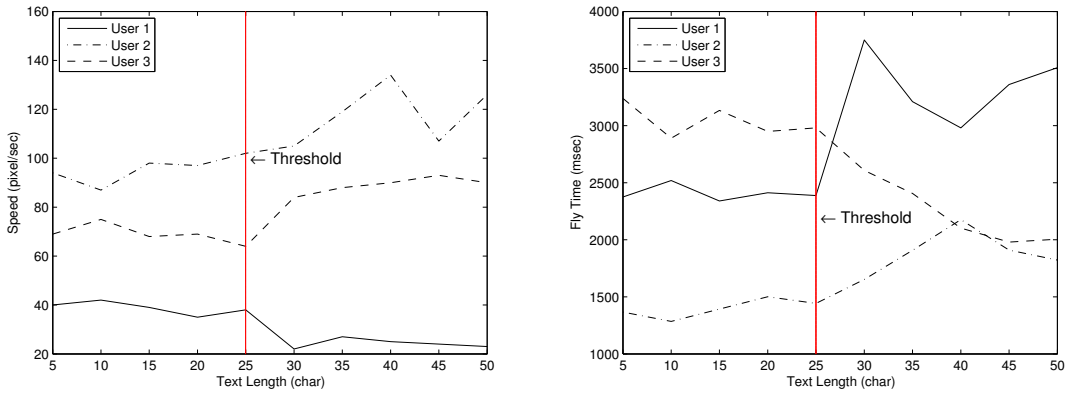


Figure 4.2: Sample Shuffled Keyboard. The top left text box displays the random text the user is requested to click in. User clicked keys are displayed in the right text box.



(a) Speed Test

(b) Fly Time Test

Figure 4.3: Text Length Threshold: For both Speed and Fly Time, users showed a change in behavior when text length is between 25 and 30 characters.For this study, the threshold is taken at 25 characters, to be on the safe side

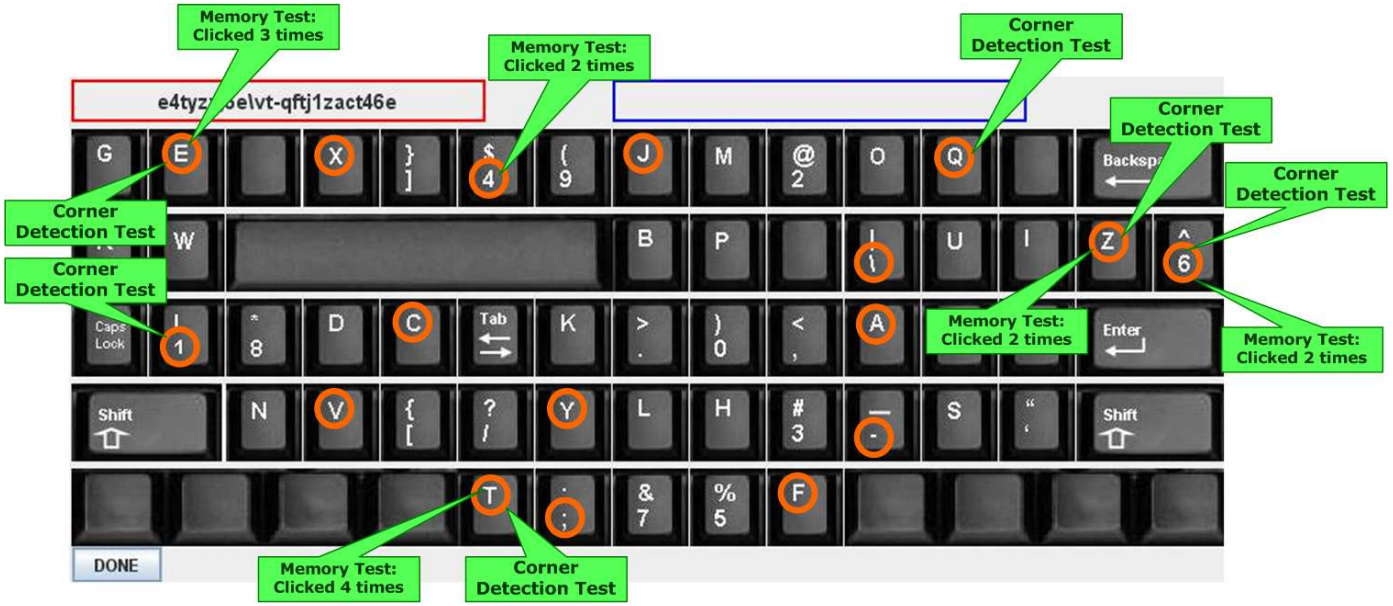


Figure 4.4: Sample Embedded Test for Corner Detection and Memory Factor. The random text is customized to have some of the letter locations serve as a test for the short-term memory and the corner detection human factors.

4.5 Data Acquisition

In this section, we describe data characteristics and acquisition procedures and challenges. To design our system, we selected a small subset at random from the data collected for the experiment purpose. Details of the data collection will be given in the Experimental Evaluation Procedure (Section 6.1).

4.5.1 Preliminaries

Raw data acquired during the experiment faced some fundamental challenges that had to be addressed at the early experiment design stage.

Human Factor The collection of biometric data falls into the category of human subject testing [69]. Participating subjects voluntarily give their biometrics. This raises the data confidentiality concern. Currently, there is no internationally approved standard procedure that details how subject confidentiality is preserved. There are, however, isolated efforts from individual institutes to submit proposals for Human Subject Research Guidelines, such as [70]. In this research, we observed a number of related measures that we believe satisfy the confidentiality issue:

1. Subjects were given thorough explanation of the experiment details and overall goals, but without informing them about the type of collected biometrics.
2. Subjects were given locally generated IDs which were not associated with their true identification. For future further verification, only a small group of identities were mapped to generated IDs and kept confidential in a secure place.

3. Upon extracting subject biometrics, raw data collected is removed permanently.

Novelty Cost In known physiological and behavioral biometric systems, raw data could be acquired through publicly available databases such as the European Languages Resource Association (ELRA), Fingerprint Verification Competition (FV2000), Multimodal Face and Voice Database (M2VTS), and NIST Standard Reference Data. Such databases significantly cut the research time. The novelty of this research and the type of collected data resulted in losing this privilege. Additionally, in order to build high confidence in the proposed biometric system, we had to raise the number of participating subjects. Typical research in this field requires between 20 and 40 participants. In this research, we successfully enrolled 274 participants.

Acquisition Iterations The data acquisition process had to go through iterations until raw data sufficiently matured to fulfill the desired feature extraction. Iterations had to be minimal and efficient, because each iteration failure means to discarding valuable raw participant data.

4.5.2 Collected Data

The raw data collected in our experiments consist of 18 parameters described in Table 4.1; Table 4.2 shows sample raw data collected for one user. During any test session, raw data is not recorded in the logs in the same sequence for each key click. This randomness in recording data is caused by two things: First, the instantaneous action logging functionality of the system makes it difficult to tabulate data at the input time. Second, raw data capturing is done through five concurrent threads working at different data segments; the synchronization mechanism of the program does not always grant write privilege to threads in the same order. Figure 4.5(a) shows a partial

segment of the initial raw data collected for one key search and click. It includes information about the keyboard used such as the generated code, and the sequence in the experiment session. It records all the clicked key information, such as location in the keyboard, key value, and whether a Shift key was needed or not. Additionally, all mouse movements that were needed until the key is clicked are logged; these include Fly Time, Wait and Fly Time, and Click Time. Furthermore, the system records other user behaviors during that time, such as the movement style. The movement style is defined as hesitation and oscillation, and trip continuity. Figure 4.5(b) shows how data logging does not follow the same sequence all of the time.

Upon the completion of the test session, the raw data shown in table 4.1 are processed. First, a data health check is performed. Next, calculations are performed to generate additional primitive attributes such as distance traveled, and speed. Processed data are then organized in a tabular format, as shown in Table 4.2, for the next stage of processing.

4.5.3 Feature Analysis

This phase is an intermediate stage between the raw data initial processing, and the feature extraction stage. In this phase, more complex features are introduced by establishing all possible relations between different attributes: for example, impact of key occurrence over all other collected attributes. When the same key is repeated in the random text string, the user starts building awareness of that special key. This affects the other attributes such as fly time and speed the next time that same key is observed. Another example is the influence of the key display location over other attributes. Edge detection capability measurement is different from one user to the next, and therefore attributes related to finding and moving to keys at different edges are expected to be different among users.

Table 4.1: Collected Raw Data Types

Parameter	Units	Parameter Description
Test Type	-	Refers to either shuffled or QWERTY test session type.
Session	-	Refers to the session number the user is currently in.
Key Order	-	Refers to the order that a key appears at in the random text.
Key	-	Refers to the value of the clicked key.
Key Index	-	Refers to the location where a key appears on the keyboard.
Occurrence	-	Counter for the number of times that a key appears in the random text.
Move Direction	-	Describes the location of the clicked key on the keyboard in terms of direction. Directions are: Left Uppers Corner (LU), Right Upper Corner (RU), Left Bottom Corner (LD), Right Bottom Corner (RD). All other locations are identified as RANDOM.
W-n-F	ms	Wait and Fly is defined as the short silence time (no mouse activity) right after a key click until the user decides on the next move.
W-n-C	ms	Wait and Click is defined as the short silence time (no mouse activity) just before the key click.
FT	ms	Fly Time is defined as the elapsed time while the user is actively (continuous mouse activity) searching for the next key.
CT	ms	Click Time is a measurement of the average user's mouse click time speed.
Actual Distance	pixels	Defined as the total distance measured in pixels that the mouse traverses between two key clicks.
Calculated Distance	pixels	Defined as the shortest distance between the last clicked and the current clicked key, measured in pixels.
Actual Speed	pixels/s	Defined as the average speed of the mouse during movement between two key clicks. It is calculated by dividing the Actual Distance over the Fly Time.
Calculated Speed	pixels/s	Calculated by dividing the Calculated Speed over the Fly Time.
Traveled X-Distance	pixels	Defined as the total distance measured in pixels the mouse traverses in the x-axis direction.
Traveled Y-Distance	pixels	Defined as the total distance measured in pixels the mouse traverses in the y-axis direction.
X2Y	-	Defined as the ratio of Traveled X-Distance to Traveled Y-Distance.

Table 4.2: Tabulated Raw Data of One User. The first nine variables are displayed in the top table, and the remaining variables are continued in the bottom table

Test Type	Session	Key Order	Key	Key Index	Occurrence	Move Direction	W-n-F (ms)	W-n-C (ms)
2	6	14	l	8	0	LD	42	2
2	6	15	h	42	0	RANDOM	128	2
2	6	16	6	36	2	LU	824	1
2	6	17	2	24	1	RU	1404	1
2	6	18	d	13	0	LD	97	2

FT (ms)	CT (ms)	Actual Dist. (pixels)	Calc. Dist (pixels)	Actual Speed (pixel/s)	Calc. Speed (pixel/s)	Traveled X-Dist (pixels)	Traveled Y-Dist (pixels)	X2Y
16509	113	1797	407	108	24	1321	692	1.91
1046	98	145	283	123	240	97	104	0.93
1515	160	241	352	102	150	166	103	1.61
11686	137	1127	74	86	5	850	461	1.84
1058	126	115	173	99	149	78	46	1.70

```
Keyboard Code: 48-40-28-44-46-49-1-30-38-17-3-41-2-47-9-
26-32-22-51-18-6-11-34-39-14-0-27-31-43-37-36-16-35-29-
42-7-13-10-24-5-33-21-19-15-50-45-4-12-20-25-8-23
Test Type:      s
Sequence:       2
Double Clicked:      false
Boundary:       normal
W-n-F:    1063
FT:       1531
W-n-C:    2
Pixel Segmentation: 53 23 89 41 31 109 40 50 49 23 117 42
Shift Status:    false
Corrected:       false
Key Location:    31
Key: f
CT:  105
Oscillation:          false
Hesitation Level:     high
Continuous Trip:      false
Click Location: UR
```

(a) Sample recording of activities of mouse movement until a key is clicked

```
Keyboard Code: 48-40-28-44-46-49-1-30-38-17-3-41-2-47-9-
26-32-22-51-18-6-11-34-39-14-0-27-31-43-37-36-16-35-29-
42-7-13-10-24-5-33-21-19-15-50-45-4-12-20-25-8-23
Test Type:      s
Sequence:       6
FT:       2592
Double Clicked:      false
Boundary:       normal
W-n-C:    17
W-n-F:    587
Oscillation:     True
Shift Status:    false
Click Location: RANDOM
Corrected:       false
Key Location:    23
Key: k
CT:  133
Hesitation Level:     low
Continuous Trip:      false
Pixel Segmentation: 101 78 77 69 86 79 81 66 74 97 64 83
```

(b) Another sample recording of activities. Depending on the Java Virtual Machine (JVM) Synchronization between data recording threads, captured activities can be logged at different sequence

Figure 4.5: Initial Recording Format of One Key Click. Sequence of entries in the log could be different as shown in (a) and (b)

4.5.4 Data Filtering Algorithms

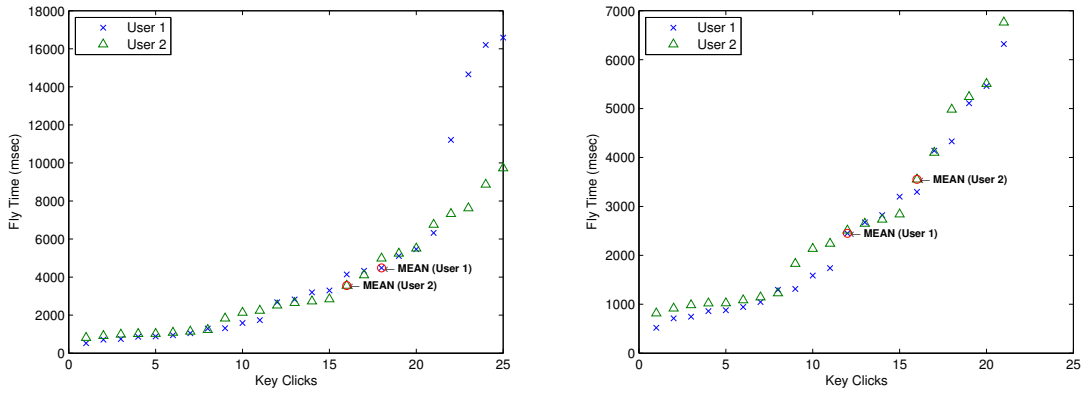
Part of the intermediate data processing phase is the data filtering. Data filtering was divided into two separate actions: data cleansing and outlier removal.

Data cleansing is the process of cleaning data from any system malfunctioning. The primary source of any malfunctioning in the system is the fact that the system is designed to handle human actions and behaviors. The amount of identified exceptions that special routines are designed to handle adds layers of complexity to the system. For example, a participant could change the OS focus by clicking on an area outside the GUI frame; this causes interruption to some of the threads monitoring user actions. The OS could be a second source of data-recording problems in some special cases due to I/O module glitches, or during memory page swaps.

In this stage of data cleansing, a special process is designed to scan and remove all records that show signs of malfunctioning infection. An example of a bad record would be an entry that does not have key value recorded.

The second data filtering action taken is outlier removal. An outlier in a set of data is defined as an observation (or subset of observations) that appears to be inconsistent with the remainder of that set of data [71]. Outliers are frequently generated as the result of the natural variation of a population or process, which one cannot control [72]. In this research, outlier is defined as heterogeneous values of subject extracted features. In many cases outlier values are detected as beyond human bound ranges. For example, by examining the collected data of the sample users, fly time between two clicks of less than 150 ms is beyond human bounds. This type of outlier is the easiest to detect and remove. With enough samples from a large subject group, boundaries could be accurately estimated. The more complicated type of outlier values is when detected values are discordant to a particular subject's ranges. Since biometric systems are based on comparing user recorded measurements, outliers could

cause a genuine user to be rejected, or more seriously, an imposter to be accepted as genuine. Figure 4.6 shows the outlier effect on calculating the fly time mean of two subjects.



(a) Data set with outlier

(b) Data set without outlier

Figure 4.6: Fly Time Mean: (a) When data is not filtered, user 1 has a higher Fly Time average than user 2, and (b) when outlier is removed, Fly Time average of user 1 becomes lower than that of user 2.

In physiological biometric systems, outlier detection and removal is a relatively straightforward task. That is because the data values of the extracted features, such as iris features, are relatively stable. The outlier source is usually in the feature reading device, or feature extraction mechanism precision [73].

In behavioral biometric systems, outlier removal becomes a subjective matter. By definition, human behavior is a collection of exhibited actions that are influenced by attitude, emotions, values, skills, genetics, mental state, health, and others factors. Therefore, designing a behavior-recording system will have to consider the range of variation of certain behaviors. It becomes a challenge how to distinguish between an outlier behavior element, and simply a genuine subject behavior change. For example, during the experiment, a participating subject might stop to do something

like sneezing. That will affect the recorded times, speed, and silence time. On the other hand, another participating subject might get frequent absent-minded periods in the later test sessions. The first scenario is a clear example of an outlier, and has to be detected and removed from the data set. The second is debatable, and not necessarily to be considered an outlier.

Different approaches have been developed to evaluate biometric sample quality such as the one proposed in [74]. In this research, we built a hybrid outlier detection and removal model that monitors the nature of each collected behavior attribute. First, 50 sessions from five random subjects (with 10 sessions per subject) were used to study each attribute, and to propose an outlier removal mechanism. The proposed mechanism was then used over another five random subjects. Produced results from the latter group matched the manual analysis. This result gave confidence to generalize the proposed outlier removal model, and use it over all the other users' samples.

