SIGNAL PROCESSING FOR UBIQUITOUS BIOMETRIC SYSTEMS

by

Danoush Hosseinzadeh, B.Eng
Ryerson University, 2004

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of
Master of Applied Science
in the Program of
Electrical and Computer Engineering.

Toronto, Ontario, Canada, 2006

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Author's Declaration

I hereby declare that I am the sole author of this thesis.
I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.
Signature

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.
Signature

Instructions on Borrowers

Ryerson University requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

Signal Processing for Ubiquitous Biometric Systems

© Danoush Hosseinzadeh, 2006

Master of Applied Science
Department of Electrical and Computer Engineering
Ryerson University

This work presents two hardware independent and ubiquitous biometric solutions that can significantly improve security for computer and telephone related applications. Firstly, for computer security, a GMM based keystroke verification method is proposed along with the up-up keystroke latency (UUKL) feature which is being used for the first time. This method can verify the identity of users based on their typing pattern and achieved a FAR of 5.1%, a FRR of 6.5%, and a EER of 5.8% for a database of 41 users. Due to many inconsistencies in previous works, a new keystroke protocol has also been proposed. This protocol makes a number of recommendations concerning how to improve performance, reliability, and accuracy of any keystroke recognition system.

Secondly, a GMM based text-independent speaker identification scheme is also proposed that utilizes novel spectral features for better speaker discrimination. Based on 100 users from the TIMIT database, these features achieved an identification error of 1.22% by incorporating information about the source of the speech signal. This represents a 6% improvement over the MFCC based features.

Acknowledgments

I would like to thank Dr. Sri Krishnan for his continuous support and mentoring throughout my undergraduate and postgraduate experiences. In many ways he shaped my education and provided with many outstanding opportunities.

Also, thanks to my friends in the SAR lab, especially Arunan Ramalingam, Lam Le, Jiming Yang, and Mohammad Aram for making the lab fun and interesting. A special thanks goes to Dr. Karthikeyan Umapathy for his invaluable assistance on many occasions.

Last but not least, I would like to thank my family and April Khademi for their long time support and encouragement.

Contents

List of Figures

List of Tables

List of Acronyms

AIC	Akaike Information Criteria
AR	Auto Regressive
ANN	Artificial Neural Network
AWGN	Additive White Gaussian Noise
CMN	Cepstral Mean Normalization
dB	decibel
DARPA	Defense Advanced Research Projects Agency
DDKL	Down-Down Keystroke Latency
DET	Detection Error Tradeoff
EER	Equal Error Rate
FAR	False Acceptance Rate
FFT	Fast Fourier Transform
FRR	False Rejection Rate
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IID	Independently Identically Distributed
IIR	Infinite Impulse Response
KD	Keystroke Down Time
KL	Keystroke Latency
LPC	Linear Prediction Coefficients
LPCC	Linear Prediction Cepstral Coefficients
MDL	Minimum Description Length
MFCC	Mel Frequency Cepstral Coefficients
NIST	National Institute of Standards & Technology
NN	Neural Network
RE	Renyi Entropy
ROC	Receiver Operating Curve
SBE	Spectral Band Energy
SBW	Spectral Bandwidth
SC	Spectral Centroid

SCF	Spectral Crest Factor
SE	Shannon Entropy
SFM	Spectral Flatness Measure
SNR	Signal to Noise Ratio
SVM	Support Vector Machine
UBM	Universal Background Model
UUKL	Up-Up Keystroke Latency

Chapter 1

Introduction

IN recent years there have been many advances in the areas of telecommunication and computer technology. These advances have led to what is commonly referred to as the "information revolution". This revolution has transformed the way we live, work, and even the way we entertain ourselves, by allowing unlimited access to a seemingly infinite amount of information and resources at any moment. This has led to many new and modern conveniences, but it has also created some serious problems with respect to personal privacy, security, and fraud.

The Internet and the advanced global communication networks allow people to virtually visit places or do things that have traditionally required a physical presence elsewhere. Simple things such as commerce, entertainment, research, shopping, and many other tasks can be performed from literally anywhere in the world because various interconnected computer and communication networks can provide a direct link to almost any institution in society. Unlike decades past, people really do not need to leave their house in order to pursue the natural tasks of daily life. For instance, almost anything can be purchased online and delivered to your doorstep, all institutions can provide full customer service over the telephone or the Internet, personal and business accounts can be modified online, people meet others online, people can work from home, and the list goes on.

For an average person, these advances translate to a great deal of conveniences and the freedom of being able to do things outside of official business hours and without delays.

Online and telephone banking are great examples of this freedom, where people no longer need to visit the bank to complete a transaction. If you are "connected", connected to the Internet or a telephone network, then almost anything can be done remotely from anywhere and at anytime.

Of course these conveniences come with certain drawbacks. Aside from changes to the fabric of society, such as increased health problems attributed to inactive lifestyles, always being connected can lead to other devastating circumstances; either financially, personally, or otherwise. For instance, identity theft, invasion of privacy, vandalism, and electronic theft are some of the issues that have received much attention in recent years due to their prevalence. Many of these crimes happen online or over the telephone, where a little information about a person can give criminals unauthorized access to bank accounts, email accounts, home computers, and networked computers to name a few.

These problems can only be solved through much more stringent security protocols, but unfortunately most people are either technically unable or unwilling to enhance security because available technologies are either difficult to use or expensive. Therefore, simple and intuitive solutions are needed to meet today's security challenges.

1.1 What Are Biometrics?

Biometrics refer to automated systems that attempt to uniquely identify individuals by measuring some physical or behavioral characteristic. For this purpose, any good biometric must have at least three important properties [41]:

1. It should measure a characteristic that all of the population possess (i.e. a fingerprint).

2. This characteristic should be unique for every person in the population.

3. The characteristic should not vary significantly under a given set of test conditions.

The first condition is practically impossible because some people will be unable to provide certain biometrics. For example, an amputee may not have any fingerprints or a blind person may not be able to provide an iris or retinal biometric. In most cases, this condition must be relaxed such that it includes most of the population. The second condition is a necessary condition for recognition and along with the third condition determines how accurately the biometric will perform. Although these condition seem stringent, there are many types of biometrics suitable for use with human subjects.

Biometrics, primarily fingerprints, have long been used by government agencies and police departments as a way to identify criminals. In recent years, large non-governmental organizations have also started to use biometric technologies [57][35] because of the need to protect data, resources, assets, and wealth. For example, entrance gates at Walt Disney World, ATMs, credit cards belonging to Tokyo-Mitsubishi bank, and major US airports are just a few.examples of where biometric technologies are being used in industry. Nevertheless, even though biometric technologies have many potential applications for everyday use by the general population, they have historically been inaccessible by the masses because of high cost. This trend is slowly changing due to intense anxieties about personal privacy, fraud, and other security concerns.

1.2 Security Benefits of Biometrics

It is important to ensure that security issues can be resolved in a seamless and effective manner. Presently, the username and password authentication model is almost exclusively used for all computer security needs on the Internet and on computer networks. Although this technique is simple to implement, it has many potential weaknesses [1]. For example, lost or stolen passwords, password sharing, easy-to-crack passwords, or a user's inability to remember his or her own login information can leave entire computer networks open for misuse or attack. Often times the problem is compounded because users have multiple accounts within different systems, and therefore, they choose easy-to-remember passwords that can be easily cracked using dictionary attacks or terms associated with the individual [1][55]. Similar problems exist for telephone based authentication protocols, where a few pieces of personal information such as a birthdate and an address are often enough to impersonate someone and "hack" into bank accounts, credit card accounts, or mobile phone accounts.

Biometrics seem to be a good response to the current problems that are a byproduct of the information revolution. As the vast amounts of information at our fingertips grows, and the ability to perform sensitive tasks remotely becomes common place, security measures should also be enhanced to protect the users. However, any security enhancement is not likely to be widely accepted if it significantly complicates the authentication process for valid users [57][1]. In this regard, some biometric technologies such as voice recognition, face recognition, keystroke recognition, and fingerprints can add a nearly effortless authentication process with significant security benefits, especially for those tasks that are performed remotely.

1.2.1 Computer Security

As already mentioned, computer security protocols are often based on the simple username and password model. This security protocol can be very effective if used properly by the users, but often times users are not able or do not understand how to use it effectively, as described in Section 1.2. This problem is further complicated because users often require the flexibility to remotely access home or office computers via the Internet. Biometric technology could eliminate these problems by ensuring that only the rightful users are given access to protected resources.

Another major weakness with the username and password security model is that it is a static authentication protocol. Static authentication refers to a one-time validation of the user's identity, which usually occurs at the beginning of a new session. This authentication mode is almost always used because of its simplicity and practicality. However, static authentication does not ensure that the protected resource will remain under the control of the rightful user. With static authentication, it is completely possible for an intruder to get access to the resource after the initial authentication stage, with or without the knowledge or cooperation of the valid user.

In contrast, continuous authentication is much more secure because it ensures that the protected resource is never "hijacked". This is achieved by continuously monitoring the user's activity, which can indicate who is in control. Keystroke authentication and facial recognition are well suited for continuous authentication because these technologies can gather data without the user's participation. Continuous authentication is therefore a huge leap from the traditional static authentication because it can ensure that the rightful user remains in control of the protected resource for the entire duration of the session.

1.2.2 Identity Theft and Fraud

In general, identity theft is the use of an identity not belonging to the user for the purpose of fraudulent activities. These types of crimes have become one of the fastest growing crimes worldwide and the fastest growing crime in the United States (U.S.) [7]. In fact, the Boston based Aberdeen Group, reports that the losses of individuals and businesses worldwide due to identity theft was an estimated US $221 billion in 2003 [35]. Although reports vary, these losses are expected to grow as much as 300% per year, according to different sources [35][17].

Since personal, banking, and credit card information is easy to obtain by criminals, identity theft has become a common problem. In fact, the huge financial damage inflicted by criminals in one year is more than enough to eliminate disease and hunger all over the world [68]. Yet, the solution to these problems are very simple and can be solved by any of the leading or emerging biometric technologies [35].

1.3 Biometric Technologies

In general biometric systems can function in two modes; either as a verification system or as an identification system. In an identification system, the goal is to choose the identity of the user from all known users in the database based on the provided biometric signature. In contrast, a verification system attempts to validate a claimed identity based on the provided biometric signature. These two modes are very different because identification requires a N-group classification scheme, where as verification requires a binary classifier (that is the claimed identity is verified or rejected); where N is the number of users in the database.

Therefore, identification is considered to be a more difficult task than verification. Throughout this work, the terms 'recognition' and 'authentication' are used interchangeably, and they refer to the two different tasks of verification and identification.

In the past, many different techniques for user authentication have been proposed. In fact, all of these techniques can be categorized in one of the following three groups:

- Group A: Those that require some secret information from the user such as a password.
- Group B: Those that require the user to provide a physical object such as a key or an electronic access card.
- Group C: Those that rely on biometric information.

The biometric based technologies that belong to Group C have several advantages when compared with the authentication schemes of Groups A or B. Since biometrics rely on information that is embedded in an individual they cannot be lost, stolen, transferred and are usually very difficult to copy. These qualities make biometric technologies very secure and the preferred choice for many applications. Increasingly, societies will need to rely on these technologies to meet the need for enhanced security. Figure 1.1 shows the approximate popularity of different biometric technologies based on market share.

1.3.1 Physical Biometrics

Physical biometrics are those that evaluate an anatomical characteristic for the purpose of person recognition. These biometrics work by capturing an image or a signal from the body, such as a fingerprint, retinal pattern, iris pattern, hand geometry, and speech. Therefore, most of the physical biometric technologies (excluding speech) require advanced image processing techniques that can match a template image to the sample image with good accuracy. As a result, physical biometrics often require sophisticated imaging sensors to produce good results. The only exception to this general rule is the fingerprint scanner which in recent years has been produced significantly cheaper and smaller [56].

Iris, retinal, and fingerprint pattern recognition are some of the most accurate biometrics

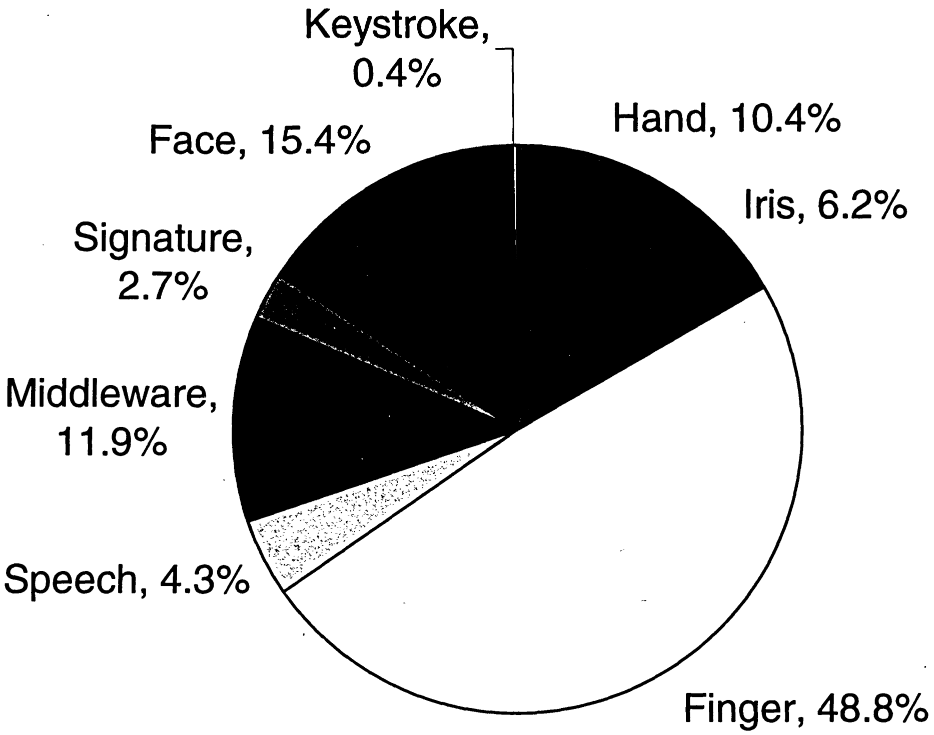


Figure 1.1: Percentage of market share for different biometric technologies as of 2001. Another report published by the same group in 2006 indicates that these figures are largely unaffected [34].

because these patterns remain the same for decades and are known to provide a truly unique signature [65][80]. Facial recognition and voice recognition have had less success commercially because of the variabilities that must be overcome to use these technologies effectively. However, it should be noted that voice recognition is significantly affected by the behavioral state of the individual and therefore, it is not a true physical biometric. Lastly, hand geometry based biometrics can be effective but are not as cost effective, secure, or unique as fingerprint based systems [57].