Collected data from any participating subject passed through the following steps:

1. Predefined boundaries for each attribute were used to trim the collected data set.

2. Based on attributes'data distribution and likelihood, attributes were segregated using predefined rules for each.

3. Depending on the data quality, the average was then calculated:

   (a) for the stable data set, the average is either calculated after the data was trimmed using 95% confidence interval calculation, or by considering the median instead of the mean.

   (b) for the more scattered data, such as speed and fly time, the Minimum Volume Ellipsoid (MVE) technique was used to cleanse the data before

calculating the average. An API was built to interface with the R-Project statistics software for that purpose.

4.6 Feature Extraction

In order to produce the feature golden set, data of 30 sessions from three random subjects (with 8 shuffled keyboard sessions and 2 QWERTY keyboard sessions per subject) were analyzed. All drawn conclusions were then verified against data of 50 sessions from five other random subject (with 8 shuffled keyboard sessions and 2 QWERTY keyboard sessions per subject) for generalization purposes.

4.6.1 Large Number of Possibilities

The first observation we recoded was the number of features that could be compounded. At ground zero, there were 11 independent parameters collected that were considered features (Table 4.2, starting at the W-n-F column) such as speed, fly time, and distance. The next level of complexity added features by observing changes in one feature as influenced by others: for example the change in fly time average when keys are located at the different corners of the keyboard image. We were able to identify more than 60 additional compound features at this level. Moreover, some features allowed establishing complex relations across them. For example, it was observed that if the search for a letter is prolonged, some participants tend to slow their mouse speed, while others go even faster as a sign of frustration. At this level, 20 additional features were identified. Furthermore, we were able to isolate additional features by observing the way features change from one state to the other. For example, when the subject clicks the same key more than once throughout a test session, she starts building a memory awareness of the location of that key. The new feature is defined as the threshold location, or *Knee* location, when the rate

of change of some features starts increasing or decreasing in an apparent different slope, as depicted in Figure 4.7. For this particular feature, however, although data values were distinctive among users, data values of different sessions of the same user showed poor reproducibility, and hence the feature was not considered. This is shown in Figure 4.8. By making these types of special observation, we were able to produce an additional 10 possible features. That gave a total of more than 100 candidate features that the system can use for identification and verification.

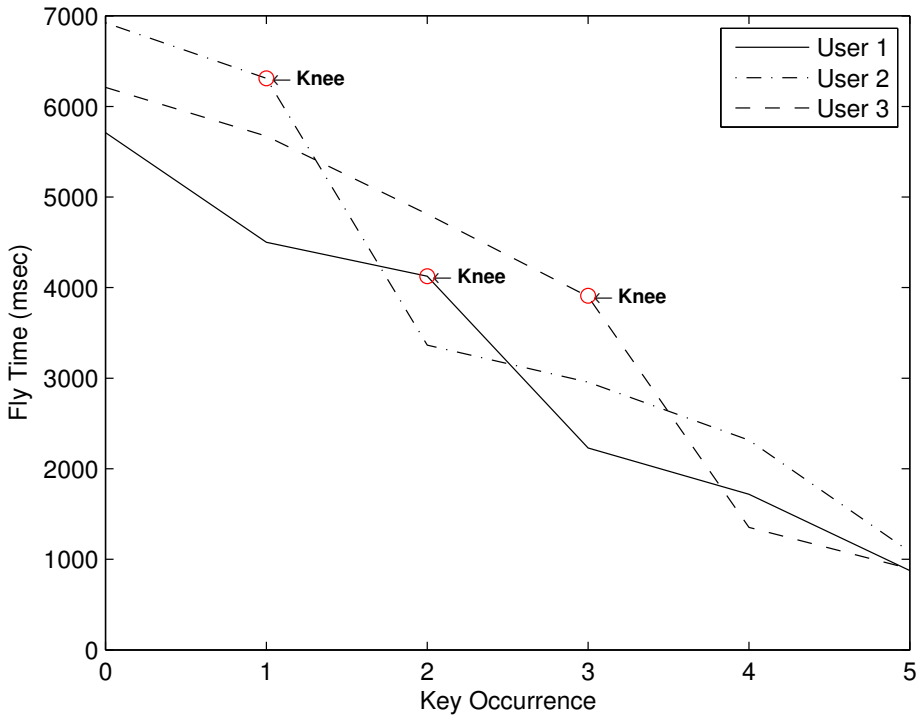


Figure 4.7: Impact of Memory Over Fly Time for Three Subjects. At a certain occurrence, referred to as *Knee* point, the Fly Time drops significantly because of the memory awareness impact. The *Knee* point is different from one user to the other, and hence could be a biometric feature candidate

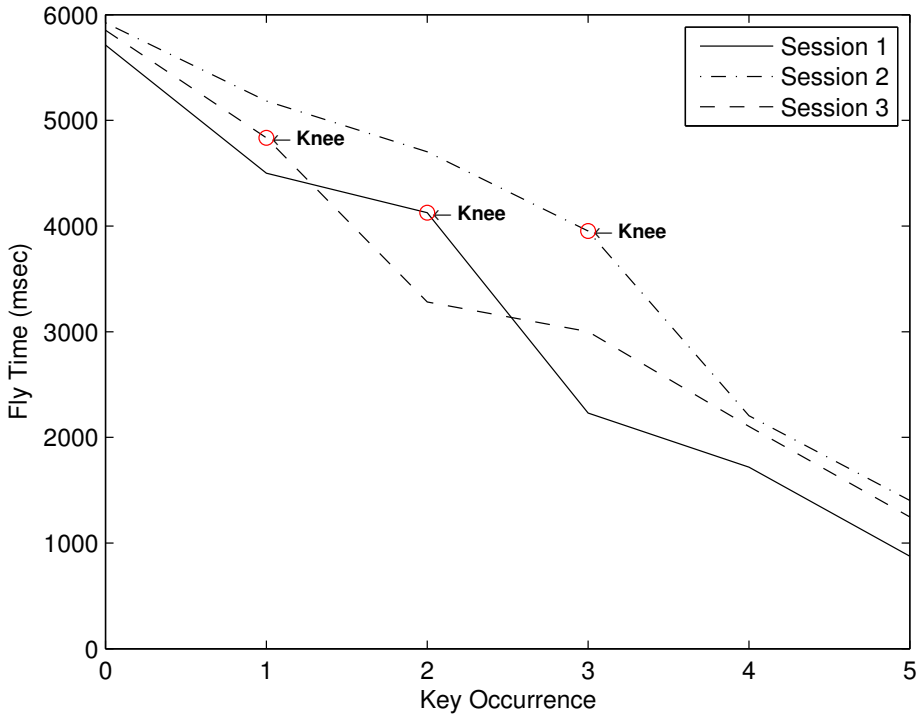


Figure 4.8: Inconsistent Recordings of Memory Impact on Fly Time for One Subject. When memory impact was studied using sessions from the same user, it showed inconsistent Knee location. Hence, the corresponding feature has poor reproducibility

4.6.2 Feature Selection Criteria

In section 4.6.1, we discussed the large number of features that could be calculated and constructed from the collected raw data. It was not practical then to do an exhaustive analysis to test each feature's candidacy to be among the biometric set. Rather, a more efficient methodical approach was followed to identify the golden set suitable for the scope of this research. More specifically, the following criteria were considered:

Suitability Feature candidacy is determined according to the relevancy of the feature to the biometric system components, as presented in section 4.2. Any candidate feature can be classified under one or more of the biometric divisions.

Distinctiveness In order to achieve better biometric classification results, it is important that used features have mean values different from one subject to the other. Close averages or overlapping regions produce poor classification results.

Stability Features with stable data sets are those with less variability among their data members. In other words, data values of features of the same user are reproducible from one session to the next, with relatively small difference. In order to test for stability, we chose a cross-correlation analysis technique to determine features' data variability level. In other biometric research [73], it is suggested to use 0.9 correlation as the cutoff point.

Singularity Features that satisfy all the previous observations can still need further filtration in order to remove duality. By duality we mean that two or more features possess the same behavior in response to different test states: for example, traveled distance, total time, and speed. Since speed is derived from the other two, it is redundant to keep the three features; only two are sufficient.

4.6.3 Final Feature List

The feature selection mechanism explained in Section 4.6.2 was used to filter the proposed features, and finally produce the golden set used in this research. Ten sessions from each of five random subjects were used to filter out the features that did not meet the mechanism's criteria, and only keep the ones that matched. Due to the large number of feature groups, this process was done iteratively to refine selections. Features that were finally selected as the basis for the proposed biometric system are summarized in Table 4.3.

To discuss further, feature samples of five different users are shown in Table 4.4. Looking at the data of any column shows how selected features satisfy the selection criteria in section 4.6.3. For example, the data set for one user has relatively small standard deviation, while it is significantly different compared to other users' data sets.

4.7 Summary

In this chapter we showed that the human cognitive features of visual scan and detection and short-term memory are possible biometric factors. By adding mouse dynamics, we were able to model a testing framework that uses the standard mouse as the data input device. The experiment followed a methodical approach in collecting and filtering data, and extracting features. In feature analysis, we used certain selection criteria to produce the final feature list to be used in biometric analysis. Biometric analysis is detailed in the next chapter.

Table 4.3: Biometric Feature Set

Feature	**Biometric Component**	**Feature Description**
W-n-F	Visual Scan & Detection and Short-Term Memory Groups	Wait and Fly is defined as the short silence time (no mouse activity) right after a key click till the user decides on the next move.
FT	Visual Scan & Detection and Short-Term Memory Groups	Fly Time is defined as the elapsed time while the user is actively (continuous mouse activity) searching for the next key.
FT Q2S	Visual Scan & Detection Group	Defined as the ratio of average Fly Time in QWERTY to shuffled keyboards of the same user.
FT OCC	Short-Term Memory Group	Measures the Fly Time improvement rate as the one key occurrence increases in one test session.
DST Q2S	Visual Scan & Detection Group	Defined as the ratio of average Traveled Distance in QWERTY to Shuffled keyboards of the same user.
DST OCC	Short-Term Memory Group	Measures the Traveled Distance decrease rate as the one key occurrence increases in one test session.
X2Y	Mouse Usage Dynamics Group	Defined as the ratio of traveled X-axis distance to the Y-axis distance in Shuffled test sessions.
Speed	Mouse Usage Dynamics Group	Defined as the average mouse speed in Shuffled test sessions.
CT	Mouse Usage Dynamics Group	Click Time is a measurement of the average user's mouse click time speed.

Table 4.4: Tabulated Raw Data of Three Sessions per User. Among one user's sessions, values of each feature are relatively close (stable), but different (distinctive) from one user to the other.

USER	**W-n-F**	**FT**	**FT Q2S**	**FT OCC**	**DST Q2S**	**DST OCC**	**X2Y**	**Speed**	**CT**
USER 1	1074	4603	0.63	0.87	0.36	0.58	2.01	48	171
	1021	4733	0.59	0.85	0.39	0.50	2.48	50	183
	1002	4587	0.60	0.88	0.32	0.49	2.37	47	176
USER 2	906	2376	0.78	0.40	1.02	0.42	1.37	22	122
	954	2340	0.79	0.52	1.07	0.38	1.40	22	137
	819	2412	0.77	0.48	1.03	0.46	1.30	22	137
USER 3	1410	1834	1.10	0.71	0.99	0.36	1.54	22	119
	1524	1965	0.94	0.64	1.12	0.39	1.67	17	28
	1493	1964	0.94	0.73	1.10	0.41	1.52	16	137

Chapter 5

Biometric Analysis

In order to verify the system proposed in the previous chapter, it has to be analyzed using methodical approaches. In this chapter, we explain the analysis framework to measure the effectiveness of the system in user authentication and identification. We do that analysis using different neural networks and computational statistics models. In this research, we consider two forms of biometric analysis:

User Authentication Defined as the process of validating that a user is who she claims to be. It compares input template of an identity claim against a stored genuine template of that user.

User Identification Defined as the process of distinguishing a user from a set of possible candidates [75]. It compares an anonymous user's template against all the stored users' templates to give a *match* with one, or *no match* with any.

5.1 Biometric Analysis Using Neural Network

Biometric analysis in this section is done using Neural Network, known for its powerful generalization capability [76]. User authentication process is divided into two phases:

creation of the user's master template, and verification using an efficient classification mechanism. The user's master template uses the golden feature list defined in section 4.6.3 to train a customized neural network. The produced template is used in the second phase to evaluate a user claim (verification) by comparing her input with the stored master template.

5.1.1 Neural Network Design

Neural network design has always been an open question for researchers [1] [77]. This is because a large number of parameters need to be carefully selected to design a neural network. Since many neural network designs can solve the same problem, performance and accuracy become the two factors in selecting one design over another. To narrow the scope of search, the following key design parameters were the only ones considered:

Neural Network Type Depending on the nature of the problem, different neural network architectures are suitable for different set of problems. Different comparative studies such as [78], [79], and [80] presented performance change in response to neural network architecture change.

Number of Layers Number of layers should be carefully considered when designing a neural network. The optimum number of hidden layers should be large enough to ensure sufficient number of degrees of freedom for the network function and small enough to minimize the problem of loss in generalization ability of the network [81].

Number of Nodes Nodes which are the building blocks of a neural network architecture could vary in number from one layer to another. Experimentally, number of nodes in each layer could be optimized to give a better performance and accurate results.

Transfer Function Depending on the problem class, different transfer functions are known to be most suitable. For example, in the identification problem class, Log Sigmoid is widely used.

Training Algorithm Some training algorithms are known to shorten the neural network learning time, while others are for accuracy. Training algorithm could also be chosen based on the nature and sensitivity of the input data.

In order to reach a good neural network architecture design, experimental approaches are frequently used[1]. Therefore, to design a suitable network for our research, data from 70 sessions of seven users (with 8 shuffled keyboard sessions and 2 QWERTY keyboard sessions per subject) were used to test different possible neural network prototypes. It is important to note that this was not meant to be an exhaustive exercise; this means that not all possible combinations and variations were tried. In other words, it is possible that with extra experimentation, a better design may be found that gives higher accuracy. In this research, the starting point was the famous Feed-Forward network with one hidden layer and Log Sigmoid transfer function. Then iteratively, a few parameters were modified, and results were observed. The primary goal was to achieve high classification accuracy. Other factors such as computing resource consumption, or size of data storage needed, were considered as less consequential in this research. This is because the backend component of this application is expected to be designed for and hosted in the data center of a corporate organization or institution, where computing resources are not a limiting constraint. Based on the above consideration, the proposed neural network design is depicted by Figure 5.1 and described in the following.

The proposed architecture features the following design aspects:

[1]There are other methodical approaches that can be used to optimize the design such as [82].

- Feature vectors[2] are the input to the neural network for training, and later for verification purposes. The neural network produces output ranges from 0 to 1, where 0 represents a perfect nomatch, and 1 represents a perfect match result. In neural network, however, output is not usually that distinct. Instead, output values between 0 and 1 exclusive are produced, which we call confidence values. Based on a predetermined threshold, an output confidence value is determined to be a match or non-match. The threshold is experimentally determined to reach the best performance. In user verification for example, the threshold is tuned to make it harder for a user claim to be falsely accepted; in user identification, the threshold could be tuned to make it harder for an identification accusation to be dismissed.

- Two hidden layers are used in addition to the input and the output layer.

- The first hidden layer and the output layer use linear transfer function, while the second hidden layer uses Sigmoid transfer function.

- The first hidden layer is composed of 15 nodes, while the second hidden layer has 20 nodes.

- MATLAB MAPMINMAX normalization algorithm was used to normalize the input data, as shown in Equation 5.1. Given a feature x, the normalized value y is defined as follows:

$$y = \frac{(y_{max} - y_{min}) \times (x - x_{min})}{x_{max} - x_{min}} + y_{min} \tag{5.1}$$

MATLAB initializes y_{min} and y_{max} to -1 and 1, respectively.

[2]A feature vector is the golden set features group of one session of one particular user.

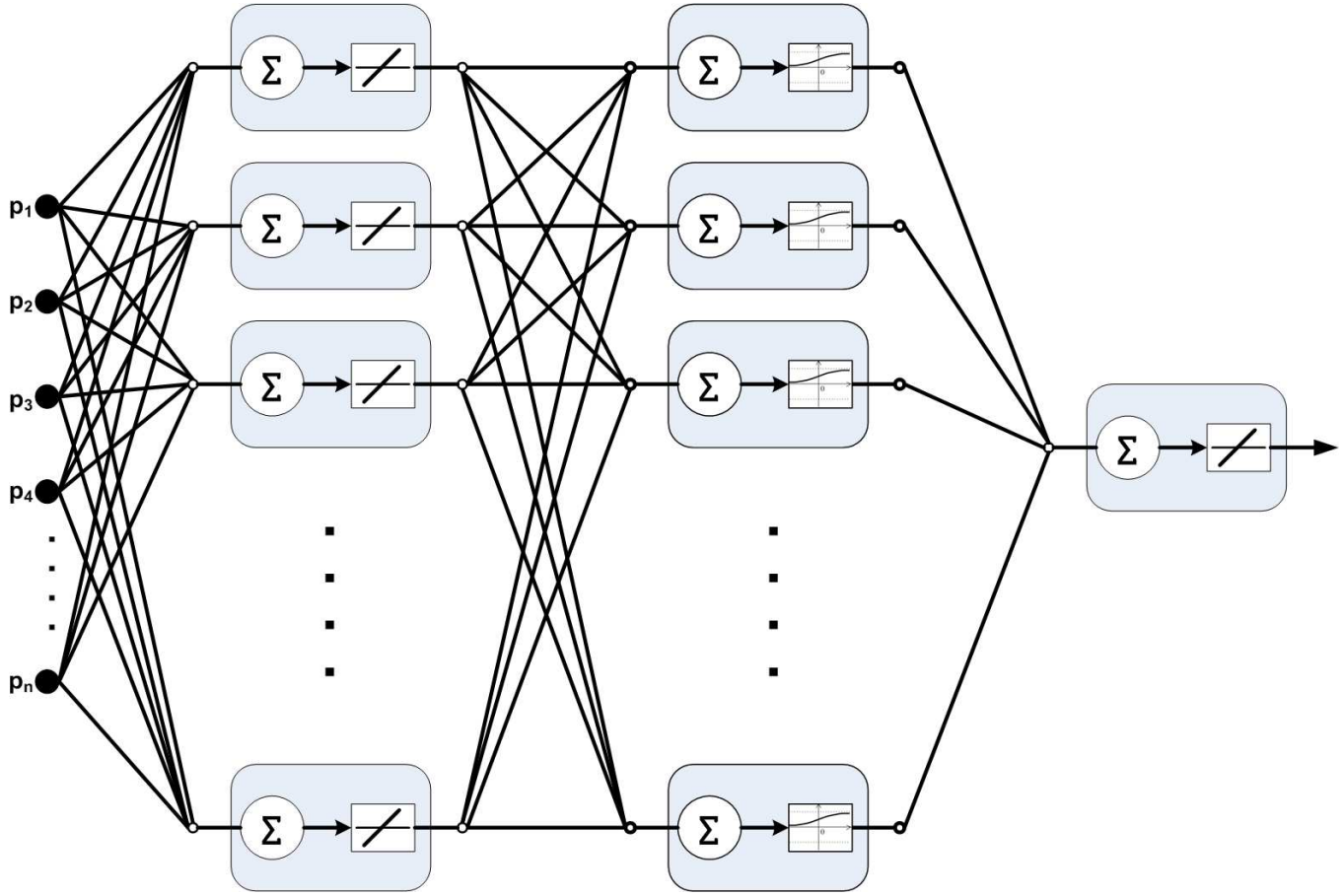


Figure 5.1: General Neural Network Design Proposed. The input layer corresponds to the feature list used. The first hidden layer has 15 nodes and uses linear transfer function. The second hidden layer has 20 nodes and uses sigmoid transfer function. The output node uses linear transfer function

5.1.2 Enrollment Strategy

Biometric enrollment is the process of building user profiles for later verification and identification. In this section, neural network is used as the enrollment mechanism. In a typical access control system, users are divided broadly in two categories: authorized users and unauthorized users, to whom we will refer as legal users and imposters, respectively. A common evaluation strategy for such systems consists of dividing the user population into two different sets corresponding to these categories.

Our strategy to build a profile for a genuine user consists of training a neural network using sample data from only the legal group. By using the genuine feature data samples from a user being profiled (*self data*), as well as data samples from other legal users (*non-self data*), the neural network generates and stores a master template for that user, which will be recalled later for authentication or identification purposes. This process is summarized in Figure 5.2. The sample data subset that is used for the neural network training is also divided into two groups. The first phase of the training is called *Zero Knowledge*. Data group 1 is used in this phase to train the neural network. After training is complete, training proceeds to the second phase, which is called *Proof of Knowledge*[3]. The second data group is used in this phase in order to verify that the network successfully profiled that user. Once the training is successful, the generated master template is stored in the users' signature repository database. Successful enrollment is measured based on a predefined threshold of the confidence value the neural network produces. To accept a user profile, the network is supposed to give a confidence value that exceeds the threshold when a user *self* sample is fed to the network. Unfortunately, the neural network is not always successful in building the user profile. This is known as *Failure to Enroll (FTE)*. In general, *FTE* could be due to various reasons; if the user biometric samples suffer from high

[3] *Zero Knowledge-Proof of Knowledge* terminology was suggested in [83]

variance, this could cause neural network failure. Other types of biometrics such as iris scan, require high-precision systems, in which case *FTE* could be due to system design issues.

In this research, *FTE* is mainly due to a participating subject being distracted during some of the testing sessions. For example, in one incident, a subject answered a phone while doing three of the test sessions. During the phone call, attention capability, short-term memory, and mouse movements were all impacted by the subject dividing the attention between the test and the call. This resulted in three out of ten sessions being very different from the rest of the sessions. Table 5.1 shows sample data of one subject that the network was not able to train throughout almost all trial rounds. In one training trial round, five sessions are used for the *Zero Knowledge*, and the remaining two are used for *Proof of Knowledge* phase. The other trials would do the same, but with different session combinations. For that particular user, average of all trials shows a confidence value for the *self* sessions of 45.7%, which is below the predefined threshold set at 50%.

Training a neural network is not a straightforward task. In fact, modeling the neural network data feed could be as challenging as designing the neural network itself. If too many or very few parameters are used, statistical efficiency can be severely impaired [84]. Using smart trial and error approach, different input modeling techniques were tried to make a good judgment of the network's result accuracy, and network convergence and divergence. As a result of this exercise, three data input techniques were proposed:

1. *Feed All* technique. In this case, the neural network gives all the input parameters equal importance in training. This model is the same as the general neural network design shown in Figure 5.1.

2. *Divide & Fuse* technique. In this model, input data are clustered into two

separate feature groups. Each group is fed to separate but identical neural networks. The two final outputs are given the same weight and fed to an adder to produce the final output. This model is shown in Figure 5.3.

3. *Divide & Select* technique. This model is similar to the *Divide & Fuse* technique. The difference is in the mechanism used to fuse the two partial outputs together. Instead of averaging out, the two outputs compete to be picked as the representative of the overall output. The champion is the one that has a stronger *yes* (close to 1) or *no* (close to 0). In the case that both are hesitant or giving opposite answers, they are averaged to produce the final output. This model is shown in Figure 5.4.

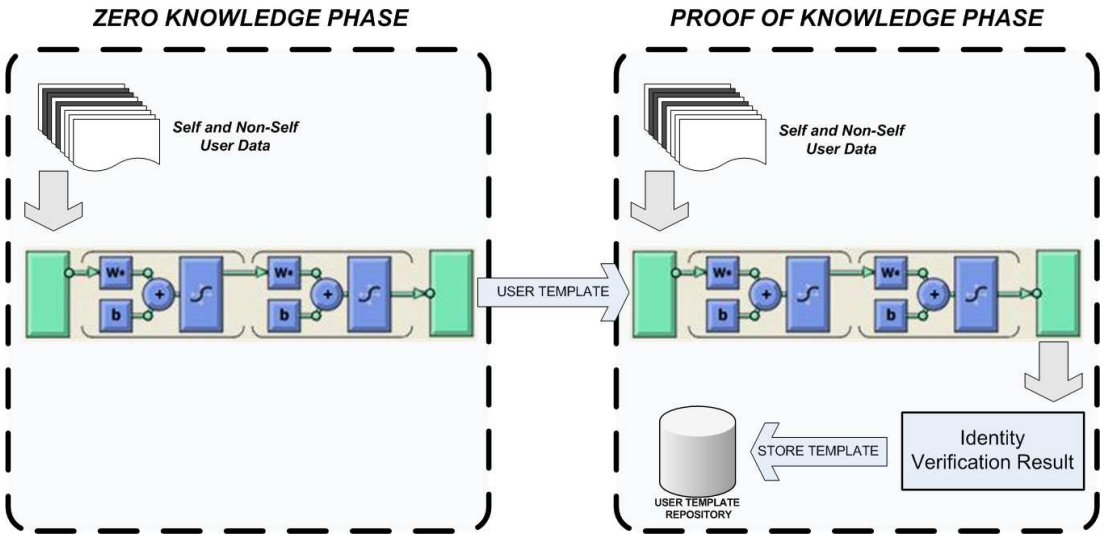


Figure 5.2: Neural Network Training Phases. In the *Zero Knowledge* phase, true and false user data is used in training. In the *Proof of Knowledge* phase, network is initialized using the produced template. Output is then compared with the actual results; when successful, user template is stored

Table 5.1: Sample *FTE* Results. Throughout all but one of the training trials, the neural network on average produced below 50% threshold confidence values for the user *self* sessions.

Trial	**Session**	**CV (%)**	**Average (%)**	**Training Status**
Try 1	1	44	47.5	Failed
	2	51		
Try 2	2	51	43.5	Failed
	3	36		
Try 3	3	38	43	Failed
	4	48		
Try 4	4	61	58	Passed
	5	55		
Try 5	5	59	46	Failed
	6	33		
Try 6	6	37	36.5	Failed
	7	36		
Try 7	1	50	45.5	Failed
	7	41		

It is worth mentioning that grouping the features was done in a selective manner. In one trial, features of similar nature were chosen to be in the same group to ensure homogeneity among the group. In another trial, features of different nature were put together to ensure diversity among each group. Comparing the results of the two prototypes showed that the first setting gave better performance results than the second setting. It was then used in both the second and the third input techniques described above.

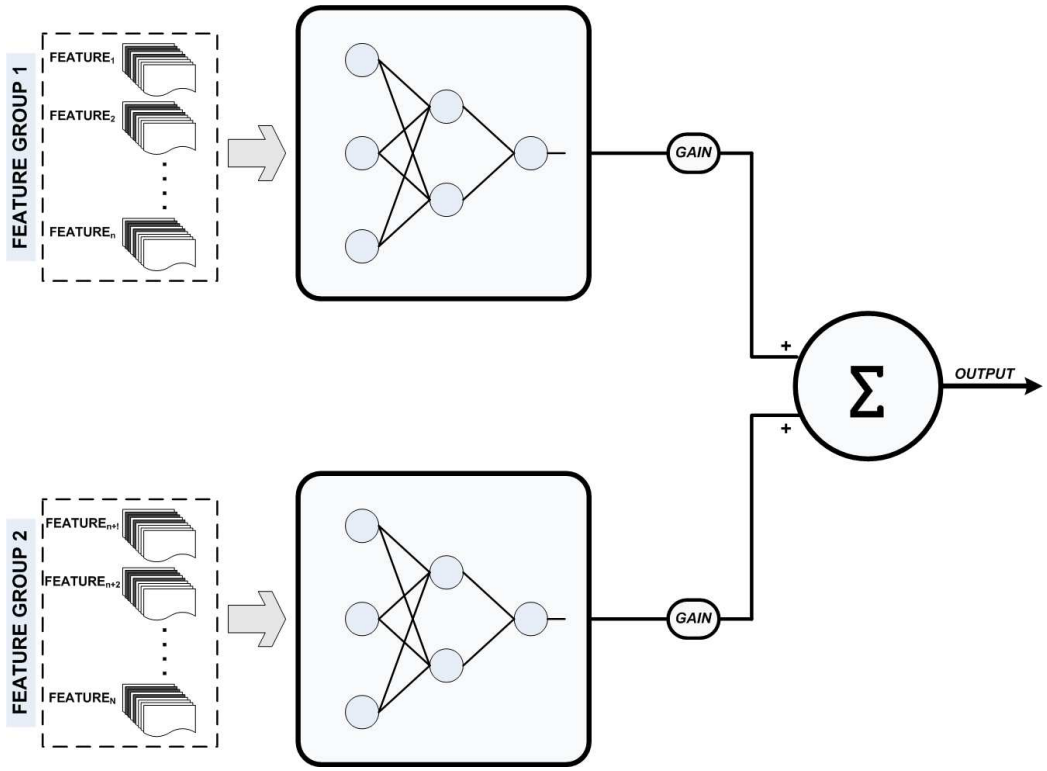


Figure 5.3: *Divide & Fuse* Training Technique

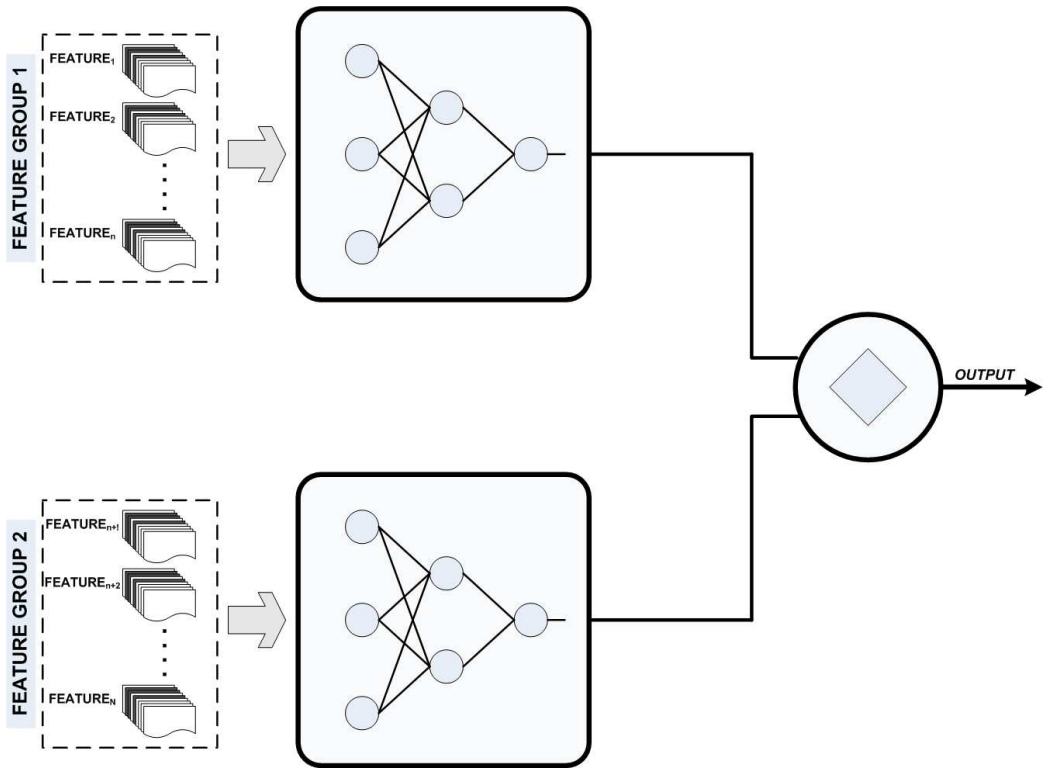


Figure 5.4: *Divide* & *Select* Training Technique

5.1.3 User Authentication

After the training is complete for all users, the *User Template Repository DB* now has a genuine master template for each user. User templates can later be retrieved by a *user ID*, which could be a user name, or a generated user code. In a practical setting, the verification process will start by displaying a login page, which includes user name input field, and a shuffled keyboard. The user will click in her passcode phrase using the mouse. Collected raw data will follow the normal processing sequence, and a feature vector will finally be generated. This feature vector will then be fed into the neural network to verify the user identity claim.

For the scope of this experiment, however, verification starting point is at feeding user features to the network. The test platform uses the assigned test sessions to generate test cases. For each user, an input file is created. Genuine sessions of the user are flagged as *self* and added to the file, and a copy of one test session taken from every other user's test sessions is added after flagging it as *non-self*.

For each user, the verification process begins by reading the *user ID* from her input file. That ID is used to retrieve the user's master template from the template repository database. After the neural network is loaded, test entries from the user input file are sequentially fed into the neural network, and the output is cached. Upon completion of all test cases, the average number of False Acceptance *FA* and False Rejection *FR* are calculated by comparing the cached output vector with the record flags in the user input file. This test process is explained in Figure 5.5.

5.1.4 User Identification

The goal of user identification is to use an anonymous biometric sample to determine the actual identity of the the subject from among all users known to the system. This process requires a one-to-many matching against all stored profiles in the DB.

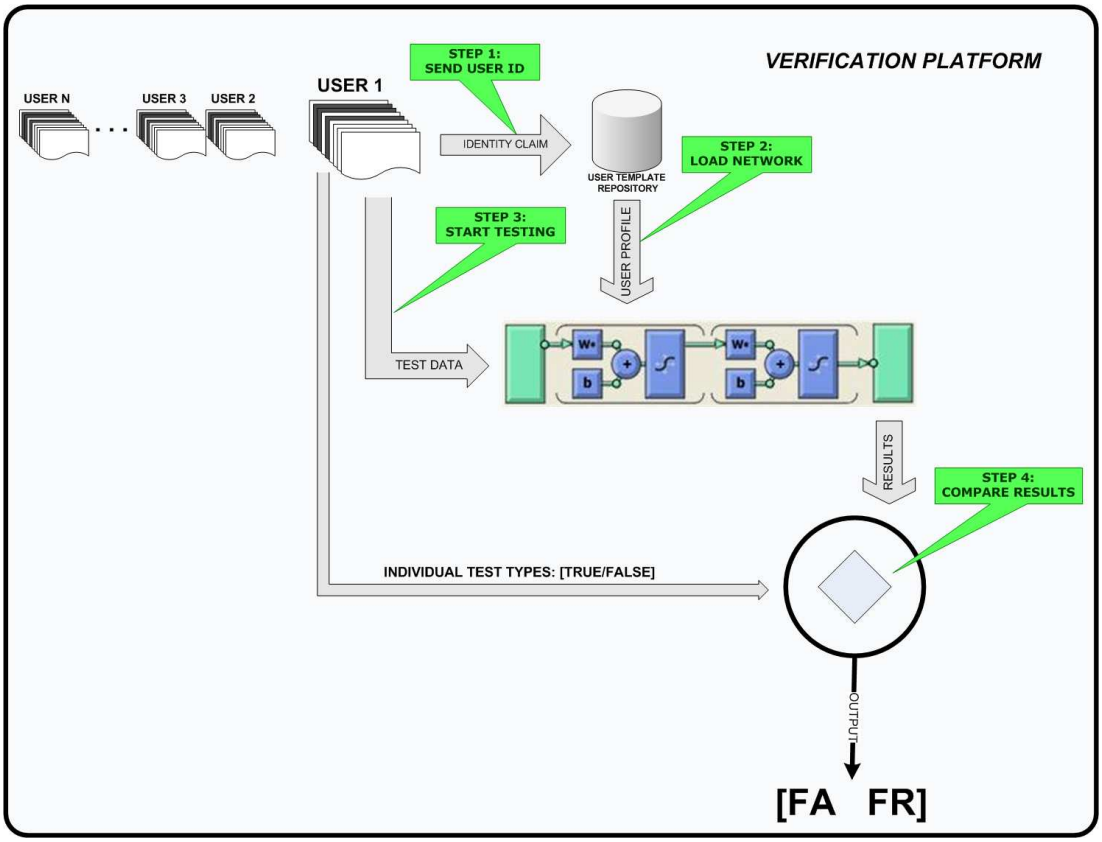


Figure 5.5: Verification Process

In this research, the user identification test starts by generating one input file composed of test cases. Test cases are composed of all the test sessions of all the users that were not used in building user profiles. Additionally, each test case has the feature data values of one session of a particular user and is tagged with the user ID for later comparison.