1.3.2 Behavioral Biometrics

Behavioral biometrics are those habitual characteristics than are observed from the way people perform certain tasks. Often times there is a pattern to these characteristics over which there is little or no conscious control, hence these characteristics can be unique. Good examples of behavioral biometrics are keystroke recognition, handwritten signature recognition, and gait recognition (gait is the analysis of posture for a moving person). The main use of gait recognition systems is for surveillance from video captured by security cameras [37]. As an interesting side note, U.S. government agencies such as the Federal Defense Advanced Research Projects Agency (DARPA), are interested in using this technology for identifying suspicious individuals in airports and other sensitive facilities [72].

In general, behavioral biometrics are less accurate than physical biometrics because there is some variability in the patterns being observed; since these patterns are not a physical characteristic. However, some behavioral biometrics such as keystroke recognition and signature recognition are extremely easy to implement and integrate into existing security systems either because of existing infrastructure (for keystroke recognition and gait) or because of a long history of usage for authentication (signature recognition).

1.3.3 Privacy Concerns

Although biometric technologies can be quite useful, there is some cause for concern. Many have concerns about who has access to the biometric signatures and how they will be used. As a result, some people may be uncomfortable or unwilling to provide their biometric signatures in fear of abuse or misuse of the data. This is a particular problem for physical biometrics because the data must be voluntarily provided, hence creating the impression of "big brother", among other concerns. However, there tends to be a much more relaxed attitude about biometrics such as keystroke recognition, voice recognition, or face recognition since there is no explicit attempt at collecting data. For these biometrics, and other behaviorial biometrics, data can often be collected during the process of normal activities without requiring the user to perform specific tasks. For example, speech can be easily collected from a telephone conversation, keystroke patterns can be collected from regular computer use, and facial scans can come from either a computer camera (webcam) or security cameras. Thus, people accept these technologies more easily because they are not intrusive.

There are also other concerns about biometrics. Of particular interest are some physical biometrics which have been shown to have diagnosing capabilities with respect to detecting disease. Although the evidence is not conclusive, there are some studies that suggest a link between fingerprint patterns and certain genetic and non-genetic diseases and also, iris and retinal patterns can be used for detecting hypertension and arteriosclerosis, in addition to other eye related conditions [80]. This could open the door for disease screening using these technologies and could also raise questions about the storage of biometrics and access to the biometric signatures. The behavioral biometrics are much less susceptible to these issues because they do no gather direct biological measurements. This is another reason why they may be better accepted by the public.

These issues aside, physical biometric technologies require large capital investments and this is the main reason that they are not widely used. These costs stem from the need to purchase specialized hardware to capture the biometric data. Therefore, many of the physical biometrics are not well suited for large scale deployment over the Internet or over telephone lines. The only exception to this general rule is speech based biometrics because of the extensive worldwide telecommunication infrastructure that already exists.

1.4 Organization of Thesis

This thesis focuses on biometric technologies that are easy to use and easy to integrate into the existing security infrastructures, which therefore implies technologies that are cost effective and practical. To this end, Chapter 2 introduces two ubiquitous biometric technologies, namely keystroke and voice recognition, and gives the reader some details on why they are effective and the current state of these technologies. The remainder of this thesis is organized as shown in Figure 1.2.

Chapter 3 focuses on a novel method for keystroke verification based on Gaussian mixture models (GMMs). Details regarding the GMM estimation process, the decision criteria for user verification, and the experimental results are also discussed. Chapter 3 also presents

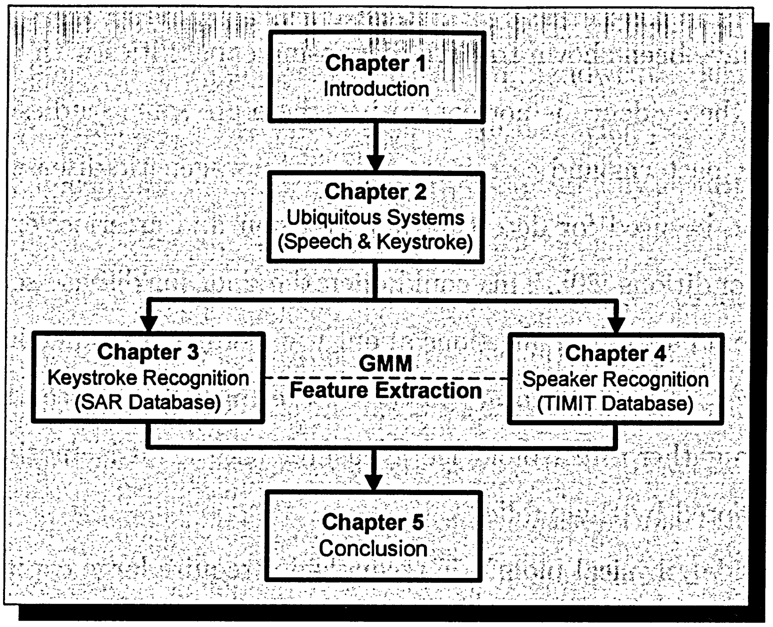


Figure 1.2: Organization of thesis.

a detailed protocol for use with keystroke recognition systems which is primarily useful for accurate data collection, enhancing classifier performance, and accurate reporting of experimental results. Chapter 4 discusses a GMM based voice identification scheme utilizing spectral features in addition to the commonly used features and presents some experimental results regarding the classification performance of the proposed features on the standard TIMIT database [50].

Chapter 5, which is the last chapter, presents the conclusions, recommendations, and future works resulting from the presented studies. In summary, this work proposes a voice identification scheme and a keystroke verification system that can meet modern security needs, especially for the Internet and telephone-based applications.

Chapter 2

Ubiquitous Biometrics

ACCESS to the Internet and vast telecommunication networks has revolutionized the way people interact with society and the institutions of society. Due to factors outside the scope of this text, life has become "fast-paced" and therefore people often look to technology to provide convenient solutions. As a result, there has been a lot of effort by business and governments alike to allow users to remotely perform tasks that have traditionally been performed in person. Among other benefits, these conveniences have also produced significant cost savings for those involved.

Remote activity refers to transactions that are performed online via the Internet or over a telephone that would otherwise have to be done in person at some specific location. Today, remote transactions can be conducted with banks, retailers and wholesalers, governments, businesses, computer networks, websites, and the like. Consequently, there has been an increased rate of fraud, identify theft, vandalism, and other privacy related concerns. In many cases, remote transaction fraud can be prevented if the true identity of the client could be verified remotely.

Biometric technologies are particularly well suited for these kinds of problems because they do not rely on knowledge that can be stolen, such as passwords, bank or credit card account information, social security numbers, and so on. Instead, biometrics rely on some intrinsic characteristic within individuals that is difficult to reproduce even if the perpetrator has good knowledge of the person being victimized. Despite these advantages, biometric technologies can be very expensive to implement on a large scale. For example, would it be possible to equip every computer connected to the Internet and every telephone in the world with a fingerprint, iris, or even a facial scanner? If this is not possible, then criminals have an easy way to bypass these particular biometric technologies. Then, the problem becomes how can the existing infrastructure be used to implement a biometric security system?

To overcome the lack of infrastructure that exists for many biometric technologies, ubiquitous biometric systems should be pursued.

ubiquitous - 'present, appearing, or found everywhere.'

- *Compact Oxford English Dictionary of Current English*

The two ubiquitous technologies for which a worldwide infrastructure exists are keystroke and speech biometrics. The existing telecommunication infrastructure can be used for speech biometrics and all computers are equipped with a keyboard, which could be used for keystroke biometric. Using these two technologies, the vast majority of all remote access transactions which are performed using a computer or a telephone, can be made much more secure; because the identities of the remote users can be accurately verified.

No other biometric technology is ubiquitous. Some of the most reliable biometric technologies such as fingerprints, iris, and retinal patterns can be extremely expensive to implement and somewhat intrusive, therefore, they are not widely used. Although fingerprint technology is becoming more affordable, it is still far from becoming standard hardware on general purpose computers or telephones. Thus, it is also far from being a ubiquitous biometric technology. Due to these concerns, there is a lack of cheap, secure, and easy to implement biometric technologies in the current marketplace.

Keystroke and speech based user recognition systems may be the solution to these problem because of their truly ubiquitous nature. Again, this is supported by the extensive infrastructure that exists worldwide for speech communication and the fact that all personal computers are equipped with a keyboard. It is also important to point out that these two ubiquitous technologies will likely be readily accepted by the general public because they can be seamlessly integrated into existing security protocols. This chapter provides more details on these two technologies with respect to user recognition.

2.1 Keystroke Recognition

Keystroke recognition is the process of analyzing human typing patterns in order to identify individual computer users. This biometric is a very natural and practical authentication method for any computer security application because keyboards are an integral part of how humans interact with computers. As a result, keystroke recognition can be integrated into existing security protocols very easily, without the need to invest in any additional hardware or change the way that users login (hardware independency is further discussed in Section 2.1.1). This is a critical point because it implies that this technology can be used from any laptop or personal computer, anywhere in the world. This amazing flexibility can be applied to remote authentication via the Internet or local authentication of computer users on a network or a personal computer.

Another advantage of keystroke recognition is that it can be used in both static and continuous authentication modes. As described in Section 1.2.1, continuous authentication schemes are much more powerful than the traditional one-time or static authentication techniques because a session that has been initiated by a valid user cannot be hijacked by any other person. Regardless, whether keystroke recognition is used for static or continuous authentication, the keystroke pattern can be captured with any keyboard, any computer, and it can be completely implemented in software. For these reasons, this technology is hardware independent, secure, and conveniently ubiquitous.

Keystroke recognition can be used to enhance the username and password authentication model simply by examining the way these strings are typed. This would add a hidden layer of security that can greatly improve the reliability of the username and password security protocol. Keystroke recognition can protect computer systems against unauthorized access even if authorized persons have revealed, lost, or shared their login information. In effect, keystroke patterns can be used as a digital signature for the purpose of validating an identity.

This idea has many computer based applications, content-control applications, and would be especially beneficial for online banking, email, and user account protection, just to name a few.

The use of a digital signature to enhance security for the username and password model is more convenient for the end users as well. For example, if people can reuse the same login information for all their computer security needs, it will save them the frustration of forgetting this information for rarely used accounts. In this case, users would be relying more on the uniqueness of their digital signature than a secret password. Although this is not a necessary condition, it is certainly possible.

2.1.1 Uniqueness of Keystroke Patterns

As early as 1975, scientists have noted that keystroke patterns have characteristics that are unique to individual typists [74]. In fact, well before the advent of the computer and throughout the 19th century, telegraph operators were known to recognize each other by the rhythm of their Morse code [39][74]. However, experimental work in the area of keystroke recognition did not start until the 1980s [26][36][78][38][6][3].

Keystroke patterns as a biometric is based on the principle that every person has a unique typing pattern, similar to a hand written signature [26][78]. In fact, the same neurophysiological factors that create unique signatures, also produce unique keystroke patterns [36]. In particular, for regularly typed strings, these patterns can be very consistent and therefore, they can be effective for user recognition. This is further supported by studies that show a great deal of keystroke pattern variability even among professional typists [26], which implies that forgery is very difficult even if the imposter is a good typist.

Furthermore, because of the way the they are produced, keystroke patterns are also hardware independent. Research has shown that typing is a very structured process where a certain amount of text is stored in a short-term buffer somewhere in the brain and finger movements are planned accordingly before execution (typing) [71][70][76]. This implies that regardless of the type of keyboard or computer, each user has preplanned finger movements for a given text and that the physical size and shape of keyboards does not effect the way the brain coordinates the finger movements. Thus, keystroke patterns are a true behavioral biometric and are likely not severely impacted by the type of keyboard (hardware) used to capture them.

We further argue that a person's keystroke pattern (or digital signature) would be much harder to duplicate than a handwritten signature because the imposter cannot know the keystroke pattern and at the same time, the imposter cannot practice another person's keystroke pattern. A signature is much more susceptible to being copied because it can be observed and practiced many times until a reliable forgery can be produced. This is unlike keystroke patterns which cannot be perfectly observed because of natural typing speeds and the complexity of up to ten simultaneous finger movements. More importantly, keystroke patterns can not be practiced by an imposter because there is no feedback mechanism to indicate the quality of the forgery.

For further protection, in a commercial system, a user who cannot successfully log in after a predetermined number of attempts (i.e. after 3 failures) can be locked out from the system or be subjected to intense observation. This mechanism would severely limit an intruder's practice time. This level of control is needed because imposters have been known to take extreme measures to defeat security systems. Even the most secure biometrics such as fingerprints are susceptible to duplication or forgery [57], especially for databases with a large number of users.

2.1.2 Keystroke Features

Keystroke identification examines the timing pattern that is produced as a typist presses the different keys on the keyboard. From this typing pattern, there are several unique features that can be extracted, these are shown in Figure 2.1. One such characteristic (feature) is the key down time (KD), which is the amount of time that a particular key is held down. Another feature is the keystroke latency, which is the time between pressing two consecutive keys; we shall refer to this feature as the down-down keystroke latency (DDKL). These two features

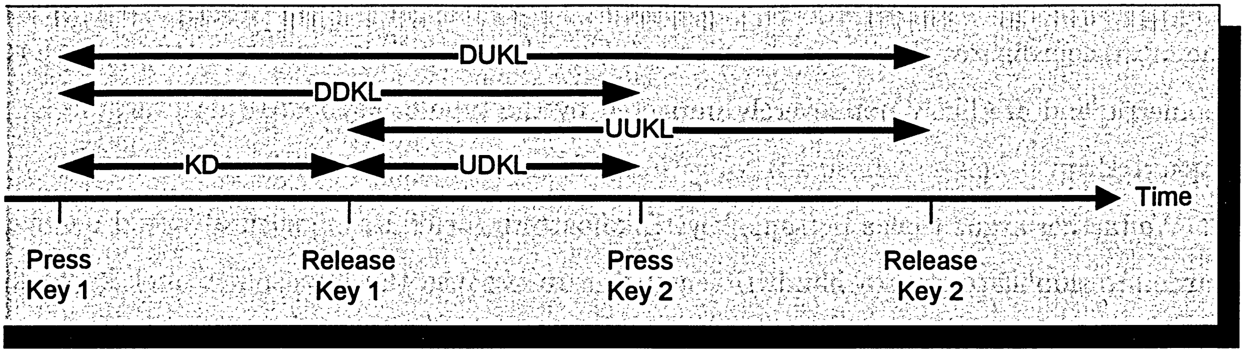


Figure 2.1: Diagram illustrates how features are extracted from keystroke timing patterns.

have been used in previous research to produce good results in user identification. Similar to the DDKL feature, the up-up keystroke latency (UUKL) is another latency measure which measures the time between releasing two consecutive keys. This feature is being used for the first time in this work.

In [3], Araújo et al. have shown that other keystroke latency measures such as the up-down keystroke latency and the down-up keystroke latency can also be effective features for user identification. For clarity, the up-down keystroke latency is the time between releasing one key and pressing the next key, while the down-up keystroke latency is the time between pressing one key and releasing the next key. These two features were not analyzed and are mentioned here for completeness.

In general, for a N character string, there are $N-1$ keystroke latency (KL) data points for each latency measure and N KD data points. Figure 2.2 shows the DDKL and KD plots for a particular user that has typed his name repeatedly. Figure 2.2 is included to illustrate the stability and correlation that exists between each of the feature vectors, KD and DDKL. Similar characteristics can be seen for the UUKL feature.

2.1.3 Previous Works

One of the earliest works in the area of keystroke recognition was presented in [26] by Gaines et al., based on a statistical study of keystroke latencies. Seven secretaries were asked to

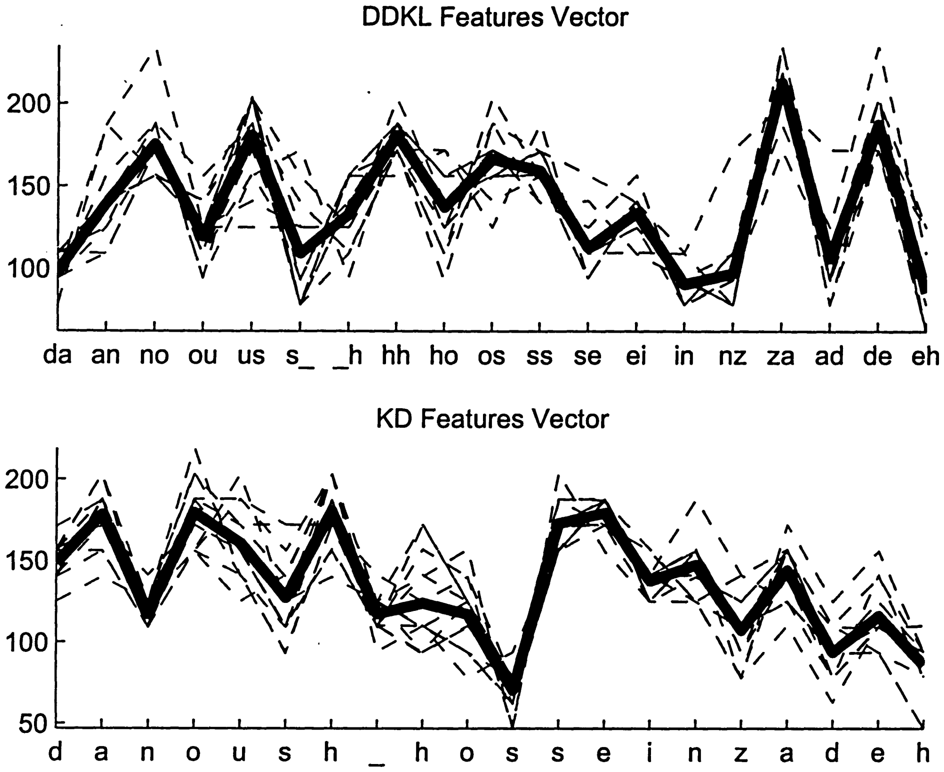


Figure 2.2: Several plots of the keystroke latency (DDKL) and key down time (KD) feature vectors for one user. The bold line is the average of the vectors. The space character is represented by "_".

type three passages of text between 300-400 words at two different times, separated by four months. By performing a classical two sample t-test of keystroke latencies, while assuming a log-normal distribution of these latencies, they were able to achieve a 0% error rate. However, despite these impressive results, this experiment had three major shortcomings. Firstly, only seven users were enrolled in the system, which is too few for reliable error results and secondly, each of the users were expert typists, which does not represent the average user in a practical situation. Thirdly, each user was required to type a lengthy passage during both the authentication and enrollment sessions, which is also not practical. Despite these concerns, this seminal work does serve to show the potential of keystroke identification as a biometric.

Umphress and Williams [78], and Leggett and William [38] used a distance based classifier for keystroke recognition. In this approach, authentication was confirmed if 60% of the sample features were within 0.5 standard deviations of the reference pattern. These experiments were similar to those in [26], because in both the training and verification stages, lengthy passages of up to one thousand characters were typed by each user. These results are still important because they significantly improve on two of the shortcomings of [26] by increasing the number of users and including typists of varying abilities. With seventeen users a false acceptance rate (FAR) of 5% and a false rejection rate (FRR) of 5.5% was achieved.

Another contribution of [38] was the introduction of a single temporal low-pass filter to remove the outlier data (extreme latencies) caused by long pauses or abnormalities in a user's typing pattern. Although Mahar [42] has shown that different typists require different filters because the mean of keystroke latencies can range from 96ms for expert typists to 825ms for novice typists. Therefore, a single low-pass filter would not be appropriate for all users. These outlier values can be viewed as noise in the recorded keystroke signals and their removal improves the performance of statistical or distance based classifiers for user recognition.

In fact, many authors have proposed simple classifiers based on lower order statistical moments, distance, or probabilistic methods assuming a Gaussian distributions for keystroke features. In [36], Joyce and Gupta used keystroke latencies from four strings (first name, last name, username, and password) and calculated the L_1 norm between the test strings and the reference strings. With this simple classifier they reported an FAR of 0.25% and a FRR of 16.36% for 32 users. Here, the authors do not report any information on failures due to typographical errors or corrected errors (errors that are corrected using backspace). Since both of these errors cannot be modeled, they are always ignored by keystroke recognition systems. Given that four strings are required for authentication in [36], the failure to acquire rate may be an important factor for analysis. Another even important factor is that the test data used was collected from the valid users immediately after the enrollment session. This condition does not account for much of the variability in keystroke patterns. For best results, test samples should be collected at several different sessions as recommended by Mansfield and Wayman in their benchmark study on biometric evaluation practices [44].

In [47], Monrose and Rubin perform a study of 63 users and four different classifiers; Euclidian distance classifier (83%), Gaussian classifier (86%), weighted Gaussian classifier (87%), and a modified nonlinear Gaussian classifier (94%), where the values indicate the classifiers performance and a Gaussian classifier indicates that the keystroke features were assumed to have a Gaussian distribution. Of course this is not an optimum solution since there is no data to suggest that such an assumption is entirely correct. As a result, the performance of the classifiers reflects this approximation and can be expected to decrease with more users. This is evident since the nonlinear classifier performed best, which indicates that keystroke patterns (or keystroke features) do not have a Gaussian distribution. Despite these conclusions, these results are further hampered because of the seemingly large number of training samples used to train these classifiers. This fact is not explicitly stated, but data was collected from users over a period of 11 months and half of this data was used for training and half for testing. These conditions are therefore not conducive for realtime identity verification since training data cannot be captured in this manor for any practical application.

Since keystroke data appear to have non-Gaussian distribution, others have used more sophisticated methods to model these patterns. Neural networks (NNs), hidden Markov models (HMMs), support vector machines (SVMs), and numerous clustering techniques have been applied. Lin [40], Obaidat and Sadoun [52], and Brown and Rogers [8] have used many types of neural networks with some success on small databases. Here, a major concern is the number of training samples needed to train the NNs in [8] and [52], which are approximately 70 and 900 respectively. Although good results are reported in [40], little information is given about the number of users and the results were based on a total of 151 tests which includes valid user tests and imposters tests. Certainly these methods should be explored further with more users, more tests, and ways to reduce the training samples to a practical amount. Nevertheless, a major concern with these techniques is the complexity of the neural nets used. Since they must be retrained each time a new user is introduced to the system, their scalability and usage in many situations will be limited. Also, the need to include imposter samples in the training set, is not a desirable characteristic because the training set will not be representative of all imposters.

In [82], Yu et al. have applied a SVM approach to keystroke identification, although a major issue with their technique is the large number of training samples needed for enrollment (up to 400 samples per user). In [12] and [10], Chang and Chen used HMMs for keystroke recognition and achieved seemingly good results. However, a major flaw that exists in this experiment is that only 10 samples were collected from each user for testing and they were all collected after the training session. These are too few samples for accurate and reliable results and this procedure does not capture the natural variabilities in keystroke patterns since all of the test data was collected after the training session; leading to highly correlated training and test data. This is similar to a facial recognition system reporting results based on one session, in the same environment, and from the same viewing angle. As mentioned earlier, for best results, test samples should be collected at several different sessions [44], and over a much longer period of time since keystroke patterns are a behavioral biometric and are effected by user's state of mind.

Some other works such as [43] have used a fuzzy c-means classifier with little success. In [3], Araújo et al. showed that classifier performance can be improved by using multiple features, however they used an unreliable experimental setup similar to that of [10] and [12]. In [6], Bleha showed a plausible exponential relationship between string length and classifier performance that supports results reported for other types of classifiers.

2.2 Speaker Recognition

Speaker recognition is the process of recognizing an individual based on an utterance from the speaker. For clarity, it is noted that *speaker recognition* (which encompasses speaker verification and speaker identification) is different from *speech recognition*. Speech recognition is the process, usually performed by a computer, that analyzes speech for the purpose of understanding the content of an utterance; i.e. the words that were spoken and not *who* spoke those words. And as already mentioned, the difference between speaker verification and speaker identification is that in speaker verification the system attempts to authenticate a claimed identity, whereas in speaker identification the goal is to pick the correct user from a group of possible users entirely based on a sample utterance.

Speaker recognition is particularly useful in telephone based applications. Often, when using the telephone, identities are validated through some personal information such as a person's name, address, and the like. This type of authentication is not very secure because it is extremely easy to obtain this kind of information about other people. Speaker recognition can greatly improve these security problems since it provides a method for validating the identity of a user. And by implementing a text-independent speaker recognition system, forgeries will be very difficult because imposters will not be able to use pre-recorded utterances of valid users during authentication.

2.2.1 Uniqueness of Speech

The simplified human speech system is composed of the larynx and the vocal tract, which extends from the larynx to the mouth and lips, refer to Figure 2.3 for an illustration. The larynx (sound box) contains small fibers (vocal cords) that are capable of vibrating at a broad range of frequencies when air from the lungs is pushed through them. The vocal tract acts like a tube of varying thickness to shape the frequencies which are emitted from the vocal cords. These components work in tandem to produce speech quality sounds that are generally below 8kHz. The oscillatory part of speech is known as the voiced component and is caused by vowel-like sounds. The voiced components of speech have a well defined periodic shape in the time domain. Unvoiced components, which are also produced by air forced from the lungs sound like 'sh', 's', or other consonants. These unvoiced components of speech appear like random noise in the time domain but they are an essential part of the

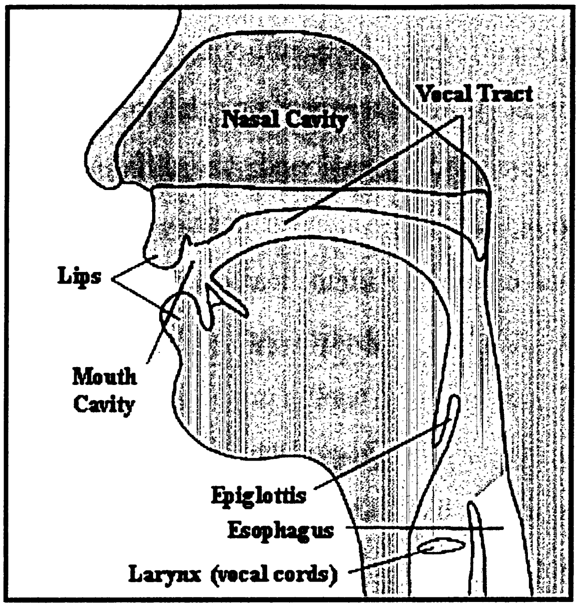


Figure 2.3: Simplified diagram of human speech production system. Adapted from [27] with permission.

signal.