Opposite to the authentication experiment, the identification process proceeds by reading one test session from the input file, and sequentially the neural network loads master user templates from the template repository database. The anonymous test session is verified against each of the loaded master templates. Output *Confidence Value (CV)* of each master template verification is then cached with the master template ID.

After all verifications on that input session are complete, the template owner of the highest produced CV is selected as the identified subject.

Upon completion of verification of all test sessions in the input file, a *Misclassification Ratio (MR)* is calculated as the ratio of the number of mismatch identity sessions to all verification attempts. Identification procedure is illustrated in Figure 5.6.

5.2 Biometric Analysis Using Computational Statistics

In classification problems when statistical distribution of data is unknown, empirical evaluation techniques such as neural networks are superior to the statistical methods [85]. However, the generalization rules in the neural networks models are computationally complex. Additionally, finding a neural network model with good learning performance is always a challenge.

In our research, since the clustering of feature data is known, this gives an advantage to computational statistics models over the neural networks in terms of accuracy and speed.

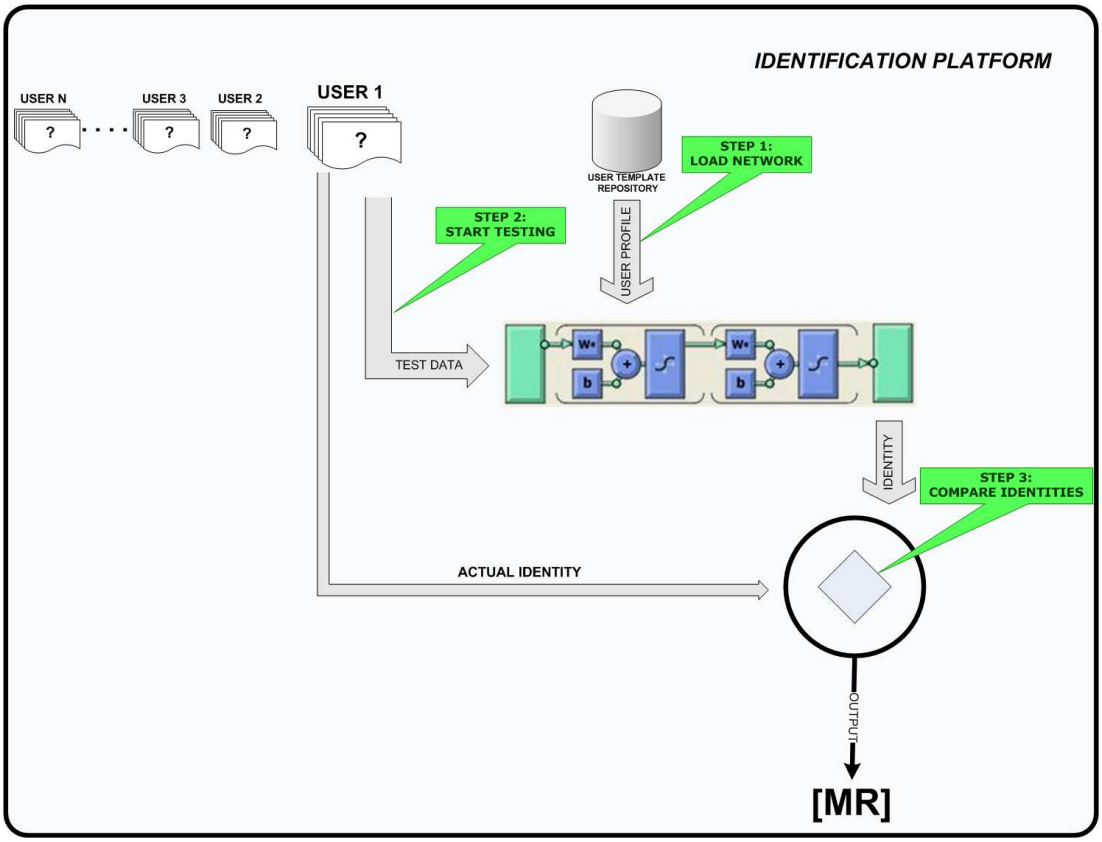


Figure 5.6: Identification Process

Furthermore, performance achieved using the neural networks experiments (presented in Chapter 7) was another motive to try the computational statistics approach to try to enhance the performance.

Computational statistics techniques are based on clustering and discriminant analysis. Clustering groups features data values of users using different clustering techniques. Discriminant analysis on the other hand uses supervised classification techniques to relate features data values of an anonymous user to one of the clusters (users'profiles) [86].

5.2.1 Clustering Analysis

Clustering is a common technique for statistical data analysis. There are many types of clustering such as hierarchical, petitional, and spectral clustering. Suitability of one technique over the other depends primarily on the type of problem, nature of data, and size of analysis space [87]. The methodology success is primarily dependent on finding a decision rule efficient enough to classify patterns with the minimum misclassification rate. This decision rule is designed based on the probability distribution of the input features data vectors. In this research, we follow a supervised nonparametric method in designing the decision rule. Supervised refers to the learning method that takes place once a decision rule is produced; this method is appropriate here because the learning input data is all from known classified subjects. The nonparametric characteristic is because for each user, data values of each feature do not follow a known (parametric) data distribution.

After studying other related literature, we decided to build a simple clustering algorithm that is using some of the principals of the *K-Means* clustering algorithm. The *K-Means* algorithm is one of the simplest and fastest clustering techniques that

is commonly used in biometrics and related fields.

We describe the process of building a statistical classifier for each subject in the following steps.

5.2.2 Reference Signature

During the enrollment, each user will provide a number m of sessions data used to build her reference signature. Each session j ($1 \leq j \leq m$) consists of an n-dimensional feature vector denoted $x_j = [x_{ij}]_{1 \leq i \leq n}$.

Let X denote the enrollment matrix, obtained by grouping the enrollment sessions:

$$X = [x_{ij}]_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}}$$

For each feature i ($1 \leq i \leq n$), let $x_i = [x_{ij}]_{1 \leq j \leq m}$ denote the corresponding enrollment row vector. We assign the elements x_{ij} of x_i to a cluster, with centroid c_i and maximum distance $dmax_i$. For each feature row vector x_i ($1 \leq i \leq n$), we calculate the cluster's centroid c_i and range $dmax_i$ such that:

- c_i is calculated as the mean value of the members of the cluster:

$$c_i = \frac{1}{m} \sum_{j=1}^{m} x_{ij}$$

- $dmax_i$ is calculated as the largest *Euclidean Distance* between the centroid c_i and any member $x_{ij} \in x_i$ of the cluster:

$$dmax_i = \operatorname*{Max}_{1 \leq j \leq m} distance(c_i, x_{ij})$$

We now define the user reference signature to be the matrix $[C\ D]$ where:

$$C = [c_i]_{1 \leq i \leq n}$$

$$D = [dmax_i]_{1 \leq i \leq n}$$

Decision Rule

A decision rule is a function that processes an input vector and matches it against a reference signature producing an identity confidence output. In this research, we introduce two decision rules derived from kernel density estimation:

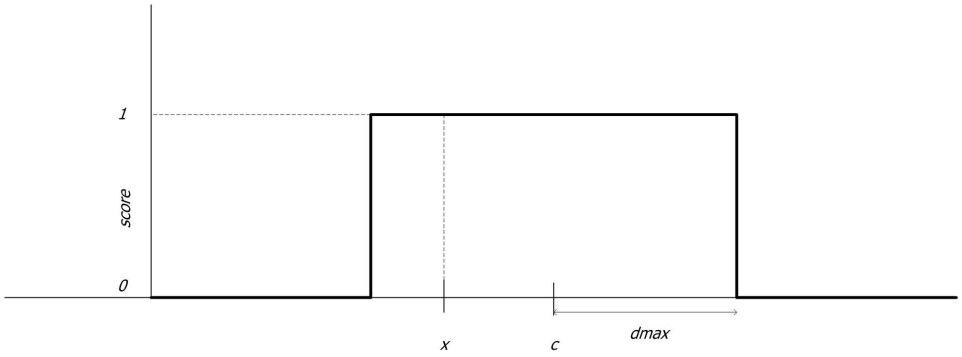
1. *Step Rule*: This rule decides on whether a feature value belongs to a cluster or not in a binary mode. For an input feature data vector $Y = [y_i]_{1 \leq i \leq n}$ and for each feature i $(1 \leq i \leq n)$, the following score is produced:

$$score_i = \begin{cases} 1 & \text{if } distance(y_i, c_i) \leq dmax_i \\ 0 & \text{otherwise} \end{cases}$$

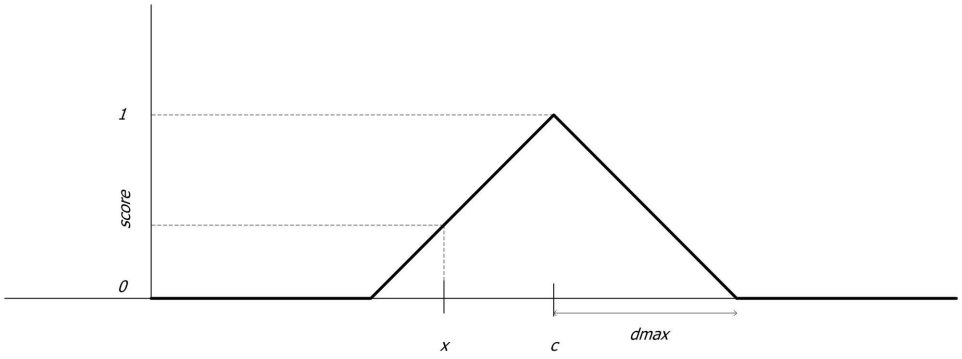
2. *Triangle Rule*: This rule is different from the *Step Rule* in that for the feature values that fall within a cluster, a decimal value $(0 \leq v \leq 1)$ is produced depending on how close they are from the centroid point. For an input feature data vector $Y = [y_i]_{1 \leq i \leq n}$ and for each feature i $(1 \leq i \leq n)$, the following score is produced:

$$score_i = \begin{cases} 1 + \frac{y_i - c_i}{dmax_i} & \text{if } distance(y_i, c_i) \leq dmax_i \ \& \ y_i \leq c_i \\ 1 - \frac{y_i - c_i}{dmax_i} & \text{if } distance(y_i, c_i) \leq dmax_i \ \& \ y_i > c_i \\ 0 & \text{otherwise} \end{cases}$$

Figure 5.7 shows a graphical illustration of the two decision rule functions. Given the input vector $Y = [y_i]_{1 \leq i \leq n}$, let $S = [score_i]_{1 \leq i \leq n}$ denote the score vector obtained using one of the above decision rules. This score vector shall be used in the biometric fusion techniques explained in the following section.



(a) Step Rule



(b) Triangle Rule

Figure 5.7: Statistical Decision Rules: (a) *Step Rule*: a binary score is produced based on whether or not an input feature data value belongs to the feature data cluster. (b) *Triangle Rule*: a normalized value between [0,1] is produced based on the distance between input feature data value and the cluster centroid

Biometric Fusion

Unlike the neural network model, biometric data fusion in our computational statistics model is done at the individual feature matching score. Biometric data fusion at the matching score level is known to maximize the utilization of the information from each biometric feature [88]. This makes the decision process transparent, which allows for better tuning of the system performance. On the other hand, biometric fusion for the neural network model is done at the feature extraction level. This is because in general, neural networks training and classification performance are proportionally related to the number of features data input. Figure 5.8 illustrates the score matching fusion model.

Using the normalized score vector S generated in the previous section, we define the following two fusion methods:

1. *Simple-Sum (SS)*: In this method, features normalized scores produced from the decision rule function for all features are arithmetically added to produce the final *Confidence Value (CV)*. This method is expressed in Equation 5.2.

$$CV = \sum_{i=1}^{n} \mathbf{score}_i \tag{5.2}$$

2. *Weighted-Sum (WS)*: In this method, features normalized scores are given different voting power based on their contribution in minimizing the misclassification rate. The contribution is expressed in terms of the *EER* produced from each feature. To calculate the voting power of each feature, data of sample population of 10 users were used. The system used 70% of the users' sessions for building profiles, and the remaining 30% for testing. At different thresholds, *FAR* and *FRR* were recorded. The tabulated values of *FAR* and *FRR* were used to find the *EER* value for each feature.

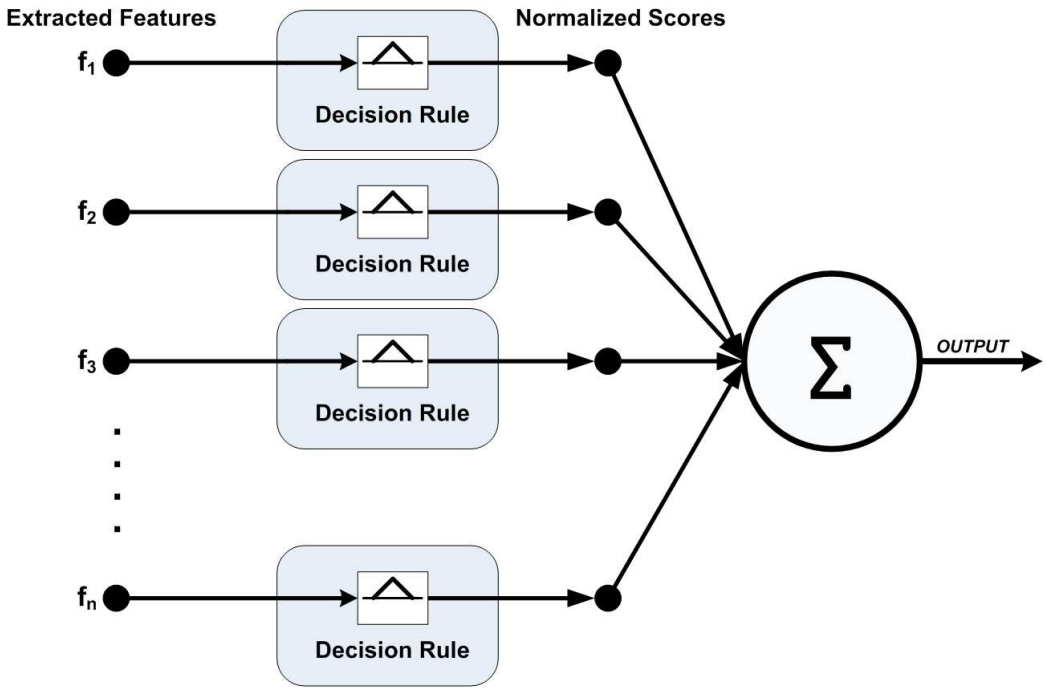


Figure 5.8: Score Matching Fusion Method

Section 7.3.2 gives a detailed explanation on how to calculate *FAR* and *FRR* using individual features. In this research we calculate the voting weights of each feature normalized score using the inversely proportional relationship presented in [88]. For an *EER* rate r_i ($r_i > 0$), voting weight w_i is shown in Equation 5.3.

$$w_i = \frac{1}{\mathbf{r}_i \sum_{k=1}^{n} \frac{1}{\mathbf{r}_k}} \tag{5.3}$$

We note that $\sum_{i=1}^{n} w_i = 1$. Table 5.2 shows the calculated *EER* and w_i for each biometric feature. The confidence value output based on the *Weighted-Sum* is then calculated as shown in Equation 5.4.

$$CV = \sum_{i=1}^{n} \mathbf{w}_i \ \mathbf{score}_i \tag{5.4}$$

Table 5.2: Individual Biometric *EER* and *Weight (w)*. *Weight* is inversely proportional to *EER*. We also note that $\sum_{i=1}^{n} w_i = 1$.

Feature	*EER* (%)	*Weight (w)*
W-n-F	31.6	0.0950
FT	19.4	0.1547
FT Q2S	26.7	0.1124
FT OCC	18.2	0.1649
DST Q2S	32.3	0.0929
DST OCC	35.1	0.0855
X2Y	24.4	0.1230
Speed	30.2	0.0994
CT	41.7	0.0720

For the *Step Rule* technique, the confidence value output is an integer number o ($0 \leq o \leq n$). Similarly, for the *Triangle Rule* technique, the confidence value output is a real number r ($0 \leq r \leq n$). Based on a predefined threshold value,

confidence value output is used as a match/nomatch boolean indicator. An *EER* is then achieved experimentally, by changing the threshold predefined value. Figure 5.9 gives an illustrative flow of all the previous described steps.

5.2.3 Enrollment Strategy

The enrollment mechanism in the proposed computational statistics model is fundamentally different from the one presented in the neural network model. In the latter, the enrollment mechanism for one user requires sample data from that user (*self data*), as well as data samples from other users (*non-self data*). *Non-self data* are used for *negative training*.

In the computational statistics models presented in this research, only *self data* are required to enroll a user. Feature data vectors are randomly divided into two groups, training group and the testing group. For each user, the training data subset is used to numerically configure the decision rules for each feature. The configured rules for all features are then stored as the user template in the user repository database.

5.2.4 User Authentication

As explained in Section 5.2.3, a training session group for each participant is used to configure the decision rules for each feature. The testing session groups are now used to measure the authentication accuracy of the system. For each participant, the test input file is created using the participant's test session group marked as *self*, and all the sessions of all other participants marked as *non-self*. This increase in the number of the *non-self* test sessions is due to the fact that configuring the decision rules does not require negative training, and hence, all *non-self* sessions are considered new to that participant's profile.

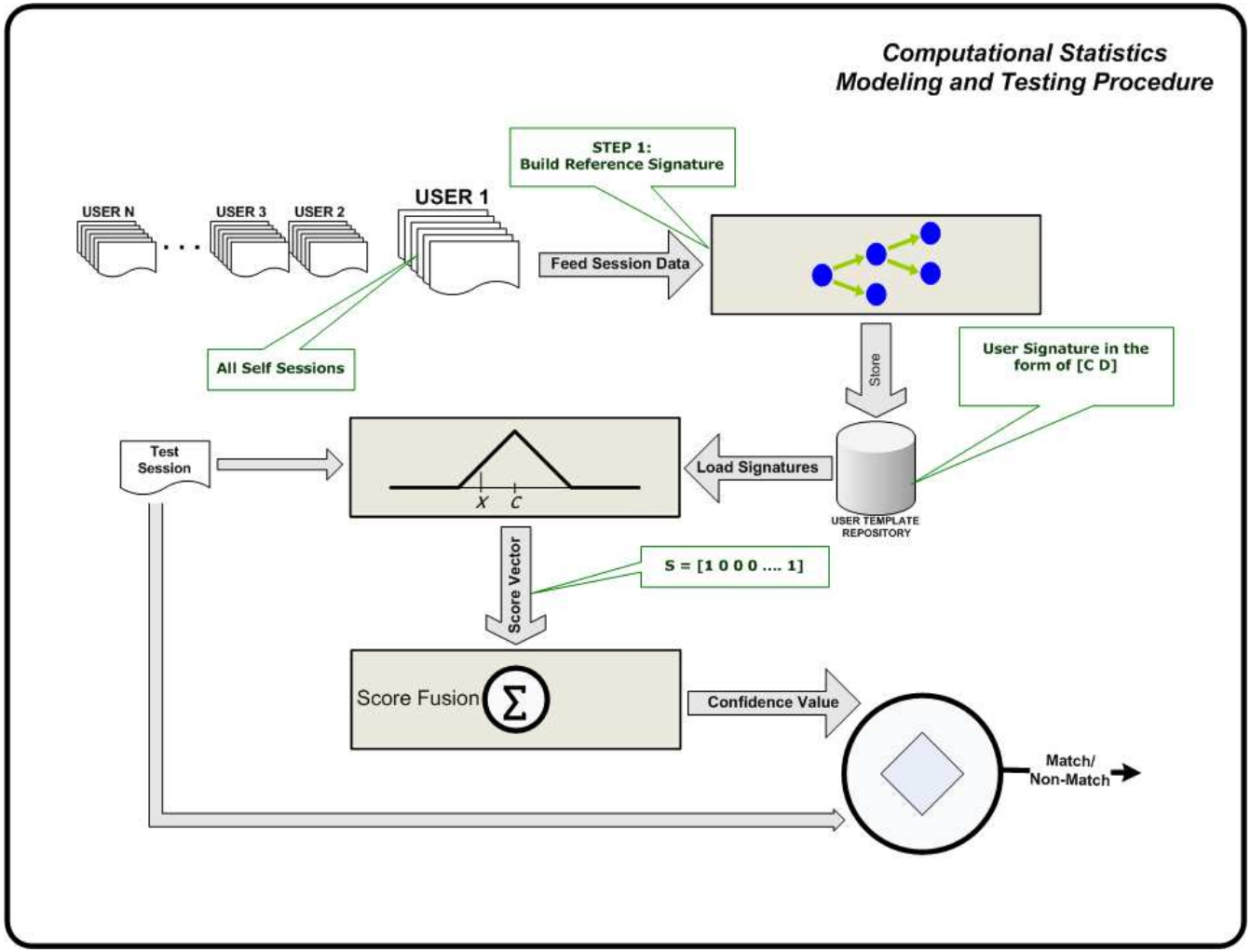


Figure 5.9: Computational Statistics Procedure

The rest of the procedure is identical to the authentication procedure using the neural network model explained in Section 5.1.3.

5.2.5 User Identification

The identification procedure is similar to the process used in the neural networks analysis explained in Section 5.1.3. We start by building test cases composed of users' self-sessions that were not used in building users'profiles. Each test session is fed to the system. The system sequentially loads each user's profile and generates a *Confidence Value (CV)* from the output score. The final output is the user whose profile scored the highest *(CV)*. To verify the result, this output user is compared with the actual identity of the test session.

5.3 Summary

In this chapter, we explained in details different models used for biometric analysis in our proposed system. As we discussed, there is no one known method that could be considered the best analytical approach. Therefore, we explored different methods using empirical approaches. For the Neural Networks analysis, we presented three different models, differentiated by their fusion methods. Similarly, we presented two different computational statistics models that differ by their score fusion rules. In the following chapters, we detail our experiments to measure the system performance based on all the above models.

Chapter 6

Experimental Evaluation

In this research, a large-scale experiment was conducted to evaluate our system. The hypothesis in this experiment is to establish that the proposed framework can be used to discriminate between different users, and recognize users based on their physiological and behavioral characteristics, which were shown in Table 4.4.
In this chapter we detail the experimental evaluation steps and approach. For the neural network and computational statistics models, we explain the enrollment procedure and performance calculation method.

6.1 Procedure

Each participant was seated in a non-distracting testing environment. By non-distracting we mean there were no audible continuous or burst noises, and no attention distraction such as sudden movements or conversations. Furthermore, no visually distracting elements were present in the participant's Wide Field of View (W-FOV). Participants did the experiment without prior knowledge of recorded behaviors or actions. They initially were given a detailed description of the experiment procedure followed by a short demo to get them comfortable. Participants were then asked to

get seated in such a way that their gaze was pointed approximately at the center of the display. The experiment was composed of 10 consecutive sessions; two sessions used QWERTY keyboards, and the remaining eight sessions used unique shuffled keyboard for each session. Later, one of the QWERTY keyboard sessions will be used for training, and the second session will be used for testing. Random text for each session was generated based on the randomly generated keyboard. It is important to mention that the random keyboard and text pairs are not generated at the time of the test. Instead, the eight shuffled keyboards and their associated random text phrases are generated and stored in the system before the experiment. When a subject starts the experiment, the system pulls in sequence the pre-stored shuffled keyboards and random text. Therefore, all participants are exposed to the same eight shuffled keyboards (in addition to the two QWERTY ones), and in the same sequence of display.

For the typing errors, the system is designed to handle typing erros in three different scenarios:

1. The participant notices the typing error, and uses *backspace* to correct. In this scenario, the first *backspace* click data is recorded as any other key. All the consecutive *backspace* clicks (if any) are ignored.

2. The participant does not observe up to two typing errors. The system is designed to accept that small number of typing errors. This allows participants to be focused on the next letter they need to search for, and not to spend time confirming each clicked letter. Recall that the experiment is mainly about how letters are being typed.

3. The participant, for any reason, makes more than two errors. The system is designed to alert the participant in the middle of the session to review and make at least one correction, then proceed for the remaining characters.

There was a time delay between sessions set to 45 seconds. This time was designed to help participants release their focus and attention before they started the next sessions. During that time delay, participants were encouraged to perform some short relaxing actions such as change the gaze focus to other points of interest in the room, move shoulders, or walk a few steps in the room. Depending on the participant's skills and capabilities, the experiment took between 10 and 20 minutes to complete.

6.2 Apparatus

All participants conducted the experiment using the same equipment. The application was designed as a JAVA SWING desktop application, developed using SDK 1.6 running on Windows XP SP2 OS. The hardware used was Dell VOSTRO 200, 1.6GHz, and 2 GB RAM. Display was a Dell 19" Flat Screen set to 1600x1200 pixels, where the keyboard image was set to 800x200 pixels.

6.3 Subjects and Test Sessions

Over a period of 12 weeks, sessions from 274 users were collected, at 10 sessions per user, for a total of 2740 sessions. These sessions are divided into 2192 shuffled keyboard sessions and 548 QWERTY keyboard sessions. The general purpose of the study was explained at a high level to the participants; they were informed that this experiment was designed to prove a certain hypothesis related to their visual capabilities. However, participants were not informed about the details of the features, and hence had no prior knowledge of the type of behavioral data being collected. They possessed the following qualities amongst them:

- 55% males and 45% females.
- Right- and left-handed users.
- Age range from 17 to 40 years old.
- Different computer skills from light users to professionals.
- English as first and second language diversity.
- All subjects had normal, or corrected to normal, vision range.

Participants were scheduled for a 30-minutes time slot on a voluntary basis for the majority, and partial course credit for the remaining participants. This time included a short demo, experiment explanation, and the actual testing.

6.4 Evaluation Based on Neural Network Model

6.4.1 Enrollment Details

In Section 5.1.2, we explained our enrollment strategy. For the actual implementation, our approach consisted of using a five-fold cross validation strategy. To this end, we divided our user population into five subsets, each involving 20% of the users. We then used one of the five subsets as a legal-user database and built a profile for each of these known users. Testing each profile against *self* sessions, and against *non-self* sessions of other users was done differently. For the 20% legal users, one profile was tested against all the users' sessions that were not used to build user profiles. For the other 80% imposter users, the profile was tested against all the users' sessions. The same process was repeated five times, by switching in each new round the legal-users subset with one of the imposters subsets from the previous round.

We decided that 70% of the sessions from *self* and *non-self*, selected randomly, would

be used to build a profile for a genuine user. For each user, the random selection of sessions included six shuffled keyboard sessions and one QWERTY keyboard session. The remaining 30% of the sessions would be used for evaluation purposes. Similarly, for each user, evaluation sessions included two shuffled keyboard sessions and one QWERTY keyboard session. Recall that participants were each asked to perform 10 sessions (with 8 shuffled keyboard sessions and 2 QWERTY keyboard sessions per subject) (Section 6.1). Seven of these sessions were then used for enrollment; the remaining sessions were kept for verification.

It is important to stress that in the evaluation phase, the neural network has no previous knowledge of any of the used *self* sessions, and no previous knowledge of any of the users whose sessions are used as *non-self*.

6.4.2 Performance Calculation

The performance of the system is expressed mainly in terms of *FAR* and *FRR*. As explained in Section 6.4.1, and as will be shown in Section 7.1.1, the evaluation is done using five-fold cross validation. Therefore, for each user, five testing rounds are performed before final *FAR* and *FRR* values can be produced. For each round, sessions of one user marked as *self* and sessions of other users (in the combination explained in Section 6.4.1) marked as *non-self*, are fed to the network. Based on the threshold, a confidence value is rounded to either 1 or 0. Each rounded value is compared against its producing session. Based on the comparison, one of the following actions is performed:

1. Output matched a *self* session type: Increment TS_COUNT
2. Output matched a *non-self* session type: Increment TNS_COUNT
3. Output is rounded to 0, while session is *self*: Increment FR_COUNT by 1.

4. Output is rounded to 1, while session is *non-self*: Increment FA_COUNT by 1.

Where *TS_COUNT* and *TNS_COUNT* stand for *True_Self_Count* and *True_Non-Self_Count* respectively.

After all sessions of this round for one user are completed, an average *FAR* and *FRR* of this round for that user are computed as shown in Equations 6.1 and 6.2 respectively.

$$FAR_{user} = \frac{FA\_COUNT}{TS\_COUNT + FA\_COUNT} \times 100 \tag{6.1}$$

$$FRR_{user} = \frac{FR\_COUNT}{TNS\_COUNT + FR\_COUNT} \times 100 \tag{6.2}$$

The above calculations are repeated for all the k users in the 20% genuine user subset. Therefore, the average *FAR* and *FRR* of that round are shown in Equations 6.3 and 6.4 repetitively.

$$FAR_{round} = \frac{\sum_{usr=1}^{k} FAR_{usr}}{k} \tag{6.3}$$

$$FRR_{round} = \frac{\sum_{usr=1}^{k} FRR_{usr}}{k} \tag{6.4}$$

Finally, the final *FAR* and *FRR* are computed as the average of all the FAR_{round} and FRR_{round} values computed from the 5 rounds, as shown in 6.5 and 6.6 respectively.

$$FAR_{final} = \frac{\sum_{round=1}^{5} FAR_{round}}{5} \tag{6.5}$$

$$FRR_{final} = \frac{\sum_{round=1}^{5} FRR_{round}}{5} \tag{6.6}$$

6.5 Evaluation Based on Computational Statistics Model

6.5.1 Enrollment Details

As explained in Section 5.2.3, enrollment sessions of each user are divided randomly into two (training/testing) groups. However, each division includes one of the two QWERTY keyboard sessions of each user. Similar to the neural network model, we decided to use 70% of the sessions randomly selected from each user's sessions (*self data*) for configuring the decision rule (training); the remaining 30% of the sessions are kept for evaluation purposes. Therefore, configuring the decision rule function of each feature is done using 7 feature data points. Additionally, a total of 63 points ($9\ features \times 7\ feature\ data\ values$) are used to build each user's statistical model.

6.5.2 Performance Calculation

Similar to the neural network model, performance is also expressed in terms of *FAR* and *FRR*. For each user, decision rules are fed in parallel with features data vectors. Each feature data vector is composed of the reserved testing *self data* of the user, and *non-self data* of all sessions of all the remaining users. Produced normalized scores from all the nine decision rule functions are fused using the fusion techniques presented in Section 5.2.1. Based on a predefined threshold, the confidence value output is rounded to either 1 or 0. Rounded values are finally compared against their producing session's owner; based on the comparison result, actions similar to the ones mentioned in Section 6.4.2 are carried out.

The *FAR* and *FRR* performance criteria are computed using Equations 6.1 and 6.2 respectively.

The above process is repeated for all users, and the final *FAR* and *FRR* are computed

as the average of all FAR_{user} and FRR_{user} values, as shown in Equations 6.7 and 6.8 respectively.

$$FAR_{final} = \frac{\sum_{user=1}^{N} FAR_{user}}{N} \tag{6.7}$$

$$FRR_{final} = \frac{\sum_{user=1}^{N} FRR_{user}}{N} \tag{6.8}$$

6.6 Summary

In this chapter, we presented in details the enrollment procedures for both neural network and computational statistics models. For the performance, we detailed the equations that will be used in calculating the performance for each model. These equations will be used in the next chapter to provide and discuss the experimental analysis results.

Chapter 7

Experimental Analysis and Results

In the previous two chapters, we explained our biometric analysis technique and performance evaluation methods. This chapter computes the system performance for user authentication and user identification, based on both neural networks and computational statistics models. Additionally, special experiments were conducted and are presented in this chapter to study certain variations of the system.

7.1 User Verification Results Using Neural Network Model

7.1.1 User Authentication

To evaluate the proposed system performance in user verification, we repeated the same approach applied in the enrollment phase (Section 6.4.1). The difference is that the remaining 30% of each user's sessions were used. *FAR* and *FRR* results from cross validation rounds were finally averaged, and are presented here. In this section, results of the performance evaluation are presented. User verification performance was evaluated using the three proposed neural network models (Section 5.1.2). For the three models, verification was repeated to measure performance at different thresholds. Starting at 20%, and at 10% increment, the user verification test was

repeated eight times for all the users. Table 7.1 shows the achieved *FAR* and *FRR* at each threshold. As expected, the lowest *FAR* of 0.52% was achieved at 90% threshold using the *Divide* & *Select* model, while the lowest *FRR* of 0.0% was achieved at 20% threshold using the *Divide* & *Fuse* model. These results were achieved based on the successfully enrolled user group. Another sample analysis was done using the *Feed All* model that included all users[1] (i.e., users whom the neural network could not successfully train using their samples). Table 7.3 shows the achieved *FAR* and *FRR* at each threshold. Comparing this table and the *Full Feed* result columns of Table 7.1 shows the impact of the overall performance degradation as a result of the bad samples.

Table 7.1: Performance of the Three Neural Network Models

Threshold	***Full Feed***		***Divide & Fuse***		***Divide & Select***	
	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**
20%	**30.00**	**0.43**	**34.05**	**0.0**	**37.33**	**0.21**
30%	**22.40**	**0.66**	**28.41**	**1.47**	**25.02**	**0.98**
40%	**15.60**	**4.07**	**17.15**	**3.83**	**14.25**	**2.27**
50%	**8.12**	**8.50**	**6.77**	**5.22**	**6.01**	**4.37**
60%	**6.90**	**9.20**	**4.93**	**7.04**	**4.11**	**6.53**
70%	**4.24**	**18.63**	**3.06**	**13.92**	**2.73**	**8.04**
80%	**2.88**	**29.13**	**1.38**	**23.3**	**1.07**	**12.84**
90%	**1.71**	**48.67**	**0.79**	**35.04**	**0.52**	**26.14**

[1]In this experiment, the rate of users that *failed to enroll* (FTER) is 9%

Table 7.2: Performance Results Obtained for the Three Neural Network Models When Varying the Threshold. *Feed All* model achieved EER of 8.3%. *Divide & Fuse* model achieved EER of 5.98%. *Divide & Select* achieved EER of 5.7%

Threshold	***Full Feed***		***Divide & Fuse***		***Divide & Select***	
	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**
20.0%	**30.00**	**0.43**	**34.05**	**0.0**	**37.33**	**0.21**
30.0%	**22.40**	**0.66**	**28.41**	**1.47**	**25.02**	**0.98**
49.7%	**8.30**	**8.31**	**7.08**	**4.95**	**6.71**	**4.16**
51.6%	**7.94**	**8.62**	**6.23**	**5.68**	**5.69**	**5.70**
54.3%	**7.46**	**8.85**	**5.98**	**5.97**	**4.11**	**6.11**
70.0%	**4.24**	**18.63**	**3.06**	**13.92**	**2.73**	**8.04**
80.0%	**2.88**	**29.13**	**1.38**	**23.3**	**1.07**	**12.84**
90.0%	**1.71**	**48.67**	**0.79**	**35.04**	**0.52**	**26.14**

Table 7.3: Impact of *FTE* on Evaluation Performance at Different Thresholds for the *Full Feed* Neural Network Technique

Threshold (%)	FAR (%)	FRR (%)
20	39.80	2.42
30	36.92	5.55
40	28.13	7.16
50	23.84	9.11
60	18.34	12.47
70	11.27	21.60
80	10.0	34.87
90	4.31	56.18

The above performance results achieved are also presented by plotting ROC curves (Figure 7.1). For the *Feed All* technique, we achieved an *EER* of 8.3% at a

threshold of 49.7%. For the *Divide & Fuse* model an *EER* of 5.98% was achieved at a threshold of 54.25%. For the *Divide & Select* model, an *EER* of 5.7% was achieved at a threshold of 51.6%. *EER* for the three neural network models is presented in Table 7.2.