In essence, the speech system can be modeled like any other system with an input, a filter, and an output. The input to the speech system is the periodic oscillations produced by the vocal cords or air from the lungs, the output is the speech signal, and the vocal tract acts as a time-varying filter that modifies the input signal to produce speech or other sounds in general. Of course, the shape, thickness, and length of the vocal tract is controlled by a group of muscles as well as the way the speaker learns to speak. As a result of these anatomical and behavioral differences, the configuration of the vocal tract for a given sound is a unique speaker-dependent characteristic.

Often in literature, the entire speech system is modeled with a time-varying excitation and a time-varying filter [59][28][9], see Figure 2.4 for an illustration. Therefore, using this model, the speech signal ($s(t)$) is given by:

$$s_{voiced}(t) = x(t) * h(t) \tag{2.1}$$

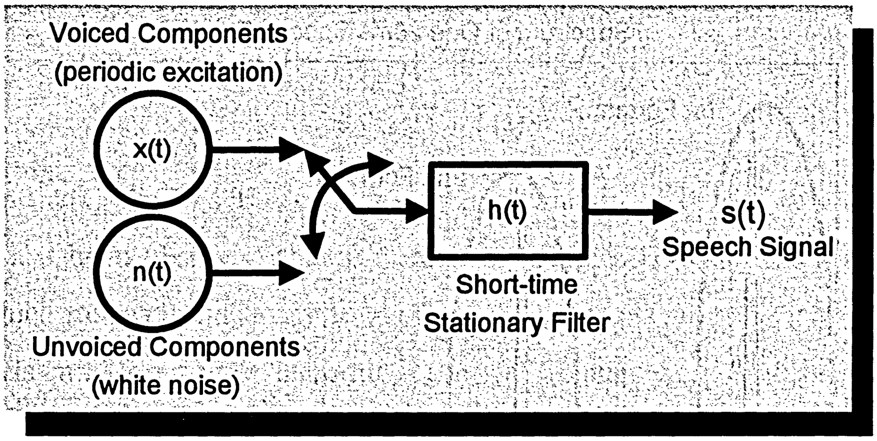


Figure 2.4: Human speech production model.

$$s_{unvoiced}(t) = n(t) * h(t) \tag{2.2}$$

where, $x(t)$ is a periodic excitation, $n(t)$ is white noise, and $h(t)$ is a time-varying filter which constantly changes to produce different sounds. Although $h(t)$ is time varying, it can be considered stable over a period of few milliseconds (ms); typically around 10-30 ms is commonly used in literature [28][9][79]. This convenient short-time stationary behavior is exploited by many speaker recognition systems in order to characterize the vocal tract configuration, given by $h(t)$. This information can be easily extracted from the speech spectrum using well established deconvolution techniques [59]. Since the anatomical configuration assumed by the vocal tract for a given sound is a unique speaker-dependent characteristic, there has been a great deal of advances in speaker recognition by exploiting this characteristic.

2.2.2 Speech Features

Two of the most basic features that can be extracted from speech are pitch and formants. Pitch is defined as the fundamental frequency produced by the vocal cords and along with its harmonics, they can be clearly seen in the speech spectrum as spikes. The formants, are the resonant frequencies of the vocal tract and appear as large amplitude humps in the speech spectrum. These two features are the most dominant structures in the speech

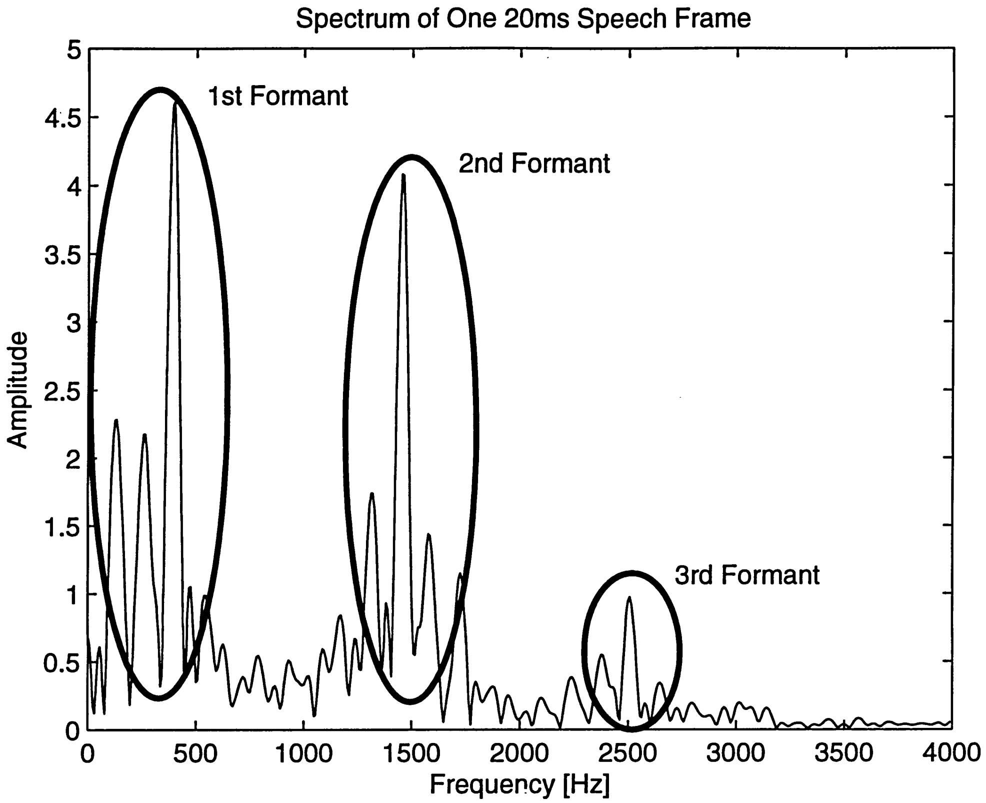


Figure 2.5: Figure illustrates the concept of formants and pitch from the speech spectrum. The location of formants are circled and every spike in the spectrum represents one particular pitch frequency.

spectrum, refer to Figure 2.5 for an illustration. Although these features provide some speaker-dependent information, they have mainly been used for male vs. female classification and speech recognition (understanding what has been said).

The most dominant speaker-dependent features that have been used to date, have been cepstrum based features. The cepstrum operator is often found in literature under homomorphic deconvolution and therefore, it can separate the components of speech found in Equation 2.1 and Equation 2.2. This powerful tool then permits for separate analysis of the vocal tract configuration (given by the filter component ($h(t)$) which is highly speaker-dependent. The cepstrum of the signal $s_{voiced}(t) = x(t) * h(t)$ is given by:

$$\begin{aligned} Cepstrum\{s_{voiced}(t)\} &= FFT^{-1}\{|\log FFT[\,x(t) * h(t)\,]\,|\} \qquad (2.3) \\ &= \overline{x(t)} + \overline{h(t)} \qquad (2.4) \end{aligned}$$

where, $\overline{x(t)}$ and $\overline{h(t)}$ denotes the excitation signal and filter in the cepstral domain, respectively. It has been observed that $\overline{h(t)}$ always occupies the beginning of the cepstrum [28][51][62], which makes it very easy to extract [18].

In [18], Davis and Mermelstein introduced the Mel-frequency cepstral coefficients (MFCCs), which modifies the cepstrum by mapping linear acoustic frequencies to the perceptually shaped Mel frequency scale. MFCCs have been shown to be more discriminative for speaker recognition than the cepstral coefficients because they mimic the frequency response of the human ear which is less sensitive at higher frequencies [18][23][9]. Two other commonly used features are the ΔMFCC and ΔΔMFCC, which are obtained from the first derivative and second derivative of the MFCC, respectively. Combining these features tends to improve the performance of the MFCC feature since these features are largely uncorrelated and the ΔMFCC and ΔΔMFCC features are more resilient to channel effects [64].

Although throughout the years many features have been extracted from speech signals, MFCC are the most popular [79]. Another common feature is the linear prediction coefficients (LPC), which attempts to model the vocal tract configuration using an all-pole linear predictor function [9][75][2]. Linear prediction has been used by Tishby in [77] and Soong et al. in [73] among others. However, these features are very sensitive to additive noise and thus are not as effective as MFCCs.

2.2.3 Previous Works

There are two types of speaker recognition systems: text-dependent and text-independent systems. Text-dependent systems usually work by creating a template for the given a phrase(s) and make a decision based on how good the sample utterance matches the template. Text-independent systems are considered to be a more difficult problem, but have the major advantage that the system can prompt the user for any text. This eliminates the possibility of using recordings of valid users to defeat the system.

Speaker recognition techniques are usually based on statistical methods that can capture some information about the speaker's speaking style or acoustic characteristics. In [9], Campbell has complied an excellent list of previous efforts in speaker recognition and for each, he provides information on speech quality, features used, type of classifier used, and its performance. Here, only the present state of the technology is further discussed which includes hidden Markov Models (HMMs) and Gaussian mixture models (GMMs).

HMM based speaker recognition systems focus on modeling the temporal sequence of phonemes. Phonemes, which are the basic units of spoken language, can be used to produce any word(s) and hence, HMMs which are often used with MFCC and its derivatives have been effective for speaker recognition. In fact, most HMM based systems can achieve less than 1% classification error if used with relatively clean speech [11][14]. Additionally, because HMMs account for the temporal arrangements of phonemes, they have traditionally worked better with text-dependent systems. This is somewhat intuitive because the training set will not always represent the full gamut of phone combinations and temporal arrangements that are found in unconstrained speech [64][77].

GMM based speaker recognition systems have become the most popular method to date [62][23]. This is because GMM systems based on MFCCs and its derivatives have been found to discriminate between speaker-dependent acoustic phenomena that are present in speech. In fact, some of the GMM clusters have been found to be highly correlated with some phonemes [4]. Therefore, for text-independent recognition systems, GMMs perform better than HMMs because they make better use of the training data. Experiments by the National Institute of Standards and Technology (NIST) have shown that the temporal information accumulated and relied upon in HMMs are not significant for text-independent recognition [4] [58].

Another notable technique used in speaker recognition is the use of universal background models (UBM) with GMM based systems. In traditional GMM systems (known as cohort models), background speaker sets are used to calculate the log-likelihood ratio, which is the decision criteria for authentication. Typically, all other users become background speakers when calculating the log-likelihood for a particular target speaker. Research has shown that this method does not yield optimum performance unless target speaker specific background sets are used [67][62]. UBM based systems resolve this problem by generating one large dimensional super-GMM based on pooled training data from a representative portion of the speaker population. From this super-GMM, a model is generated for each user, usually based on the maximum posteriori (MAP) estimation which creates a target specific model derived from the background model. Aside from enhancing identification rates, UBMs also greatly simplify the log-likelihood calculation because the UBM can replace the background set for all users [62][83].

To complete this section, some effort is spent on describing channel effects because speech recognition systems are most effective if they can operate over standard telephone networks. The main problems with this scenario is the varying frequency responses of different telephone networks. In general cepstral mean normalization (CMN), RASTA processing, and Δ-MFCC coefficients have been found to be a good response to this challenge [64]. Additionally, because the telephone channel is a bandpass channel, a bandpass range of 300-3400 Hz is usually assumed in literature [83][62]. Further details on channel compensation techniques are presented in Section 4.2.

2.3 A Completely Ubiquitous Model

As already mentioned, new biometric technologies cannot be fully utilized unless they are fully integrated and accepted in society. A good example is the prevalence of credit cards and debit cards as forms of payment. Because of the ease of use, and relatively minor investment by business, almost anything can be bought without using any real money in the transaction.

Nowadays, geographical restrictions cannot be imposed on users and account holders. So, biometrics technology must achieve the same level of integration as credit cards before they can be fully utilized by businesses and governments alike. Truly ubiquitous biometric

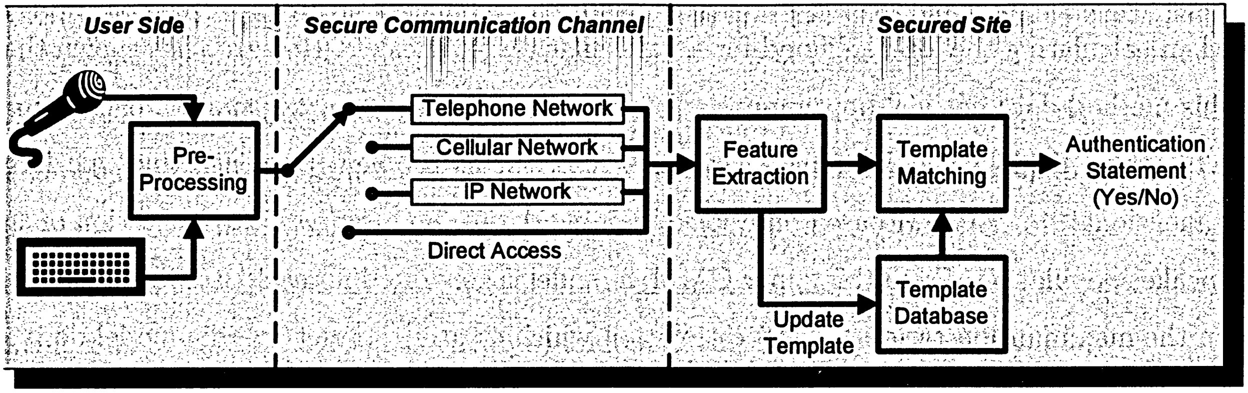


Figure 2.6: Block diagram illustrates how speech and keystroke biometrics can be used to perform remote and physical authentication.

systems must be able to perform three separate functions without requiring the end user to invest in any specific apparatus:

- Physical Authentication - Authenticate the identity of the user in designated spots such as inside a bank or outside a security gate.
- Telephone Authentication - Authenticate the identity of the user over existing telephone networks.
- Online Authentication - Authenticate the identity of the user over computer networks or the Internet.

These three conditions guarantee that no matter how the user attempts to gain access to an account, his or her identity can be verified. Figure 2.6 illustrates this concept.

The only practical solution for the above requirements is the combinations of speech and keystroke biometrics. Even though either of these technologies can be used ubiquitously[1], combining these two modalities can provide security for all remote access applications since

[1] Speaker recognition can be performed via a computer (with a microphone) and keystroke recognition can be performed on the number pad of a telephone. Therefore, it is possible to use either of these modalities regardless of how the user attempts to gain access.

telephones and computers are the only methods that can be used for remote access. Additionally, both of these technologies can provide security for physical authentication with much less cost and complexity that any other biometric technology.

Another reason for combining these two modalities is that each method is well suited for a particular task. Keystroke recognition is the preferred method for computer security due to its simplicity and versatility. For computer applications, keystroke recognition can be used as a static or continuous verifier and it can also be used in conjunction with the username and password authentication protocol for seamless integration. Similarly, speaker recognition is the preferred choice for telephone based applications since it can be seamlessly used during a regular conversation between the user and the service provider. Therefore, these technologies can be used separately to enhance security for all remote transaction over the telephone or through a computer.

2.3.1 Applications

The combination of speech and keystroke biometrics has many applications because many organization allow users to view or modify their account online or over the telephone. Some examples are banks, cellular companies, government agencies such as tax departments, retailers, email providers, and the list goes on. The applicability and versatility of these technologies for remote and physical authentication is virtually unbounded. In short, because these technologies are already integrated into the fabric of society, they are easy to use, cheap, and they can be quite effective.

Chapter 3

Keystroke Recognition

ALTHOUGH keystroke features have been shown to be a unique behavioral characteristic, there are several factors that can affect their accuracy. Unlike some physical biometrics such as fingerprints or iris patterns, keystroke patterns can vary from time to time depending on a person's state of mind among other factors, and therefore, steps must be taken to minimize these variabilities. The first half of this chapter provides some insight on what measures can be taken to optimize the accuracy of keystroke recognition systems and how to collect reliable data. In doing so, Section 3.1 proposes a protocol or guideline that suggests "best practices" for conducting experiments and reporting results.

The second half of this chapter presents a novel text-dependent keystroke verification scheme. The proposed method aims to remove the burden from the user by requiring a small number of samples during training and it provides significant security benefits over the standard username and password model. This method is completely hardware independent which makes it well suited for online or any other computer based authentication system. The details of this method are presented in Section 3.2.

3.1 Keystroke Protocol

Over the years, many different methods have been proposed for keystroke recognition. Although many of these interesting methods are promising, no superior technique has emerged. In fact, it is very difficult to compare different techniques because of major differences in the way these experiments have been performed.

In general, there are several differences that prohibit a direct comparison between different techniques. Some of these concerns are related to data collection procedures, number of samples collected, population size, and the use of non-standardized databases. Since there are no clear guidelines to indicate how experiments should be conducted, there has been a tendency to use convenient procedures. As a result, the reported results of some works, although good, are not convincing because of biases introduced by their data collection procedures. These biases can be clearly evident and therefore, do not allow for a fair comparison of competing techniques. In an attempt to resolve these concerns, Section 3.1.3 is devoted to methods that minimize errors or biases introduced by data collection techniques.

Another difference, although it is difficult to solve, is that there is no standard database for keystroke patterns. Consequently, all previous experiments have used different data sets with relatively small populations (typically 10-20 persons, but some have used as many as 63 persons). Because of the small populations used, there are some biases within different experiments stemming from the different data sets. For example, in almost all previous works, fixed database dependent thresholds have been used for the authentication decision which may not be appropriate for another group of users. To resolve this problem, Section 3.2.3 proposes an adaptive user-dependent threshold scheme that can be used with any classifier which helps to remove any bias stemming from the size of the group or its composition.

Although the concern regarding a non-standardized database still remains, this section proposes a new keystroke protocol that aims to level the field in terms of the way data is collected and the way results are reported. This will greatly improve the science of keystroke recognition because it will allow competing techniques to be compared fairly and the superior techniques to be highlighted. The proposed protocol also covers other areas, including how to design strong features by reducing the variability in keystroke patterns, statistical requirements, error reporting practices, and data acquisition requirements. Figure 3.1 illustrates the components of the proposed protocol and the details of each component are explained in detail throughout Section 3.1.

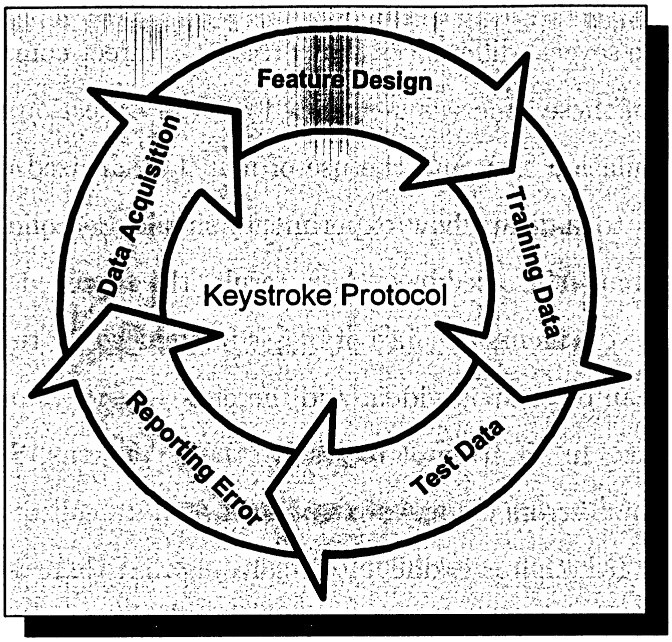


Figure 3.1: The components of the proposed keystroke protocol.

3.1.1 Designing Good Features

For keystroke identification, a robust feature pattern is one that is stable over repeated trials. To produce a stable feature pattern, the typist should be able to type the given text without much hesitation. Strings that require the typist to stop and think about the next letter or cause the typist to pause and search for a certain key, will result in unstable patterns. Although this is a particular problem for novice typists, it can be resolved by choosing familiar strings. For good typists, this is not a problem as they are familiar with the keyboard, and usually long pauses or other abnormalities are not likely. This issue is further discussed in Section 3.1.1. The use of temporal low-pass filters can also be useful for removing the above mentioned anomalies from keystroke patterns [38][42].

Type of String

For text-dependent keystroke recognition, research has shown that the best results are obtained when users type familiar text [36], such as their first and last names. These strings are intuitively easy to type for all people because they have been used for many years and therefore, a distinct pattern can be seen when users type their name. However, for increased security, any other secret or familiar string can be used. With minimal practice, users can quickly adapt to typing new strings and start to produce stable patterns [15].

There are two methods to ensure that the user is adequately familiar with the target string before the training session. One option is to allow the user to practice the string until he or she is comfortable with typing it, as was done in [10]. This method implicity relies on the users to indicate when they can produce a stable keystroke signature therefore, it may not produce the desired results in all cases. Another more structured technique is to request more than the required number of training samples and discard the first several samples. This procedure forces each user to practice the chosen string and results in a stable pattern before the actual training samples are recorded. In [15], it is recommended that the first 10-20 samples should be discarded. In this case, all of the collected samples should be without errors to ensure that the user is familiar with the target string and that he or she can reproduce it easily.

Another important criteria for choosing the best string is the set of allowable characters that can be used. Leggett and William [38], have shown that the use of all the lower case letters including the space key, produces the best results for user authentication. In this study, 11 different allowable zones from which characters could be chosen from was investigated. In [19], Magalhães et al. also concluded that the best results are achieved when the characters of a string are spread out across the keyboard. These findings are intuitive because it is not natural for the typist to be restricted on the keyboard. Another side effect of this practice is that it makes it more difficult for imposters to duplicate someone else's pattern because the pattern would be more complex.

Although these studies did not include the use of the *Shift* key for capital letters, command keys (*Alt*, *Ctrl*), or number keys, these keys are not expected to be beneficial for keystroke recognition because they can cause an interruption in typing patterns. Also, some keyboards do not differentiate between the left and right *Shift* key and the left and right command keys, which further complicates the matter by introducing hardware dependencies. As a result of these complications, the *Shift*, *Alt*, *Ctrl*, and number keys have been excluded from the set of possible characters in all previous works. The only exceptions to this have been a few studies that investigated the possibility of using keystroke recognition with number pads, such as those found on telephones or bank machines [13].

String Length

An important consideration for text-dependent recognition systems when selecting an appropriate string, is the number of characters. Research has shown that typing is a very structured process where a certain amount of text is stored in a short-term buffer somewhere in the brain and finger movements are planned accordingly before execution [71][70][76]. Furthermore, on an average this text buffer is 6 to 8 characters long [70] and when typing longer strings, users will exhibit a brief pause during which the text buffer is reloaded [16]. Therefore, 6 to 8 characters appears to be the optimal string length from a human perspective.

However, longer strings tend to produce better classification results because the classifier has more features to use and also forgeries become very difficult due to the complexity of the pattern. In previous works, it has been suggested that no less than 10 characters should be used for keystroke recognition [3][6]. At the same time, it will be said that no additional effort should be made to increase this minimum character requirement because it might be difficult or annoying for users to meet this requirement. Another major disadvantage of having a long string is that there will be an increased chance for typographical errors. Because these errors can occur in any combination and at any location(s) in the string, they cannot be modeled and therefore cannot be used for authentication. As string length increases, it becomes more difficult for the average typist to produce the string without errors and since these errors must be ignored, the failure to acquire rate will increase. Long string lengths would also pose a strict requirement if the user's chosen string does not meet the minimum character requirement.

All of these factors could have a negative impact on recognition performance if long strings are imposed. Based on experimental results and an objective study of previous works, a minimum string length of 10 characters is recommended for text-dependent keystroke recognition systems. For text-independent keystroke recognition systems much more data is required. In fact, for text-independent tasks such a continuous monitoring, the training data should be designed to include multiple instances of all two letter combination (digraphs) and several more for commonly occurring digraphs. Text-independent systems could also work by monitoring selected digraphs that are commonly used for the given application.

3.1.2 Collecting Training Data

During authentication, the user's test pattern will be compared to this reference pattern (model) before a decision is made. All the current recognition schemes operate in this manner and therefore, a good model will help to improve both the *false rejection rate* (FRR) and the *false acceptance rate* (FAR).

In many modeling techniques, lower order statistical moments such as feature mean and variance play an important role. Therefore, it is important to have enough training data before generating a model for a user. Obviously, more training data would benefit every technique, but this requirement must be balanced with the ability of the user to provide this data in a short amount of time. Commercially successful techniques will be those that do not require large amounts of training data.

To generate a representative model for a given user, no less that 10-20 training samples should be used for text-dependent systems. From all available published works, it appears that most authors have used at least 10 training samples; recognizing the fact that more samples are needed to make inferences about the statistical characteristics of the data. For example, the mean (μ) of a normally distributed random variable, estimated using M samples can be said with 95% confidence to lie in the range given by [48]:

$$\mu \pm 2.262\sqrt{\frac{\sigma^2}{M}} \tag{3.1}$$

where, σ^2 is the variance of the random variable. Since μ is relatively large (to the order of 100ms), an accurate estimate of mean can be obtained if M is comparable to σ^2. Similarly, an accurate estimate of the feature mean will result in an accurate estimate of feature variance since the variance calculation is based only on the mean and the number of sample from the random variable (X), as given in the equation below.

$$\sigma^2 = \frac{1}{M-1}\sum_{i=1}^{M}(x_i - \mu)^2 \tag{3.2}$$

Parameter estimation accuracy can be improved by using self updating models. If updates are performed frequently or after every successful authentication, the model will very quickly be populated with large numbers of sample data and it will become more accurate. Also, models that are frequently updated have the added benefit of being able to keep track of a user's day-to-day and long term typing variations.