Comparison between the results from the three techniques shows that performance using the latter two techniques has superiority over the *Feed All* technique. This indicates that individual features have different classification power. It also shows that experimental fusion techniques after segregating the features could lead to better performance. It would thus be beneficial for further neural network models to be introduced and tested for better performance.

7.1.2 User Identification

Evaluation of the proposed system performance in user identification is very similar to the user authentication approach used in 5.1.4. the three neural network models (*Full Feed*, *Divide & Fuse* and *Divide & Select*) are used in the evaluation. In this evaluation we further divide the population by taking out 10% (27 users) and marking them as unknown users. Then, we retrain the neural networks models and build user profile repository using only the 247 known users. Additionally, we add another label for the data as unknown user, used to refer to any user data sessions the network is not able to identify during testing. This is a result of the neural networks model producing a CV less than the threshold value.

We build test cases composed of test sessions of known users and all sessions of unknown users. Each test session (marked as anonymous) is fed to the neural network model that is already preloaded with the master template of one user. The produced *Confidence Value* is then added to the output vector. The neural network repeats this process by sequentially loading all master templates of all users in the repository.

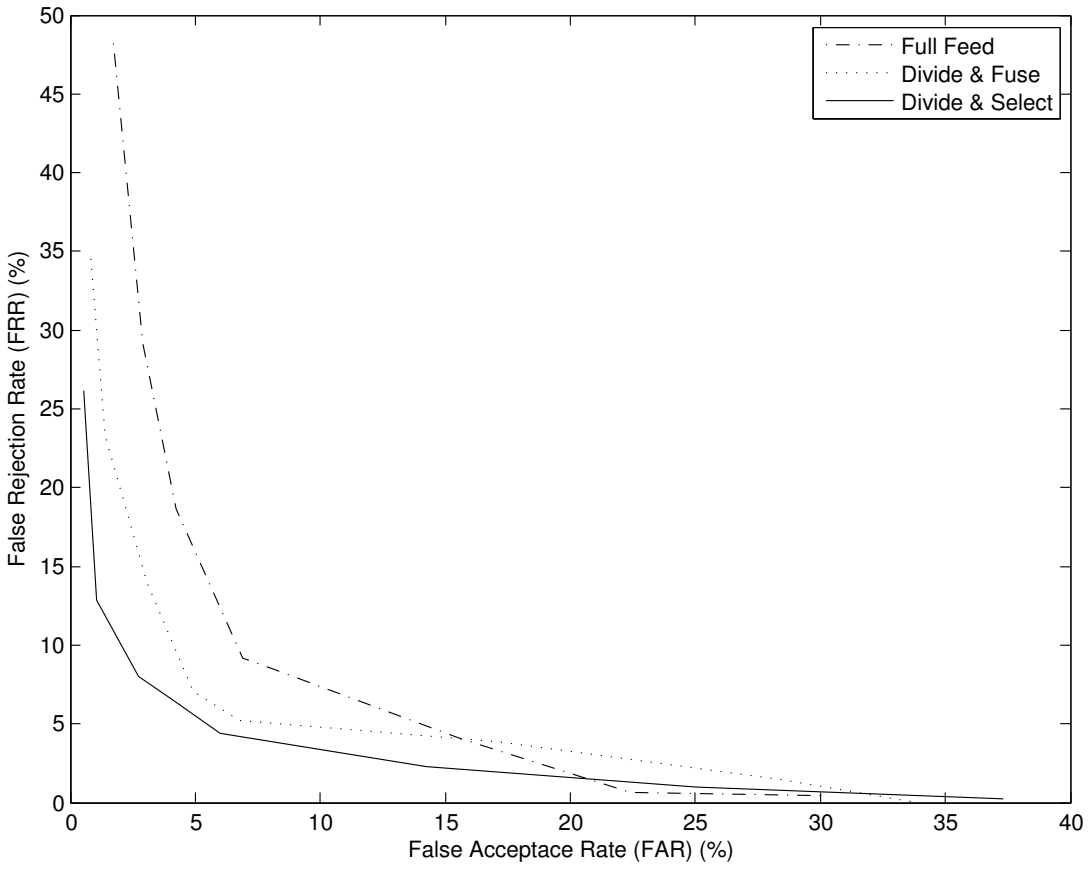


Figure 7.1: ROC Curves for Neural Network Verification Results. At different threshold values, *FAR* and *FRR* are plotted for the three neural network models. An *EER* of 5.7% is achieved using the *Divide* & *Select* model, which is the lowest among all three models.

In this experiment, performance was measured at different threshold values starting from 20% at 10% increments. 30% of the sessions of each known user in the population, and 100% of the sessions of each unknown user were used in this experiment as the anonymous sessions. For the purpose of this experiment, we further define the following terms:

Correct Classification Rate (CCR) Defined as the rate of test results that correctly predicted the user identity.

False Match Rate (FMR) Defined as the rate of test results that predicted a user identity from the users list that is different than the actual identity of the test sessions.

False Non-Match Rate (FNMR) Defined as the rate of test results that predicted a known user identity as unknown user.

Misclassification Rate (MCR) Defined as the sum of *False Match Rate (FMR)* and *False Non-Match Rate (FNMR)*.

The *FMR* in a way is analogous to the *FAR* in the user authentication. Not only it fails to determine the identity of the actual user, but it also wrongly "accuses" another known user to be the owner of the test sessions. In a forensic investigation, this could result in wrongly convicting and sending an innocent man to jail. As well, *FNMR* is analogous to the *FRR* in the user authentication; in user authentication, *FRR* means a legitimate user is denied access, while in user identification *FNMR* means failure to catch an alleged criminal. We also note that the sum of *CCR* and *MCR* is equal to 100%. Figure 7.2 illustrates by example all the possible outcomes of the identification test, and their corresponding classifications.

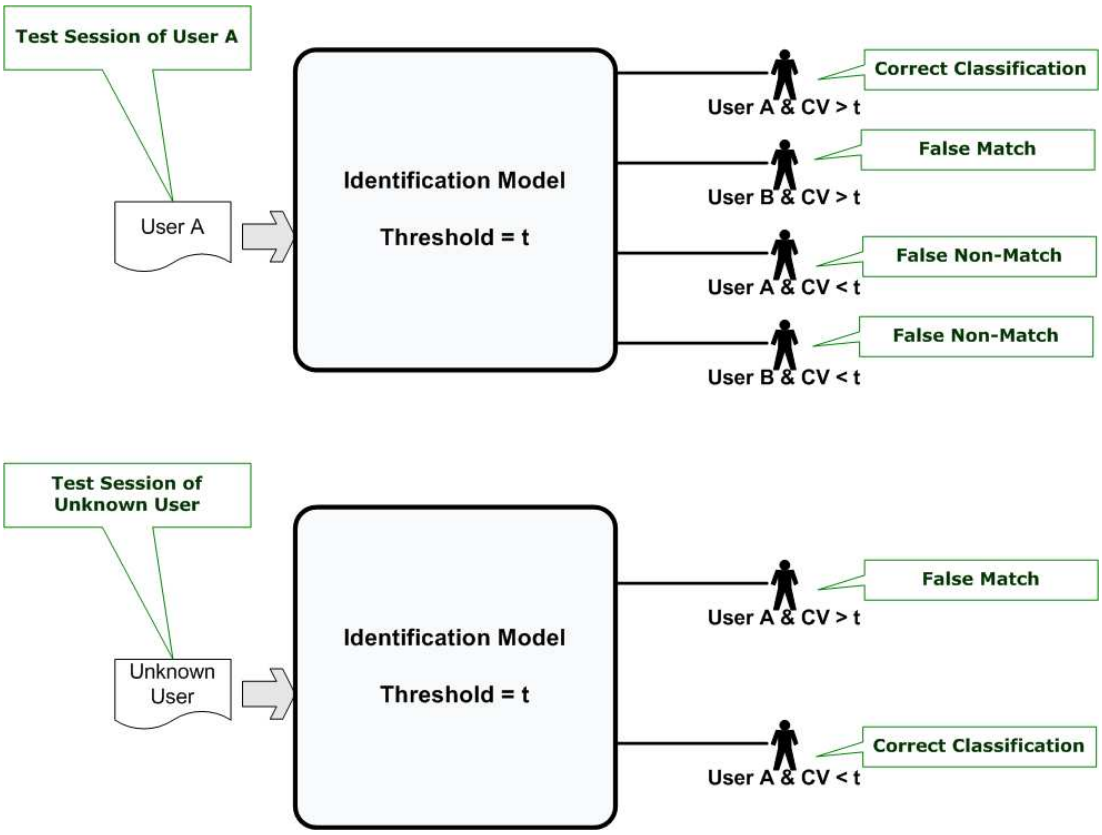


Figure 7.2: User Identification Scenarios by Example. Two sample input sessions, one for know and the other for unknown users. The output possible identifications are classified as either *Correct Classification* (CC), *False Match* (FM) or *False Non-Match* (FNM).

Table 7.4: Identification Performance for the Three Neural Network Models. *Feed All* model achieved *FMR* of 41.2%. *Divide & Fuse* model achieved *FMR* of 35.0%. *Divide & Select* achieved *FMR* of 31.4% all at threshold of 90%

Threshold	***Full Feed***		***Divide & Fuse***		***Divide & Select***	
	***FMR* (%)**	***FNMR* (%)**	***FMR* (%)**	***FNMR* (%)**	***FMR* (%)**	***FNMR* (%)**
20%	**70.1**	**0.00**	**68.7**	**0.00**	**66.5**	**0.00**
30%	**67.0**	**0.00**	**64.7**	**0.00**	**63.7**	**0.00**
40%	**64.8**	**0.00**	**60.2**	**0.00**	**61.6**	**0.00**
50%	**59.9**	**0.00**	**57.3**	**0.00**	**60.0**	**0.00**
60%	**57.3**	**0.05**	**53.7**	**0.01**	**53.9**	**0.09**
70%	**54.5**	**0.80**	**49.0**	**0.72**	**45.2**	**0.86**
80%	**48.2**	**3.74**	**44.8**	**1.56**	**37.8**	**1.64**
90%	**41.2**	**5.18**	**35.0**	**2.07**	**31.4**	**2.15**

Table 7.4 shows the achieved *FMR* and *FNMR* for the three network models at each threshold. The *Full Feed* neural network model achieved an *FMR* and *FNMR* of 41.2% and 5.18% respectively at 90% threshold. The *Divide* & *Fuse* neural network model achieved an *FMR* and *FNMR* of 35.0% and 2.07% respectively at 90% threshold. The *Divide* & *Select* neural network model achieved an *FMR* and *FNMR* of 30.4% and 2.15% respectively at 90% threshold. As the threshold ratio increased, the *FMR* decreased. This is a logical result because increasing the threshold ratio in the neural network model means higher resemblance between the anonymous session and the user profiles. However, this comes at the price of increasing the *FNMR*; if the anonymous session is not very close to the genuine user's profile, and at the same time is different from all other signatures, the system will fail to identify any user as a match. Additionally, using the feature segregation models showed a higher user identification performance. The *Divide* & *Select* model outperformed the *Full Feed* model by 23%. This suggests that although the *FMR* values produced are relatively high, further analysis could lead to better classification models.

The highly error rates obtained for identification compared to authentication is not surprising since identification is known to be a more challenging and error-prove process.

7.2 User Verification Results Using Computational Statistics Model

7.2.1 User Authentication

Recall that enrollment of users using statistical model approach required only 70% of the subject's test session group (*self*). Therefore, for each user, we will evaluate the system performance using that user's remaining self sessions and *non-self* data of all sessions of all other users. Since the experiment population is 274 users, each testing feature vector will contain 2733 data points (3 *self* data points and 2730 *non-self* data points).

In this section we present user verification performance based on the *Simple-Sum (SS)* and *Weighted-Sum (WS)* feature fusion techniques explained in Section 5.2.1. Starting at a threshold of 1 and with an increment of 1 (0.5 when *Triangle Rule* is applied), the user verification test was repeated for all data points in the testing feature vector of each user. After this verification test was repeated for all the population, results were averaged to produce the achieved *FAR* and *FRR* at each threshold.

Table 7.5 shows that using the *Step Rule*, the lowest *FAR* achieved was 0.039% using the *Weighted-Sum (WS)* fusion technique at a threshold of 9. The lowest *FRR* achieved was 0.021% using the *Simple Sum (SS)* fusion technique at a threshold of 1. Table 7.6 shows that applying the *Triangle Rule* instead of the *Step Rule* significantly improved the *FAR*; it shows the lowest *FAR* of 0.0% at a threshold of 6 using the *Weighted-Sum (WS)* fusion technique. The lowest *FRR* achieved using the *Triangle Rule* is shown to be 0.034% at a threshold of 1 using the *Weighted-Sum (WS)* fusion technique.

Table 7.5: Performance Results for the Statistical Model Using Step Rule

Threshold	***Simple-Sum***		***Weighted-Sum***	
	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**
1	**97.481**	**0.021**	**93.117**	**0.052**
2	**88.593**	**0.119**	**82.544**	**0.175**
3	**70.390**	**0.328**	**63.712**	**0.455**
4	**46.545**	**0.805**	**42.756**	**0.993**
5	**24.564**	**1.211**	**23.593**	**1.333**
6	**9.896**	**1.621**	**10.349**	**1.710**
7	**2.915**	**2.476**	**3.458**	**2.567**
8	**0.573**	**2.919**	**0.068**	**3.244**
9	**0.054**	**4.786**	**0.039**	**3.552**

Table 7.6: Performance Results for the Statistical Model Using Triangle Rule

Threshold	***Simple-Sum***		***Weighted-Sum***	
	***FAR* (%)**	***FRR* (%)**	***FAR* (%)**	***FRR* (%)**
0.5	**90.973**	**0.062**	**87.473**	**0.034**
1.0	**76.231**	**0.236**	**77.829**	**0.098**
1.5	**57.873**	**0.597**	**50.991**	**0.379**
2.0	**38.646**	**1.165**	**35.319**	**0.938**
2.5	**22.937**	**1.951**	**27.154**	**1.448**
3.0	**11.911**	**2.155**	**14.230**	**1.849**
3.5	**5.521**	**2.268**	**3.211**	**2.102**
4.0	**2.236**	**2.840**	**1.609**	**2.501**
4.5	**0.783**	**3.204**	**0.848**	**3.108**
5.0	**0.244**	**3.468**	**0.173**	**3.690**
5.5	**0.060**	**4.190**	**0.033**	**4.082**
6.0	**0.014**	**4.857**	**0.008**	**4.427**
6.5	**0.001**	**4.963**	**0.000**	**4.784**
7.0	**0.000**	**5.014**	**0.000**	**4.924**
7.5	**0.000**	**5.029**	**0.000**	**4.945**
8.0	**0.000**	**5.039**	**0.000**	**4.982**
8.5	**0.000**	**5.039**	**0.000**	**4.982**
9.0	**0.000**	**5.039**	**0.000**	**4.982**

Table 7.7: EER Highlighted for the Computational Statistics Models

	Simple-Sum				***Weighted-Sum***			
Threshold	***Step Rule***		***Triangle Rule***		***Step Rule***		***Triangle Rule***	
	FAR	***FRR***	***FAR***	***FRR***	***FAR***	***FRR***	***FAR***	***FRR***
7.15	**2.56**	**2.54**	**5.03**	**2.35**	**2.95**	**2.67**	**2.97**	**2.06**
7.2	**2.45**	**2.56**	**4.86**	**2.38**	**2.78**	**2.70**	**2.89**	**2.08**
7.55	**1.67**	**2.72**	**3.71**	**2.58**	**1.59**	**2.94**	**2.11**	**2.12**
7.85	**0.92**	**2.85**	**2.73**	**2.75**	**0.58**	**3.14**	**1.85**	**2.44**

The above performance results are illustrated using ROC curves (Figures 7.3 and 7.4). We obtain an *EER* of 2.5% for the *Simple Sum (SS)* fusion technique when the *Step Rule* is applied. As expected from previous results, a lower *EER* of 2.1% was achieved using the *Weighted-Sum (WS)* fusion technique and applying the *Triangle Rule*. *EER* achieved from the different computational statistics models are presented in Table 7.7. This result variation based on models used matches the same analogy of feature segregation discussed in neural network models. When features were given different voting power based on their effectiveness, we were able to achieve a lower *EER*.

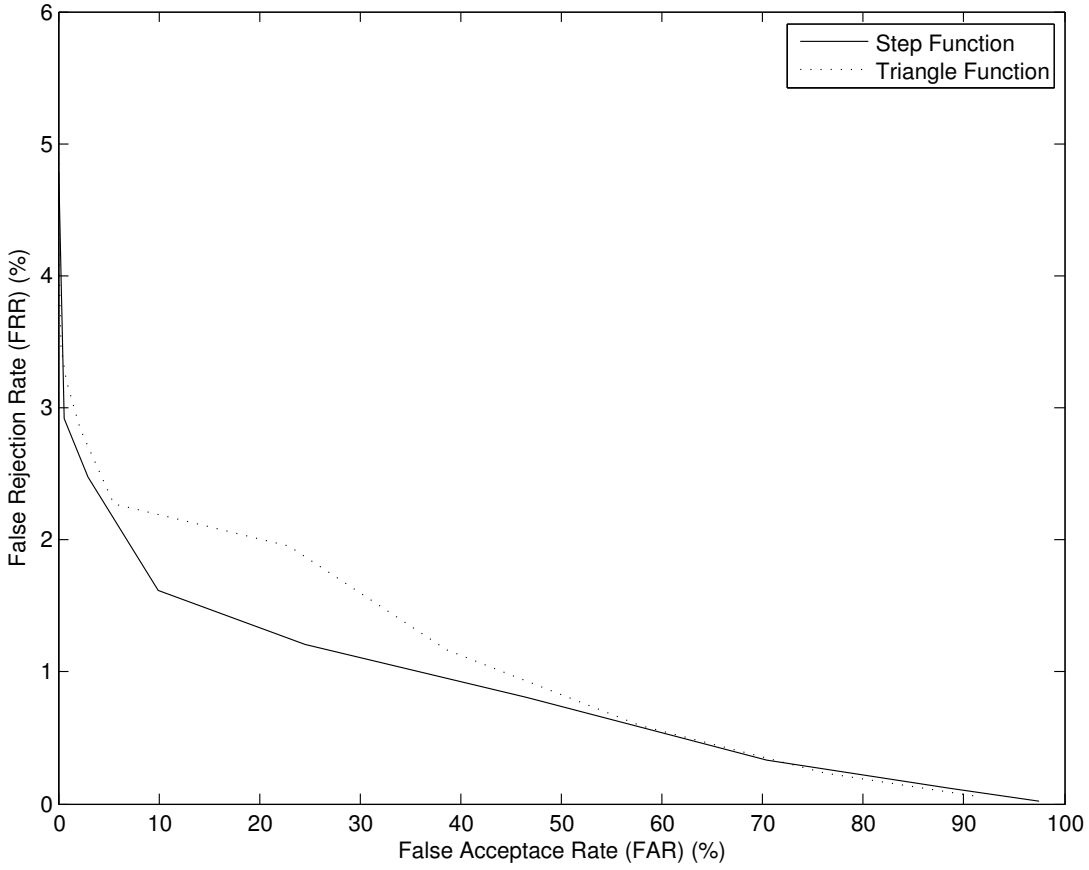


Figure 7.3: ROC Curve for Statistics Model Using *Simple Sum* Technique. At different threshold values, *FAR* and *FRR* are plotted. An *EER* of 2.5% is achieved using the *Step Rule*

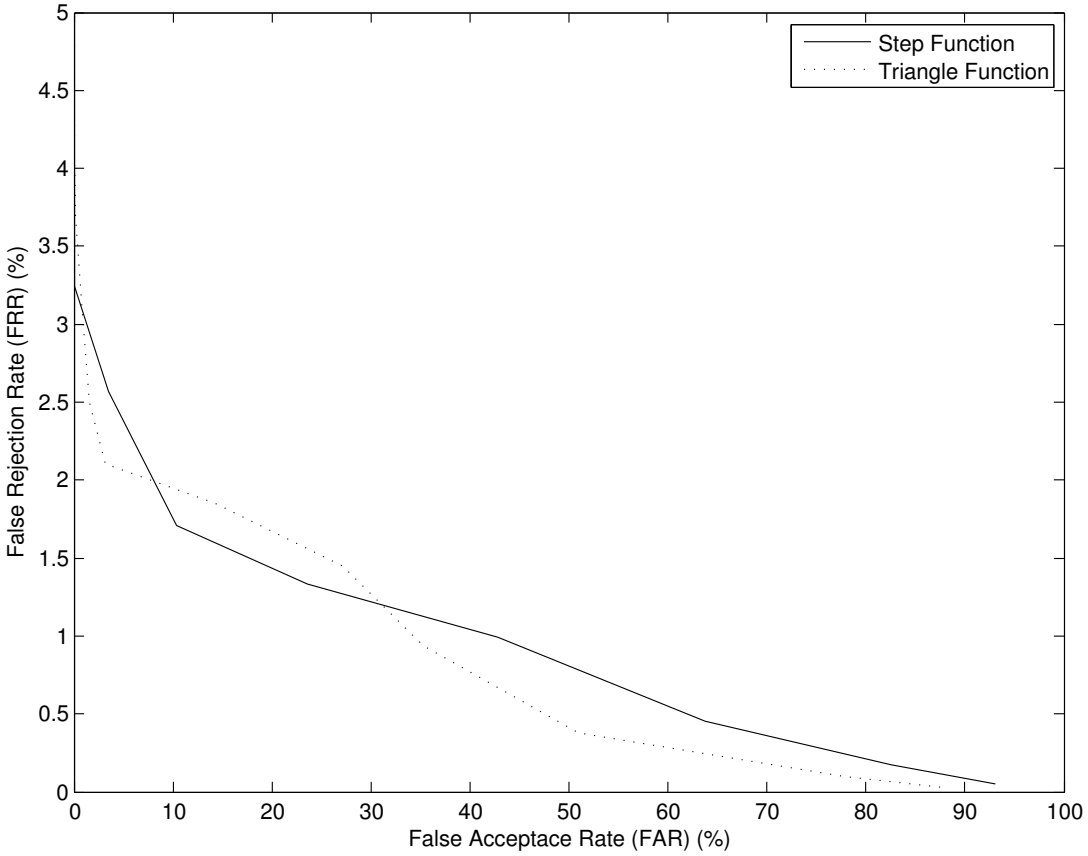


Figure 7.4: ROC Curve for Statistics Model Using *Weighted-Sum* Technique. At different threshold values, *FAR* and *FRR* are plotted. An *EER* of 2.1% is achieved using the *Triangle Rule*

7.2.2 User Identification

Evaluating the system performance in the user identification based on the computational statistics model follows the same approach used in 7.1.2. However, *CVs* are produced by linearly fusing scores of matched features (integer values). Therefore, there is a high possibility that more than one identity prediction emerge, especially when the *Simple Sum* model is used. For such cases, we decided to use the sum of Euclidean Distances of each session. The user with the lesser Euclidean Distances sum is then chosen.

In this section we present user identification performance based on the *Simple-Sum (SS)* and *Weighted-Sum (WS)* feature fusion techniques, using the *Triangle Rule* and a threshold range of 1 to 9 with an increment of 1. *False Match Rate (FMR)* and *False Non-Match Rate (FNMR)* results are tabulated and presented in Table 7.8.

Table 7.8: Identification Performance for Computational Statistics Models. *Simple-Sum* model achieved *FMR* of 32.17% and *FNMR* of 2.71%. *Weighted-Sum* model achieved *FMR* of 26.08% and *FNMR* of 1.83%

Threshold	***Simple Sum***		***Weighted Sum***	
	***FMR* (%)**	***FNMR* (%)**	***FMR* (%)**	***FNMR* (%)**
1	**92.13**	**0.00**	**88.45**	**0.00**
2	**86.14**	**0.00**	**81.64**	**0.00**
3	**79.08**	**0.00**	**76.62**	**0.00**
4	**70.27**	**0.00**	**68.40**	**0.00**
5	**67.53**	**0.00**	**65.38**	**0.00**
6	**61.07**	**0.08**	**60.61**	**0.00**
7	**57.12**	**0.12**	**55.04**	**0.06**
8	**48.26**	**1.05**	**37.27**	**0.48**
9	**32.17**	**2.71**	**26.08**	**1.83**

The *Simple Sum* computational statistics model achieved an *FMR* and *FNMR* of 32.17% and 2.71%, respectively, at a threshold of 9. The *Weighted-Sum* computational statistics model achieved an *FMR* and *FNMR* of 26.08% and 1.83%, respectively, at a threshold of 9. We note that results from the computational statistics models are following the observations noted from the neural network models. For example, as the threshold ratio increased, the *FMR* decreased. However, we note that *FMR* and *FNMR* results from the computational statistics models are better than the ones from the neural networks models by 17% and 12% respectively. One explanation for that improvement could be due to the higher classification accuracy of the computational statistics models, as mentioned earlier.

7.3 Additional Testing

7.3.1 Biometric Factors Separate Evaluation

In Section 4.2, we explained in details each of the three biometric factors our proposed system is composed of. Table 4.3 marked each biometric feature as belonging to one or more of the three biometric factors. However, later in the research, all features were fused using the different explained models. In this experiment, we study the performance of each biometric factor separately using its corresponding feature list. From Table 4.3 we note that features are to an extent equally distributed over the biometric factors. Three features belong to the mouse dynamics factor, two features belong to the *Visual Scan & Detection* factor and two features belong to the *Short-Term Memory* factor. Additionally, the last two features are common between the *Visual Scan & Detection* and the *Short-Term Memory* factors. For the latter two features, we will use them with both factors separately. For our experiment, we decided to use the *Weighted-Sum* computational statistics model based on the

Triangle Rule classifier. This model showed the highest accuracy amongst all other tested models. Additionally, the threshold for each factor is set to be equal to the number of features included in that factor's group (3 for mouse dynamics and 4 for the *Visual Scan & Detection* and the *Short-Term Memory*). Table 7.9 shows the *FAR* and *FRR* produced from testing the three biometric factors. The Mouse Dynamics factor achieved an *FAR* and *FRR* of 44.2% and 23.9%, respectively. The *Visual Scan & Detection* factor achieved *FAR* and *FRR* of 25.8% and 11.3%, respectively. The *Short-Term Memory*) achieved *FAR* and *FRR* of 31.0% and 12.7%, respectively.

Table 7.9: Performance Results for Individual Biometric Factors. Threshold is set to 3 for the Mouse Dynamics, 4 for both the *Visual Scan & Detection* and the *Short Term Memory*

Biometric Factor	***FAR* (%)**	***FRR* (%)**
Mouse Dynamics	**44.2**	**23.9**
Visual Scan & Detection	**25.8**	**11.3**
Short Term Memory	**31.0**	**12.7**

This experiment was very useful in giving us an in-depth view of the system to determine strength and weaknesses. The first observation is that Mouse Dynamics performance was the lowest amongst the three biometric factor. One explanation is that the Mouse Dynamics factor used one less feature, which affected the classification capability of the system. Additionally, it could mean that more Mouse Dynamics features need to be explored and used to enhance the accuracy of that factor, and hence the overall performance. And also not surprisingly, more mouse data might be required for accurate identification. The *Visual Scan & Detection* factor achieved a relatively better performance than the other two factors. This is an indication that preliminary work done to extract features was more successful for that factor than

the others. It also tells that there is a potential for further success in that cognitive factor with advanced research.

7.3.2 Individual Biometric Feature Impact

To further study the effect of each biometric feature, a special analysis was made to measure the *FAR* and *FRR* based on the individual biometric factor. Another preliminary analysis was done in an early stage of this research to produce the feature golden set used throughout the experiments (Section 4.6). This special test is crucial in giving an in-depth understanding of the impact of each feature, the matter that can help enhancing the system performance in future work.

For this special analysis, we designed a simple statistical classifier model that uses only one factor data value. User sessions were also divided into 70% for building profile, and 30% for testing. Table 7.10 shows the results produced from the experiment.

Table 7.10: Individual Biometric Feature Performance

Feature	*FAR* (%)	*FRR* (%)
W-n-F	68.2	9.9
FT	27.9	17.6
FT Q2S	23.9	35.1
FT OCC	19.2	28.8
DST Q2S	62.5	13.2
DST OCC	69.0	11.3
X2Y	43.9	17.5
Speed	50.7	21.3
CT	77.2	2.4

From that table, we note that *Fly Time* related features achieved the lowest *FAR*. This means this cognitive feature has high discriminative potential. We also note that

the *Distance* related features achieved the lowest *FRR*. This indicates that data values of these features are relatively more stable for each user than other features. The *CT* feature scores the lowest *FRR*. However, looking at the highest *FAR* scored (77.2%), it indicates that this feature is relatively weak in terms of discrimination between users.

7.3.3 Random vs. Comprehensible Phrases

This special test was designed to study the effect (if any) when the random phrase included some comprehensible words or phrases (for example, vancouver/2has/views). A special analysis was made to compare results using this type of comprehensible phrase and the other complete random ones. Results showed that the overall time was slightly enhanced, as subjects did not have to look at each character of the string. Nevertheless, feature data values and ratios were the same among all sessions. This is mainly because, regardless of how easy or hard the phrase is, the main focus and effort is spent by users trying to locate the keys on the virtual keyboard. Since a new shuffled keyboard is generated every session, there is no significant difference between a new random string, and a known one. To show a sample result, we repeated the experiment explained in Section 7.3.2 but using only sessions that had comprehendible phrases. Table 7.11 shows the achieved results.

Table 7.11: Study of Comprehendible Text Performance Impact. This test measures the performance when the input texts in the test sessions consists of comprehendible phrases

Feature	*FAR* (%)	*FRR* (%)
W-n-F	62.4	11.3
FT	23.0	21.3
FT Q2S	19.8	37.3
FT OCC	20.7	26.5
DST Q2S	66.1	18.3
DST OCC	74.4	17.8
X2Y	47.2	15.4
Speed	56.0	26.3
CT	76.9	3.6

7.4 Summary

In this chapter we thoroughly tested the system using different models and techniques to show the validity of the hypothesis presented in the previous chapters. Testing using different models revealed a substantial difference in the performance for both user verification and user identification. This indicates that performance is impacted by the used models, which means that finding better analytical models could lead to better performance.

Among the neural networks models, the *Divide* & *Select* model achieved the lowest *EER* of 5.7% for user authentication at 51.6% threshold and the lowest *FMR* of 31.4% at 90% threshold. Among the computational statistics models, the *Weighted-Sum* model achieved the lowest *EER* of 2.1% at 7.55 threshold, and the lowest *FMR* of 26.08% at threshold of 9.

Chapter 8

Conclusion

Human factors is the science of understanding the properties of human capabilities and characteristics. Using this knowledge to engineer and design products, processes, and systems has been a key success over the past decades. Our research uses the same foundation as human factors science. In fact, the novelty of our research added to the goals another considerable one: *to increase the human digital security.* By manipulating the selected cognitive factors on a discriminative rather than a collective basis, human behaviors related to these factors were successful in identifying individual humans.

8.1 Work Summary

The first major contribution of this research is that human visual scan and detection, long studied by medicine and psychology researchers, is modeled, quantified, and measured as a biometric factor. When users are subject to an abnormal visual distortion experiment, quantified visual detection parameters of users show unique and consistent patterns for each user. These two measurements qualify the visual

scan and detection as a biometric factor.

The second contribution of this research is the use of this behavioral biometric factor in static user authentication. Previous work on mouse biometrics has focused primarily on continuous authentication, mainly because the data capture requires a minimum amount of data for decision making. In this research we established a relatively short data sequence that may be used as one-time password (OTP), achieving encouraging performance in static authentication. An OTP has the advantage of being used once which means better security, without greatly affecting the user experience.

The third contribution of this research is to build confidence in this newly developed system. First, a mass enrollment of users was done; a considerable achievement in this research was the successful enrollment of 274 users who volunteered to produce a total of 2740 biometric sessions. Additionally several models were used in this analysis that are either based on neural networks, or on computational statistics.

Performance was primarily measured based on *FAR*, *FRR*, *EER* and *MR*. For user authentication, using the *Divide & Select* neural network model, an *EER* of 5.7% was achieved. This performance was enhanced to be 2.1% using the *Weighted-Sum* statistical model. For user identification an *MR* of 29.1% was achieved using the *Divide & Select* neural network model. Again, this rate was enhanced to be 26.08% using the *Weighted-Sum* statistical model.

8.2 Application

The experimental approach in Chapter 5 used a relatively reasonable password length in each testing session. Most of the text phrases used throughout the test sessions are completely of random structure. These phrases can be used as one-time password

(OTP). In real-life applications OTP, which is generated on the fly, does not require any memorization effort. Our proposed OTPs are more effective than traditional memorized passwords for the following reasons:

1. Resilient to dictionary attacks and other attacks that have shown effectiveness with traditional passwords.

2. No storage is required, and so cannot be stolen.

3. Since no memorization is required, OTP could use relatively longer character strings than a traditional fixed password. This gives an additional stiffness to this type of password. It also enhances the system performance without negatively affecting the user experience.

Another important feature of the proposed system is that it requires very lightweight GUI. The proposed implementation used JAVA SWING of a total size of about 20KB. Additionally, it uses only the mouse, which is a standard component in all of today's computers. Recall that the primary purpose of the JAVA SWING application is to record mouse actions associated with each character of the OTP. Isolation of features, analysis and authentication are done at the server side. Therefore, the application could be easily converted to a complete Web-based application, where all mouse movement actions are cached. Once the user clicks on the login button, all the cached data will be sent as one packet over HTTP connection to the server for authentication analysis.

For all of these reasons, this system could be proposed as an effective Web-login tool designed to target consumer applications such as Web-banking. Enrollment is an easy process of using the system to enter few specially designed OTP passwords. This could be part of the online registration process. Later at login time, the virtual keyboard is used for authenticating the user. It could be used as a stand-alone authentication

tool if OTP is adopted, or, it could be used as a password hardening tool, should fixed memorized passwords be adopted.