Collecting all of the training data in one session can also introduce unnecessary biases in experiments with text-dependent systems. At some point, after typing the same string repeatedly, users will often lose their natural typing rhythm and begin to exhibit a "machine-like" rhythm. Empirical observations indicate that when this happens, a user's typing pattern will become dissimilar with his or her natural pattern. To combat this problem during training, a short pause of at least 5-10 seconds should be allowed between each sample. This short pause will distract the user from the training task and will place the user back into a neutral mental state. Also, a short pause will induce movement or other activity that will put the user back into a physically neutral position with respect to the keyboard, hand position, and arm position. In short, this procedure will minimize any side effects from repeatedly typing the same string by simulating the users action when he or she is attempting to log into the computer in a normal situation.

In general, text-dependent keystroke recognition systems are much more popular than text-independent systems because data collection is much more practical for the former. Text-independent systems would require thousands of characters to be typed for each of the training and authentication sessions in order to identify the user [26][78].

3.1.3 Collecting Test Data

Keystroke recognition is a behavioral biometric and as such, it is affected by a person's state of mind and physical condition. Certainly, a happy person and a sad person do not have the same neurological state and therefore, cannot be expected to produce the same behavioral characteristics. Specifically, illness, fatigue, stress, emotional state, position of keyboard relative to the body, environmental conditions such as lighting or noise, and many other factors can effect behavioral characteristics, motor responses (i.e. finger or arm movements), and reaction times among other side effects. These changes in behavioral characteristics are natural and will affect typing patterns.

Since keystroke patterns are affected by the psychological and physical state of individuals, their natural variability cannot be captured in a few sessions or by a few samples. Thus, the effectiveness of keystroke recognition systems cannot be clearly understood unless this variability is captured through many authentication sessions. In the benchmark study on evaluation methods for biometric systems, Mansfield and Wayman [44], make the same recommendation about removing biases in the collected data by spreading the data collection over different sessions and under different conditions. Moreover, it is well known that randomized data collection will average out the effects of uncorrelated noise sources introduced by the subject or by the instrumentation [31]. In keystroke recognition experiments, these unbiased conditions can only be achieved if data is collected over many sessions, over some period of time, and in uncontrolled and unsupervised conditions.

Unfortunately, in many previous works, authors have tended to collect the test data directly after the training session [10][12][15][36]. In some works [40][82][43], the data collection procedures and time lines are not even mentioned but since the number of samples collected are very small, it is reasonable to assume that they were collected in one session; possibly after the training session. Yet others have used two sessions: one for training and one for authentication [8][78]. In all these cases, enhanced classification rates can be expected because the test data is highly correlated with the training data. In other words, these data collection schemes cannot reliably capture the natural variations in the keystroke patterns because the effects of behavioral characteristics on typing patterns cannot be captured in a few sessions.

In fact, many authentication sessions are required before reliable results can be produced. To this end, a data collection protocol similar to that found in [33][47][6][32] is recommended. In all of these cases, the data has been collected over a period of several weeks or several months and at the convenience of the user (in an unsupervised manner). This type of experimental conditions closely simulates the way that the proposed authentication system would be used in practice. Therefore, much more convincing results are produced because the system would be exposed to a large range of the psychological and physical states of the users and the number of authentication attempts would tend to be high.

As more samples are collected, they can be used to update the user's model. These updates ensure that the model parameters can better represent the user. This is one of the best ways to increase the training set and ensure good performance over time.

3.1.4 Measuring System Performance

Measuring the performance of any system is very important because it helps potential users (or buyers) of the technology to evaluate it and compare it with other systems. For any biometric system the most important performance indicators are the FAR, FRR, and *equal error rate* (EER). The EER can be used as a single indicator of the system's performance since it indicates the amount of error when the FAR equals the FRR [55]. In previous works, authors have tended to provide singular values for the FAR and the FRR, thereby giving the reader very little information about the overall performance of the proposed systems. A good summary about error rates and system performance can be provided through the use of a receiver operating characteristics (ROC) curve [44]. ROC curves are a plot of the systems FAR vs. FRR over a range of 0% to 100%, which allows potential users to adjust the system's performance at a given level. And for a comparison of several different systems, detection error trade-off (DET) curves can be even more useful [44]. DET and ROC curves differ only in that DET curves are plotted on a log-log scale which clearly shows the differences in similarly performing systems.

When reporting the classification performance, it is important to base the results on a reliable number of tests and the correct type of tests. Some important concerns regarding these issues are discussed below.

"Rule of 30"

The "Rule of 30" states that "To be 90% confident that the true error rate is within ±30% of the observed error rate, there must be at least 30 errors" [21]. This rule is intended to be used when the trials are independent and when there is only one source of variability in the experiment; so it is overoptimistic in many practical situations [44]. However, it is instructive in cases where authors claim very low error rates with insufficient number of tests. In these cases, the actual error rates may be significantly higher with more users or more tests. For convincing results, the "Rule of 30" is a minimum requirement for reporting reliable error rates.

"Rule of 3"

The "Rule of 3" also provides an important indicator of the accuracy for the reported results. This rule provides the minimum error rate (p) that is achievable for a given set of P independently identically distributed (IID) comparisons, and is given below [44].

$$\begin{aligned} p &\approx 3/P && \textit{For a 95\% confidence level} \\ p &\approx 2/P && \textit{For a 90\% confidence level} \end{aligned}$$

The "Rule of 3" implies that 300 authentication tests are required to have a 95% confidence level that the minimum achievable error is 1% (or 200 authentication tests for a 90% confidence level). Therefore, if very low error rates are achieved, the number of tests performed should reflect the requirements imposed by "Rule of 3".

Imposter Tests

When calculating FAR, Mansfield and Wayman [44], make an important distinction between *zero-effort imposter attempts* and *active imposter attempts.* The former is the FAR when imposters make no effort to copy a legitimate user's keystroke pattern, while the latter is the FAR when the imposters try to copy a legitimate user's keystroke pattern. For such biometrics as fingerprint detection or iris pattern recognition this distinction is not very important because forgeries are difficult to produce and require specialized equipment. However, for biometrics such as keystroke patterns, this would be a useful indicator of performance and robustness of the proposed system in a practical setting.

Achievable Goals

Ideally, a biometric system should yield a 0% error rate. This is an ambitious goal and surely cannot be achieved with any practical system, and those that claim such error rates simply have not enrolled enough users in the system to make such a claim. Therefore, an achievable or acceptable target must be defined. The author agrees with Joyce and Gupta [36] that an acceptable FRR should be below 5% and a acceptable FAR should be below 1%.

A slightly high FRR is not a significant concern if the FAR is very low because this is only a nuisance and the user can always try authentication again. If the authentication sessions are assumed independent, the chances of a valid user getting rejected twice in a row is 0.25% for a 5% FRR; this is very acceptable. A more important performance measure is how many times imposters can gain access to the system. Generally, a 1% FAR is well regarded since it implies that an invalid user will be accepted only once in every 100 attempts. For keystroke recognition applications, this means that *if* an imposter is aware of a valid user's password, only then the quoted FAR applies. This double layer of protection is major advantage of keystroke recognition systems that cannot be expressed in terms of the FAR.

3.1.5 Data Acquisition

Mahar [42] has shown that on an average the DDKL feature exhibits timing characteristics between 96 to 825 milli-seconds (ms) depending on the user's typing ability. From experimental data collected and the works of others, it can be shown that the KD and the UUKL features also exhibit similar timing characteristics. Therefore, when measuring keystroke timing characteristics, a minimum timing accuracy of 1ms should be used. In previous works, timing resolutions of 0.1 ms to 1 ms have been used.

Timing information should be collected automatically from a specially designed software module. To guarantee timing accuracy, the data acquisition software should monitor keyboard activity directly from the appropriate interrupt handler on the host computer. Additionally, since all computers have a much finer timing resolution than required and all keyboards function with similar technical specifications, the timing acquisition software should be completely hardware independent.

In some applications such as online authentication, the user's computer should only be responsible for collecting the timing data, while the service provider should be responsible for processing the data and providing a decision on it over a secure connection. This process ensures maximum security for the collected biometric signatures.

3.1.6 Summary of Keystroke Protocol

To summarize, user identification through keystroke patterns requires a stable pattern. To obtain stable patterns, to gather detailed and accurate results, and to reduce biases in the experiment, the following guidelines are recommended. This guideline also addresses some persistent issues in previous works and makes some recommendations on how to overcome them. The basis for these recommendations are presented throughout Section 3.1.

- *Target String:* The target string should be familiar to the user and easy to type. Otherwise, the user should be given a chance to practice typing it before the training sessions. The person's name is usually a good choice, but for more protection a secret string can be chosen.

- *String Length:* The minimum string length that should be used is 6 to 8 characters. However, classification performance increases with string length so a slightly longer string of 10 characters is recommended for text-dependent systems. Longer strings may degrade performance because of an increased potential for typographical errors and difficulties with finding appropriate strings; see Section 3.1.1.

- *Training:* Each user should enroll in the system by typing their chosen string at least 10 -20 times with short pauses of 5-10 second in between each sample. The number of samples are important because it will help to better train the classifier and the pauses are required to collect samples which are independent from each other.

- *Data Collection:* To capture all of the variabilities in keystroke patterns which may be caused by fatigue, illness, stress, emotional state, etc., data should be collected over a period of several weeks or several months in unsupervised conditions. Data collected from a few controlled sessions is not likely to capture the natural variabilities and will result in overly optimistic or unreliable classification results.

- *Model Updates:* User models should be updated frequently or every time the user is successfully recognized. This will help keep track of any changes in a user's pattern over time and will also capture a user's natural variations as the number of samples increase.

- *Authentication Scheme:* A two-stage authentication scheme can be employed to improve FRRs. That is to say, if the user is rejected on the first attempt, he or she should be given another chance at authentication, possibly with relaxed criteria. This scheme can significantly improve the FRR without a significant effect on the FAR. See Section 3.2.4 for details and Section 3.3 for experimental results.

- *Performance Measures:* For others to be able to evaluate the overall performance of the system and obtain the EER, an ROC curve is required; the best FAR and FRR may be obtained from this curve. When comparing several different systems a DET curve may also be useful. Also, the reported results should be based on a sufficient number of samples in accordance with the "Rule of 30" and the "Rule of 3". Lastly, FARs should be obtained with *zero effort imposter* attempts and *active imposter* attempts to demonstrate the robustness of the technique.

- *Data Acquisition:* A hardware independent user interface should be built in software that can obtain keystroke timing information with of 0.1-1 ms accuracy for best results. For online applications, the collected biometric samples should be processed on a server computer and communication should be done over a secure link to minimize the chances that the biometric data will be intercepted or captured by criminals.

By following the recommendations outlined in this protocol, the data collection would adhere to a very high standard and the reported results would be accurate and convincing for a reasonably sized population. Furthermore, this protocol will help to reduce authentication errors by providing insight on reliable practices and how to design good features. The recommendation proposed here, if followed, will also be useful for comparing future experiments on a level playing field.

3.2 Keystroke Verification System

This section presents a novel method for keystroke verification. As mentioned earlier, verification is the process of validating a claimed identity. The proposed method is a text-dependent verification system based on GMMs that has several desirable characteristics. These characteristics are: training based on small number of samples, an updating mechanism for reliable long term performance, a two stage authentication scheme, a user-dependent least-biased and adaptive authentication threshold, and it is completely hardware independent. The block diagram of the proposed system is given in Figure 3.2.

As discussed in Section 2.1.1, keystroke patterns are hardware independent. This claim can be justified because research has shown that finger movements during typing are planned before their execution. As a result, the variations in the type or size of keyboards does not sig-

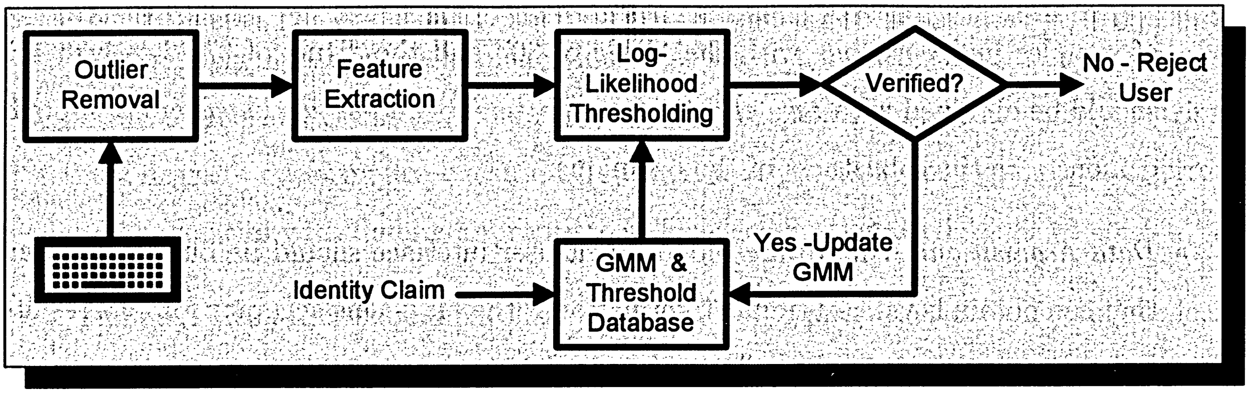


Figure 3.2: Block diagram of the proposed keystroke verification system.

nificantly effect this behavioral biometric. Therefore, the proposed technique can be widely used with any computer system.

A text-dependent verification system is proposed simply because it is a far more practical than a text-independent system. Previous works with text-independent systems have required several hundred words to be typed by the user during training and the authentication sessions [26][78]. This is not practical because of the time and effort that is required to perform a single test. On the other hand, text-dependent systems have been shown to work based on the pattern obtained from one string. This is well suited for all computer based applications, particularly for the username and password security scheme.

A verification system was chosen because it can be easily incorporated with the widely used username and password model. In this case, because the identity of the user is known from the username, there is no need to implement an identification system. Therefore, it is only necessary to verify the claimed identity based on the secret password and the keystroke pattern.

Since there can be variability in keystroke patterns, statistical or machine learning methods are needed for this biometric. In Section 2.1.3 a number of possibilities were discussed from previous works. Machine learning methods such as ANNs and SVMs have not produced favorable results mainly because large training sets are required or because of high computational complexity. Of the many statistical approaches proposed, the most successful have been distance based classifiers and classifiers based on low order statistical moments. There have not been many techniques that attempt to explicitly estimate the distribution of keystroke data.

Two effective methods for estimating the distribution of a random event are HMMs and GMMs. However, a true HMM cannot be used for text-dependent keystroke recognition since the temporal arrangement of keystrokes for a given string is known apriori. Therefore, the transition matrix for all states in the HMM would have only one none-zero entry corresponding to the only possible outcome (for a text-dependent system). This leads to a one state continuous HMM which is equivalent to a GMM [45] in a text-dependent system.

In the past, attempts to model keystroke distributions with one Gaussian function has not been entirely successful [47][26], see Section 2.1.3. Furthermore, it is unlikely that a single Gaussian distribution can be effective on a long term basis or for a large number of users since there is a lot of variability associated with behavioral and environmental changes. Since modeling keystroke patterns with a single Gaussian has had some success, GMMs will be even more effective because they can create a multi-modal distribution [22][60] for the keystroke patterns. This is advantageous because GMMs can produce a user specific distribution for a given string, since keystroke patterns do not fit any known distribution. This has been shown to be effective by the author's previous works [33]. As a result, GMMs were chosen as the statistical tool in this work.

3.2.1 Feature Selection

To have a robust user recognition system, a set of robust features are required. These features should have user-dependent characteristics and should be easy to capture. As described in Section 2.1.2, keystroke features are extracted from the timing sequence produced during typing. The two most popular features in previous works have been the KD feature followed by the DDKL feature. These two features were be used in this work because they have proven to be effective. Another latency feature, the UUKL feature, was used for the first time in this work. Results and analysis on this feature compared to other features is presented in Section 3.3.2.

Using multiple features is expected to improve the overall classification results because users will be subjected to multiple tests which will make it harder for both valid users and imposters to pass the authentication tests. Therefore, the FAR is expected to decrease and the FRR is expected to increase. However, the effect on the FRR should be much smaller because valid users are expected to pass these tests more often than imposters.

When using multiple features, a voting rule should be used to combine the decision from each feature into one authentication statement. Some popular voting methods are:

- Unanimity - The unanimity rule requires that each voting member accepts the user, otherwise the user is rejected.
- Majority - The majority rule selects the most popular decision among the voting members.
- Normalization - This rule normalizes the decision from each voting member so that they can be combined into one score. This score is then compared to a predefined threshold; so another threshold must be defined.

A good review of some popular methods for combining different scores is given in [32]. In this work the unanimity rule is chosen because this rule minimizes the FAR since there is good chance that imposters will fail the authentication test with at least one feature. This should only effect the FAR significantly since valid users are more likely to pass all authentication tests.

3.2.2 Training and GMM Estimation

Before the verification process can occur, each user must train the system by providing a number of samples. In biometrics literature, this training process is also commonly known as the enrollment process. Based on these samples, a GMM is created for each of the three feature (KD, DDKL, and UUKL) which are used for future authentications tests. To increase the accuracy of the model, each time the user is successfully authenticated, the GMM is re-estimated by adding the new sample to the previously collected samples. This process ensures that the model remaind accurate and current with a user's keystroke pattern.

In order to estimate the GMM , the expectation maximization (EM) algorithm was used [20]. The EM algorithm is commonly used to derive the parameters (Λ) for GMMs, which are given below.

$$\Lambda = \{w_i, \vec{\mu_i}, \Sigma_i\}, \quad i = 1, ..., K \tag{3.3}$$

where, the mean vector ($\vec{\mu_i}$), the covariance matrix (Σ_i), mixture weights (w_i), and number of components (K) completely describe the GMM. A brief review of GMMs is presented in Appendix A.

The EM algorithm estimates the parameters of Equation 3.3 using a two step iterative process. There are two factors that affect the estimation of GMMs: the number of components in the mixture and the type of covariance matrix used. Generally, full covariance matrices are recommended because they can capture the correlation between the components of the feature vectors (training data) and therefore, these matrices provide a good description of dependencies within the data. Nonetheless, in the proposed method, diagonal covariance matrices are used since the components of keystroke feature vectors are not expected to be highly correlated with each other. Even with statistically dependent components, a linear combination of diagonal covariance matrices can model the correlation between the data [64]. Also, diagonal covariance matrices are preferred because they are much more computationally efficient and the performance of a K component GMM with full covariance matrices can be achieved with a larger order model having diagonal covariance matrices [61][64].

The maximum number of components for each GMM was set to 8. However, the actual number of components used was estimated using the Rissanen minimum description length (MDL) method [66] because with the initial amount of training data it is not possible to estimate a N-dimensional ($N \geq 9$ is the length of the feature vectors) 8-component GMM. Also, MDL has been shown to be a good estimator of model order for GMMs, performing even better than the Akaike Information Criteria (AIC) [69] when estimating low dimensional models with a small number of samples [46].

3.2.3 Authentication Threshold

Upon verification, the user is required to provide a sample string and an identify claim (i.e. a username and a password). Since the username identifies the user, the sample string (the password) is compared with the claimed user's model in order to make a decision about the identity of the user. To calculate the likelihood ($\mathcal{L}\{.\}$) that the provided sample ($\vec{x}$) belongs to the claimed user's model ($\mathbf{\Lambda_{claimed}}$), Equation 3.4 is used. This result is then compared with the user's threshold ($\Gamma_{claimed}$) before access is granted or denied.

$$\mathcal{L}\left\{\vec{x}\right\} = p(\vec{x}|\mathbf{\Lambda_{claimed}}) = \sum_{i=1}^{K} w_i b_i(\vec{x}) \tag{3.4}$$

where, $b_i(.)$ is a L-dimensional Gaussian function given in Appendix A, K is the number of components, and w_i is the weight parameter for each component.

The determination of the appropriate threshold is an important consideration because it significantly impacts the system's performance. The threshold should be robust so that imposters can be easily detected and the threshold should also be adaptive so that it can track changes in the model. In most previous works, one fixed threshold selected arbitrarily or experimentally has been used for all users during authentication. This results in a database dependent threshold which cannot be optimal for all users or for other groups of users.

To create a robust and adaptive threshold, the Leave-One-Out-Method (LOOM) is proposed [25]. The LOOM can be used to calculate a range of possible thresholds from which an appropriate threshold can be chosen. Additionally, the LOOM provides a completely user-dependent threshold, which is based on the user's previous samples. Therefore, unlike fixed thresholds, the LOOM based threshold can adapt to a user's changes and also it is not database dependent.

A brief description of the LOOM is as follows: for R feature vectors, $R-1$ vectors are used to create a model and the last vector is used to test the likelihood that it belongs to that model, using Equation 3.4. This test can be performed R times, where each time a different vector is used to test the model. The final results of the LOOM produces R likelihood measures and can be expressed by

$$\mathcal{L}_j = \log\{p(\vec{x_j}|\mathbf{\Lambda_{R-1}})\}, \quad j = 1, 2, .., R \tag{3.5}$$

where, $\mathbf{\Lambda_{R-1}}$ is a GMM that has been trained with $R-1$ vectors not including the j^{th} vector and $\vec{x_j}$ is the test vector.

From these R likelihood values, the minimum value that falls within the range of 2.5 standard deviations away from the mean is set as the model threshold (Γ_{gmm}), as given below:

$$\Gamma_{gmm} = \underset{\forall j}{min}\left\{\mathcal{L}_j \,|\, (\mathcal{L}_j - \overline{\mathcal{L}}) < 2.5\sigma_{\mathcal{L}}\right\} \tag{3.6}$$

where, $\overline{\mathcal{L}}$ is the mean and $\sigma_{\mathcal{L}}$ is the standard deviation of the R likelihood values obtained from the LOOM. A standard deviation of 2.5 was chosen as the cut-off point because the aim is to remove extreme outliers. Without such a precaution, the threshold may occasionally be set too low, resulting in increased FARs.

The model generation and threshold calculation procedures are repeated every time a user is authenticated by adding the new sample to the model data and re-estimating its parameters. This way, the model remains accurate and the threshold value is adaptive and can change with the user over time. The LOOM was chosen because it has been shown to provide the least biased estimate for small databases [25]. Therefore, the model thresholds are optimal given the size of the sample database. This is particularly important when a new user is enrolled in the system because there are limited samples to create a threshold from. Of course, as the model grows, the LOOM will provide even better estimates of the threshold.

3.2.4 Authentication Scheme

A two stage authentication scheme was used for the experiments in this work. In such a scheme, both imposters and valid users are given two chances at authentication and therefore, they both have an increased chance of being accepted by the system. In previous works, the author has shown that this scheme produces significantly lower FRRs without a significant effect of the FARs [33]. Therefore, two stage authentication is very desirable and more detailed analysis of this is provided in Section 3.3.2.

Two stage authentication can function in two ways: either the user is requested to enter two samples at once and the system chooses the best sample automatically or the samples are requested as needed (i.e. the second sample is needed only if the user fails on the first attempt). The latter option is chosen here so that the effort required by the user is minimized. For improved FRRs, it is even possible to relax the authentication criteria on the second attempt, although that is not done here.

During the authentication session the threshold obtained by the LOOM (Γ_{gmm}) is compared to the likelihood value ($\mathcal{L}\{.\}$) obtained from the user's sample ($\vec{x}$). Access is granted if the user's likelihood value is greater than the threshold, as shown below.

$$\mathcal{L}\{\vec{x}\} = p(\vec{x}|\Lambda_{\textbf{claimed}}) \geq \Gamma_{gmm} \quad \textit{Acceptance Criteria} \tag{3.7}$$

However, using the threshold obtained from the LOOM results in a fairly constant system performance which may not be desired. By varying Γ_{gmm}, system performance can be set to any level of FAR or FRR. A good way to modify the threshold is to use its statistical properties rather than using some arbitrary value, as shown below.

$$\Gamma'_{gmm} = \Gamma_{gmm} \pm k\, \sigma_{\Gamma_{gmm}} \tag{3.8}$$

where, k is any real number, Γ'_{gmm} denotes the modified threshold, and $\sigma_{\Gamma_{gmm}}$ is the standard deviation of previous thresholds. Using Equation 3.8, the modified threshold still retains the user specific characteristics which are inherent in the LOOM. Equation 3.8 was also used as the basis for creating the ROC and DET curves in Section 3.3.2.

3.3 Experimental Setup and Results

This section presents the experimental conditions as well as the results. Section 3.3.1 explains the details of the experimental procedures and the data collection procedures, while Section 3.3.2 provides a detailed discussion on the results.

3.3.1 Experimental Conditions

The experimental conditions are very important as they determine how much bias is introduced into the collected data and the results. To minimize these biases, all of the recommendations of Section 3.1 were implemented.

The results are based on data collected from 41 subjects (30 males and 11 females) over a period of 4 weeks. The users ranged in age from 18 to 65 years old, with an average age of 30.1. Typing proficiency was not required and in fact the group consisted of users with varying typing abilities, including several two-finger typists.

Each user was given a specially designed application for keystroke pattern authentication named *KbApp*; see Appendix B for details. They were instructed to install the application on their home or office computers and provide samples as often as possible. These conditions resulted in a complectly unsupervised experiment which closely simulated a practical situation. This resulted in 1156 self authentications which can be averaged to 4 authentication tests per user per day for 4 weeks. Therefore, because the users provided samples at many different sessions, it is likely that these samples captured a wide range of the variability that are present in their keystroke patterns.

Each user was instructed to use their full name as the authentication string. This is convenient because everyone is familiar with typing their name and because the identity claim and the keystroke pattern can come from the same string. Otherwise, the users would be required to enter two separate strings since the proposed method is a verification system.

In the training session, up to 30 samples of each user's full name was collected, with a short 5 second pause between each sample. This short pause is introduced so that the collected samples are provided independently from each other, as discussed in Section 3.1.2. This process is approximately between 5-10 minutes long per user. Since three different features were being analyzed, the training data was used to create three different GMMs. Specifically, a GMM of up to eight mixtures was created for the KD feature, the DDKL feature, and the UUKL feature.

Feature	FAR (%)	FRR (%)	EER (%)
KD	21.9	2.9	11.5
DDKL	27.6	2.1	11.0
UUKL	25.4	2.0	12.5
UUKL & DDKL	16.3	3.2	9.9
KD & DDKL	7.4	5.5	6.7
KD & UUKL	6.8	5.2	6.2
KD & UUKL & DDKL	5.1	6.5	5.8

Table 3.1: Experimental results for all feature combinations (based on 1156 self test and 1505 imposter tests).

3.3.2 Results & Discussions

The keystroke authentication method analyzed in this work is based on GMMs and the LOOM. The advantage of the LOOM is that it provides a dynamic thresholding scheme for each user based on his or her previous samples. This resolves the common problem of using a database dependent or arbitrarily selected fixed threshold for all users. Also, the LOOM provides the least biased threshold estimate for small databases (a problem faced during initial usage after enrollment) and will continue to perform well for larger databases.