8.3 Future Work

Tabulated performance results based on the different analytical models showed substantial difference, especially when comparing the results of the neural networks and the statistical models. This implies that the performance achieved is somehow dependent on the analytical model used, particularly in the feature fusion stage. Therefore, a planned future work is to investigate other analytical models to enhance the overall performance, which falls under *Variance Reduction (VR)*. One suggestion is to study the feature space in an attempt to de-correlate the features using for example the *Principal Component Analysis (PCA)*. By applying the *Singular Value Decomposition (SVD)* of the covariance matrix of the feature vector before training the classifier, a better performance could be achieved. Another suggestion is to study models based on genetic algorithms and artificial immune systems.

The experiment in this research was used primarily for static authentication. The same analogy, but with some design modifications, can promote the proposed system to be used in continuous authentication as well. If a dynamic authentication system (such as a mouse dynamics system) raises suspicions about an active user, she could be automatically requested to re-authenticate her biometrics during her active session. The proposed system used the traditional computer mouse as the input device, and ignored the touch-pad device found in most laptops. Giving research attention to this input device to expand the system scope, could reveal further interesting biometric features that could help broaden the scope and enhance the overall performance of the system.

Furthermore, considering non-Latin characters to generate the shuffled keyboard can extend the application of this static authentication mechanism to other countries that do not necessary use the English standard keyboards.

Bibliography

[1] A. Jain, J. Mao, and K. Mohuiddin, "Artificial neural networks: A tutorial," *IEEE Computer*, 1996.

[2] A. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 3, pp. 165–179, July 2007.

[3] M. Pusara and C. E. Brodley, "User re-authentication via mouse movements," in *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, Washington DC, USA, October 2004, pp. 1–8.

[4] O. Hamdy and I. Traore, "New physiological biometrics based on human cognitive factors," *IEEE International Conference on Complex, Intelligent and Software Intensive Systems*, vol. 0, pp. 910–917, 2009.

[5] O. Hamdy and I. Traore, "Cognitive-based biometrics system for static user authentication," *IEEE International Conference on Internet Monitoring and Protection*, vol. 0, pp. 90–97, 2009.

[6] O. Hamdy and I. Traore, "Homogenous physio-behavioral visual and mouse based biometrics," *ACM Transactions on Computer and Human Interaction (TOCHI), submission number: TOCHI-2010-0020*, March 2010.

[7] "Authentication in an electronic banking environment," Federal Financial Institutions Examination Council," Security Policy Guidance, 2001. [Online]. Available: http://www.ffiec.gov/pdf/pr080801.pdf. Last accessed: August 2007.

[8] J. Ashbourn, *Biometrics: Advanced Identity Verification*. UK: Springer London, 2000.

[9] B. Krebs, "A fresh look at password thieves," The Washington Post," Computer Security, 2007. [Online]. Available: http://blog.washingtonpost.com/securityfix/2007/03/post_3.html. Last accessed: August 2007.

[10] W. E. Burr, D. F. Dodson, and W. T. Polk, "Electronic authentication guideline," National Institute of Standards and Technology, NIST, Tech. Rep. PS-800-63, 2007.

[11] I. C. for Information Technology Standards (INCITS) Secretariat, "Study report on biometrics in e-authentication," Information Technology Industry Council (ITI), Tech. Rep. INCITS M1/07-0185, 2007.

[12] P. Phillips, A. Martin, C. Wilson, and M. Przybocki, "An introduction evaluating biometric systems," *IEEE Computer*, vol. 33, no. 2, pp. 56–63, February 2000.

[13] A. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, January 2004.

[14] L. Cranor and S. Garfinkel, *Security and Usability: Designing Secure Systems that People Can Use*. NY: O'Reilly Media, Inc., 2005.

[15] L. Ma, T. Tan, Y. Wang, and D. Zhang, "Personal identification based on iris texture analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1519–1533, December 2003.

[16] P. J. Phillips, W. T. Scruggs, A. J. OToole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe, "Frvt 2006 and ice 2006 large-scale results," National Institute of Standards and Technology (NIST), Tech. Rep. NISTIR 7408, 2007. [Online]. Available: http://www.frvt.org/FRVT2006/docs/FRVT2006andICE2006LargeScaleReport.pdf. Last accessed: August 2007.

[17] Y.-J. Song, Y.-G. Kim, N. Kim, and J.-H. Ahn, "Face recognition using both geometric features and pca/lda," in *Sixth International Conference on Advanced Language Processing and Web Information Technology ALPIT-2007*, August 2007, pp. 248–252.

[18] S. A. Nazeer, M. Khalid, N. Omar, and M. K. Awang, "Enhancement of neuro-eigenspace face recognition using photometric normalization," in *Computer Graphics, Imaging and Visualisation, CGIV '07*, August 2007, pp. 370 –376.

[19] A. Martinez and A. Kak, "Pca versus lda," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, February 2001.

[20] V. Starovoitov, D. Samal, and D. Briliuk, "Three approaches for face recognition," in *The 6-th International Conference on Pattern Recognition and Image Analysis*, Velikiy Novgorod, Russia, October 2002, pp. 707 –711.

[21] R. J. Mammone, Z. Xiaoyu, and R. P. Ramachandran, "Robust speaker recognition: a feature-based approach," *IEEE Signal Processing Magazine*, vol. 13, no. 5, p. 58, September 1996.

[22] D. Sturim, W. Campbell, D. Reynolds, R. Dunn, and T. Quatieri, "Robust speaker recognition with cross-channel data: Mit-ll results on the 2006 nist sre auxiliary microphone task," in *Proceedings of the fourth working conference*

on smart card research and advanced applications on Smart card research and advanced applications, vol. 4, Honolulu, HI, USA, April 2007, pp. IV–49–IV–52.

[23] D. Dessimoz and J. Richiardi, "Multimodal biometrics for identity documents," Foundation Banque Cantonale Vaudoise, Research Report PFS 341-08.05, 2006.

[24] J. Richiardi, H. Ketabdar, and A. Drygajlo, "Local and global feature selection for on-line signature verification," in *The 6-th International Conference on Pattern Recognition and Image Analysis*, vol. 2, August 2005, pp. 625–629.

[25] J. J. Igarza, I. Goirizelaia, K. Espinosa, I. Hernez, R. Mndez, and J. Snchez, *Online Handwritten Signature Verification Using Hidden Markov Models*. Springer Berlin / Heidelberg, 2003.

[26] J. E. Boyd and J. J. Little, *Biometric Gait Recognition*. Springer Berlin / Heidelberg, 2005.

[27] A. B. Albua, R. Bergevinb, and S. Quirionb, "Generic temporal segmentation of cyclic human motion," *The Journal of Pattern Recognition Society*, vol. 41, no. 1, pp. 6–21, January 2008.

[28] N. Poh and S. Bengio, "Variance reduction techniques in biometric authentication," IDIAP, Tech. Rep. IDIAP-RR 03-17, 2003.

[29] A. Jain, R. Bolle, and S. Pankati, "Biometrics: Person identification in networked society," Kluwer Pulications, Tech. Rep., 1999.

[30] P. Jourlin, J. Luettin, D. Genoud, and H. Wassne, "Acoustic-labial speaker verification," *Pattern Recognition Letters*, vol. 18, no. 9, pp. 853–858, 1997.

[31] G. Hachez, F. Koeune, and J.-J. Quisquater, "Biometrics, access control, smart cards: a not so simple combination," in *Proceedings of the fourth working*

conference on smart card research and advanced applications on Smart card research and advanced applications, Bristol, United Kingdom, 2001, pp. 273–288.

[32] B. Carrier, "Open source digital forensics tools, the legal argument," Federal Financial Institutions Examination Council," Digital Forensic, 2004. [Online]. Available: http://www.digital-evidence.org/papers/opensrc_legal.pdf. Last accessed: August 2007.

[33] R. Gaines, W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: some prelimary results," Rand Corporation, Rand Report R-256-NSF, 1980.

[34] F. Bergadano, D. Gunetti, and C. Picardi, "User authentication through keystroke dynamics," *ACM Transactions on Information and System Security*, vol. 5, no. 4, pp. 367–397, November 2002.

[35] D. Gunetti and C. Picardi, "Keystroke analysis of free text," *ACM Transactions on Information and System Security*, vol. 8, no. 3, pp. 312–347, August 2005.

[36] M. Villani, C. Tappert, G. Ngo, J. Simone, H. S. Fort, and S. H. Cha, "Keystroke biometric recognition studies on long-text input under ideal and application-oriented conditions," in *Conference on Computer Vision and Pattern Recognition Workshop*, June 2006, p. 39.

[37] C.-H. Jiang, S. Shieh, and J.-C. Liu, "Keystroke statistical learning model for web authentication," in *ASIACCS '07: Proceedings of the 2nd ACM symposium on Information, computer and communications security*, Singapore, March 2007, pp. 359–361.

[38] J.-W. Lee, S.-S. Choi, and B.-R. Moon, "An evolutionary keystroke authentication based on ellipsoidal hypothesis space," in *GECCO '07: Proceedings of*

the 9th annual conference on Genetic and evolutionary computation, London, England, July 2007, pp. 2090–2097.

[39] A. Ahmed and I. Traore, "Employee surveillance based on free text detection of keystroke dynamics," in *Handbook of Research on Social and Organizational Liabilities in Information Security*, under press 2008.

[40] S. Hocquet, J. Ramel, and H. Cardot, "Users authentication by a study of human computer interactions," Proceedings of the Eighth Annual(Doctoral) Meeting on Health, Science and Technology, Tech. Rep., 2004. [Online]. Available: http://www.univ-tours.fr/ed/edsst/comm2004/hocquet.pdf. Last accessed: August 2007.

[41] H. Gamboa and A. Fred, "An identity authentication system based on human computer interaction behaviour," in *Proceedings of SPIE*, vol. 5404, 2004, pp. 381–392.

[42] K. Revett, H. Jahankhani, S. T. de Magalhaes, and H. M. Santos3, "A survey of user authentication based on mouse dynamics," in *Proceedings of 4th International Conference on Global E-Security, ICGeS 2008*, London, June 2008, pp. 210–219.

[43] B. Sayed and I. Traore, "Statis authentication based on mouse gesture dynamics," *to apprea in ACM Transactions on Information and System Security.*

[44] A. F. Syukri, E. Okamoto1, and M. Mambo, "A user identification system using signature written with mouse," in *ACISP '98: Australasian conference on information security and privacy*, vol. 1438, Brisbane, July 1998, pp. 403–414.

[45] P. Drucker, *Management: Tasks, Responsibilities, Practices.* New York: Harper & Row, 1974.

[46] C. D. Wickens, J. D. Lee, Y. Liu, and S. E. G. Becker, *An Introduction to Human Factors Engineering.* New Jersey: Pearson Prentice Hall, 2004.

[47] P. Verghese, "Visual search and attention: A signal detection theory approach," *Neuron*, vol. 31, pp. 523–535, August 2001.

[48] R. J. Watt, *Visual Processing: Computational Psychological, and Cognitive Research.* Lawrence Erlbaum Associates, 1988.

[49] D. M. Gary and J. A. Swets, *Signal Detection Theory and Psychophysics.* New York: Wiley, 1988.

[50] U. Neisser, "Decision time without reaction time," *American Journal of Psychology*, vol. 76, pp. 376–385, 1963.

[51] Y. Lio, "Interactions between memory scanning and visual scanning in process monitoring," The University of Michigan, Tech. Rep., 1995.

[52] C. G. Drury, "Inspection of sheet metal materials: model and data," *Human Factors*, vol. 17, pp. 257–265, 1975.

[53] W. K. Estes and H. A. Taylor, "Visual detection method and probabilistic models for assessing information processing from brief visual displays," in *Proceedings of the Nationa Academy of Sciences of the United States of America*, vol. 52, April 1964, pp. 446–454.

[54] R. Miller, "User interface design and implementation," Massachusetts Institute of Technology, Lecture 6.831, 2004.

[55] L. M. Hyman and H. Kaufman, "Information and the memroy span," *Perception and Psychology*, vol. 1, pp. 235–237, 1966.

[56] M. I. Posner, "Immediate memory in sequential tasks," *Psychological Bulletin*, pp. 333–349, 1963.

[57] M. I. Posner, "Rate of presentation and order of recall in immediate memory," *British Journal of Psychology*, vol. 55, pp. 303–306, 1964.

[58] H. Barlow, C. Blakemore, and M. Weston-Smith, *Images and Understanding*. Cambridge University Press, 1990.

[59] I. Gordon, *Theories of Visual Perception*. New York: Wiley, 1997.

[60] R. L. Gregory, "Perceptual filling in of artificially induced scotomas in human vision," *Nature*, no. 350, pp. 699–702, 1991.

[61] I. Biederman, "Recognition-by-components: A theory of human vision understanding," *Psychological Review*, no. 94, pp. 115–147, 1987.

[62] S. K. Card, T. P. Moran, and A. Newell, *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.

[63] D. J. Stang, N. Campus, and C. Wallach, "Exposure duration as a confounding methodological factor in projective testing," *Journal of Personality Assessment*, vol. 39, no. 6, pp. 583–586, 1975.

[64] D. Berlyne, "Novelty, complexity, and hedonic value," *Perception and Psychophysics*, vol. 8, no. 5-A, pp. 279–286, 1970.

[65] D. J. Stand and E. J. O'Connel, "The computer as experimenter: Problems and prospects for social psychological research," *Behavior Research Methods and Instrumentation*, vol. 6, pp. 223–232, 1974.

[66] T. L. Arnow and A. C. Bovik, "Foveated visual search for corners," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 16, no. 3, pp. 813–823, March 2007.

[67] W. S. Geisler and K. L. Chou, "Separation of low-level and high-level factors in complex tasks: visual search," *Psychological Review*, vol. 102, no. 2, pp. 356–378, April 1995.

[68] G. Sperling, "Comparison of perception in the moving and stationary eye," in *Conference of Eye Movements and Their Role in Visual and Cognitive Processes*, Amsterdam, The Netherlands, 1990, pp. 307–352.

[69] R. C. Schultz and R. W. Ives, "Biometric data acquisition using matlab guis," in *Proceedings of the 35th ASEE/IEEE Frontiers in Education Conference*, vol. S1G-1, Indianapolis, IN, October 2005.

[70] R. W. Ives, Y. Du, D. M. Etter, and T. B. Welch, "A multidisciplinary approach to biometrics," *IEEE Transactions on Education*, vol. 48, no. 3, pp. 462–471, August 2005.

[71] V. Barnett and T. Lewis, *Outliers in Statistical Data*. Chichester . NY . Brisbane . Toronto: John Wiley, 1978.

[72] J. Laurikkala, M. Juhola, and E. Kentala, "Informal identification of outliers in medical data," in *Proceedings of the 5th International Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, Berlin, Germany, August 2000, pp. 20–25.

[73] R. W. Ives, Y. Du, D. M. Etter, and T. B. Welch, "A novel approach to fingerprint image quality," *IEEE International Conference on Image Processing, ICIP 2005*, vol. 2, pp. 37–40, September 2005.

[74] S. Müller and O. Henniger, *Advances in Biometrics, Chapter: Evaluating the Biometric Sample Quality of Handwritten Signatures.* Berlin, Germany: Springer-Verlag Berlin Heidelberg, 2007.

[75] A. Peacock, X. Ke, and M. Wilkerson, "Typing patterns: a key to user identification," *IEE Security and Privacy*, vol. 2, no. 5, pp. 40–47, September 2004.

[76] J. Hertz, A. Krogh, and R. G. Palme, *Introduction to the Theory of Neural Computation.* Chichester, England: Addision-Wisely, 1991.

[77] S. Haykin, *Neural networks: A Comprehensive Foundation.* Prentice Hall, 1999.

[78] F. Declercq and R. D. Keyser, "Comparative study of neural predictors in model based predictive control," in *IEEE Computer Society, International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing*, Venice, Italy, Agust 1996, pp. 20–28.

[79] N. Visen, D. Jayas, J. Paliwal, and N. White, "Comparison of two neural network architectures for classification of singulated cereal grains," *Journal of Canadian Biosystems Engineering*, vol. 3, no. 46, pp. 7–14, 2004.

[80] T. Wolbank, M. Vogelsberger, R. Stumberger, S. Mohagheghi, T. Habetler, and R. Harley, "Comparison of neural network types and learning methods for self commissioning of speed sensorless controlled induction machines," in *IEEE PESC 2007 Power Electronics Specialists Conference*, June 2007, pp. 1955–1960.

[81] H. Ghedira and M. Bernier, "The effect of some internal neural network parameters on sar texture classification performance," in *Proceedings of the IEEE International IGARSS '04 Geoscience and Remote Sensing Symposium*, vol. 6, September 2004, pp. 3845–3848.

[82] J. Ortiz-Rodrguez, M. Martnez-Blanco, and H. Vega-Carrillo, "Robust design of artificial neural networks applying the taguchi methodology and doe," in *Proceedings of the Electronics, Robotics and Automotive Mechanics Conference (CERMA'06), IEEE Computer Society*, vol. 2, September 2006, pp. 131–136.

[83] A. Poncet and G. S. Moschytz, "Selecting inputs and measuring nonlinearity in system identification," in *Proceedings of the second ACM workshop on Digital identity management*, Alexandria, Virginia, USA, November 2006, pp. 63–72.

[84] A. Poncet and G. S. Moschytz, "Selecting inputs and measuring nonlinearity in system identification," in *International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing, IEEE Computer Society*, Venice, Italy, Agust 1996, pp. 2–10.

[85] J. Benediktsson, P. Swain, and O. Ersoy, "Neural network approaches versus statistical methods in classification of multisource remote sensing data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 28, pp. 540–552, 1990.

[86] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computer Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[87] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data. Prentice-Hall advanced reference series.* N.J.: Prentice-Hall, 1988.

[88] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A. Jain, "Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems," *Perception and Psychology*, vol. 27, no. 3, pp. 450–455, March 2005.