GMMs were used to create statistical models for three features (KD, DDKL, and UUKL) extracted from each user's keystroke pattern. During each authentication test, 7 authentication decisions were made about the provided sample: one decision based on each feature individually, one decision based on all two-feature combinations, and one decision based on all three features together. For combining the decisions of multiple features into one authentication statement the unanimity rule was used. The results of these tests are shown in Table. 3.1. The results are based on the initial training samples collected from each of the 41 users followed by a total of 1156 self authentications and 1505 zero-effort imposter tests. Unfortunately, active imposters tests could not be obtained.

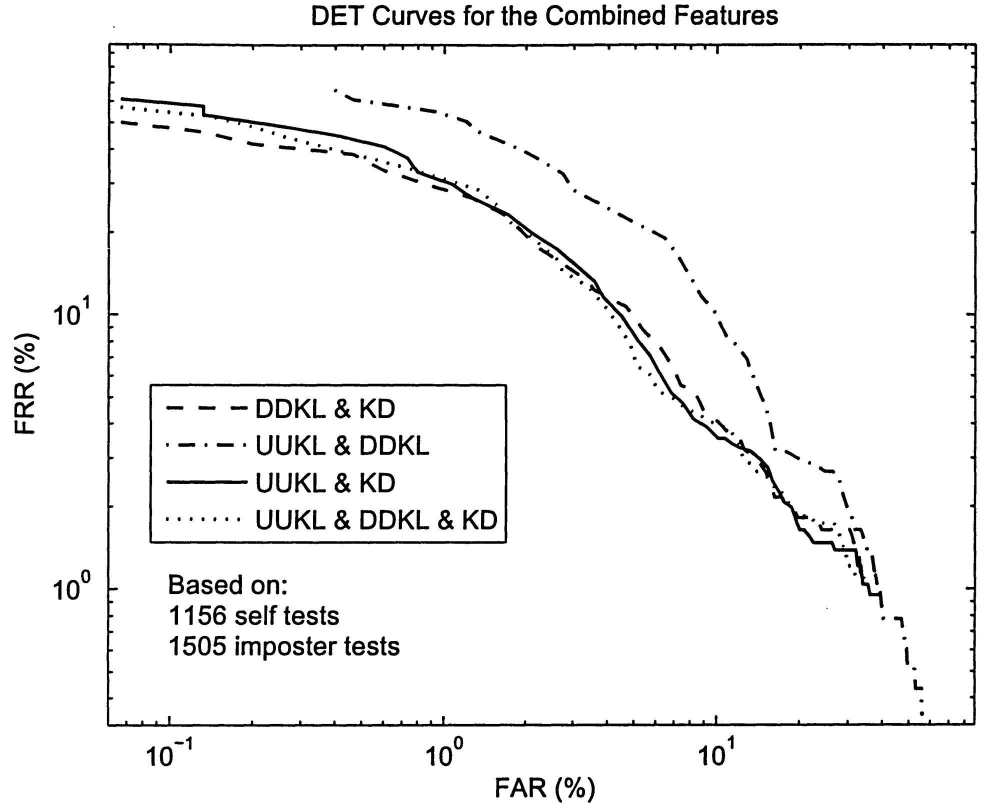


Figure 3.3: Plots of DET curves for the combined features in the two stage authentication system.

a) Feature Performance

By examining different feature combinations, it is possible to show which features work best with the proposed method. Data from Table 3.1 confirms the earlier logic that using multiple features results in better overall performance through a significant reduction in the FAR without a proportional increase in the FRR. The overall performance trend for each feature combination can also be seen from the DET curve in Figure 3.3. The best authentication accuracy was achieved using all features together, providing a FAR of 5.1%, a FRR of 6.5%, and a EER of 5.8%.

All three features have poor performance characteristics when used alone for authentication, as can be seen from Table 3.1 and also from Figure 3.4. However, the KD feature has

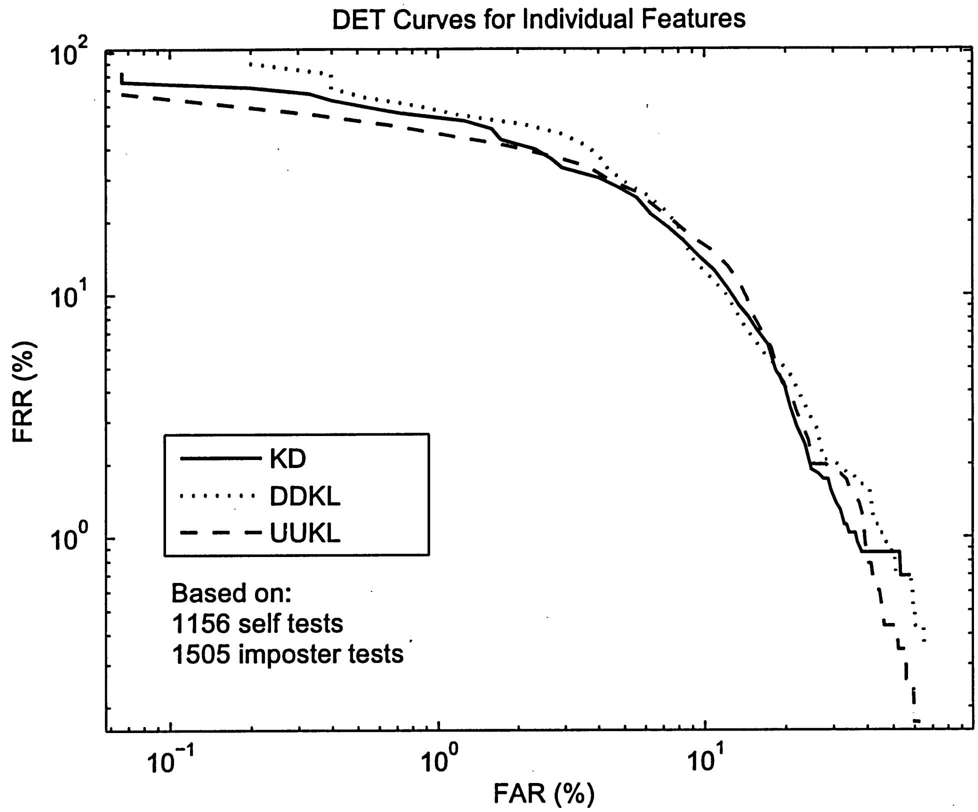


Figure 3.4: Plots of DET curves for individual features in the two stage authentication system.

a much better overall performance than either of the latency features (UUKL and DDKL). It can be seen from Table 3.1 that the KD feature has a much better false acceptance performance and a comparable false rejection performance when compared with either of the latency features. Therefore, for good overall performance, the KD feature should be used in conjunction with one or more of the latency features. In general the latency features are not as discriminating as the KD feature and produce higher FARs.

Among the latency features, the UUKL feature tends to perform slightly better than the standard DDKL feature especially for low FAR cases, see Table 3.1 or Figure 3.4. This is also true for the combined features, since the UUKL & KD feature combination outperforms the DDKL & KD feature combination, see Table 3.1 or Figure 3.3.

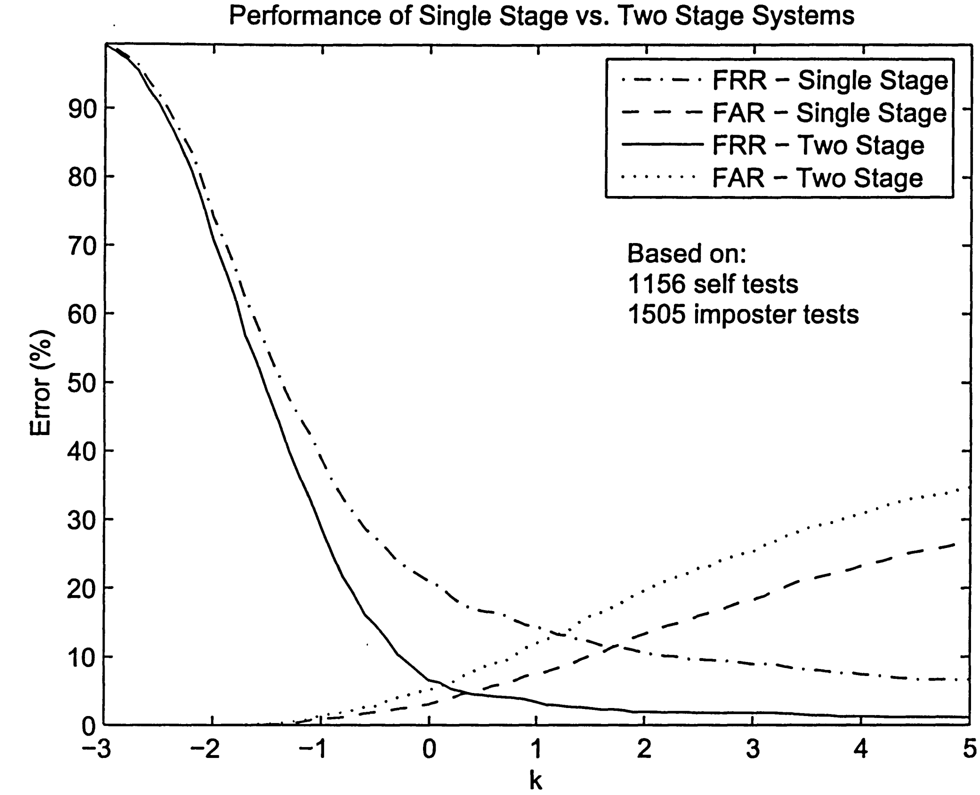


Figure 3.5: Plots of FAR and FRR versus k for a single stage and a two stage authentication system (for the KD & UUKL & DDKL feature combination). See Equation 3.8 for the relationship of k and the system threshold.

b) Two Stage Authentication

In all cases, the two stage authentication scheme described in Section 3.2.4 was used to improve the FRR. Figure 3.5 shows the effect of the two stage authentication for the GMM system when authentication is based on the combination of all three features (i.e. KD & UUKL & DDKL). It can be seen from Figure 3.5 that the change in FRR is much more than the change in the FAR in a two stage system compared with a single stage system. Especially in the areas where the system's performance is desirable (i.e. areas around $k = 0$ where the acceptance threshold is set) this effect is very pronounced. From Figure 3.5 it can be seen that the FRR was reduced by 14.5% while FAR was increased by only 2.1% on

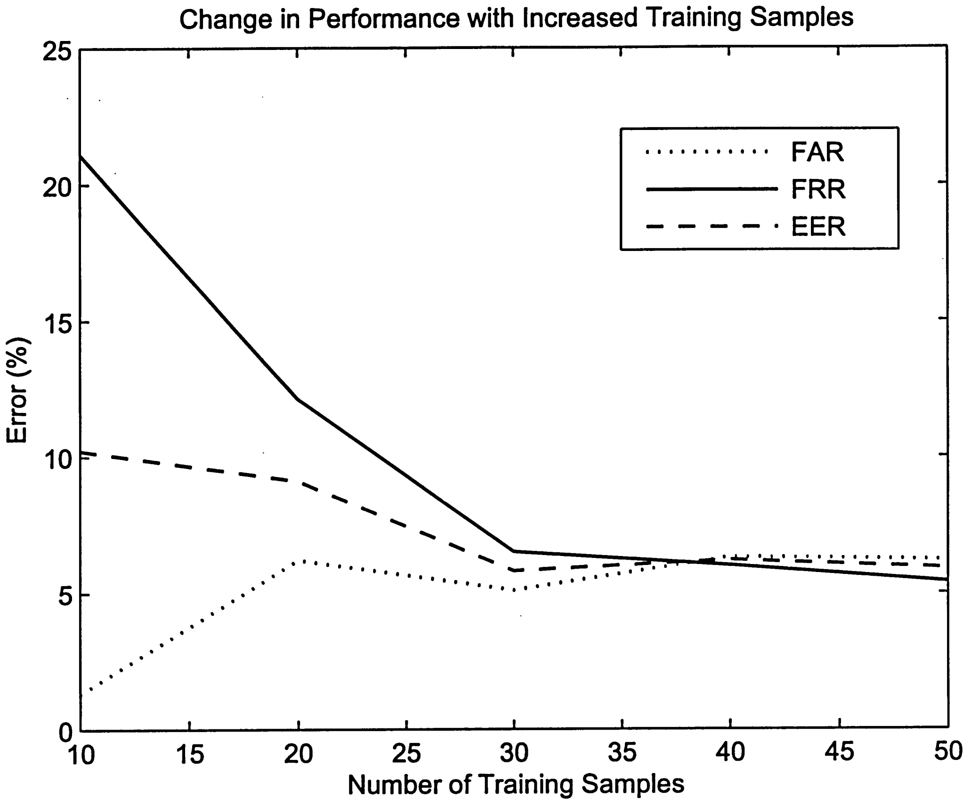


Figure 3.6: Number of training samples vs. classification performance for the combined feature set (KD & UUKL & DDKL).

the second attempt at $k = 0$, where $k = 0$ indicates the system threshold determined by the LOOM. This is a very desirable characteristics for the authentication system because it appears the best improvement occurs near the user specific threshold. This has led to marked improvement in the overall performance of the classifier. Similar results were observed for other feature combinations.

c) Effect of Training Samples

The number of training samples can also have a significant effect on the system's performance. Figure 3.6 shows that as training samples increase, the EER and FRR decrease. This is consistent with expectations because as the number of training samples increase, the model captures more user variability and performs better. Figure 3.6 also shows a sharp increase in the FAR between 10 training samples and 20 training samples, followed by minor fluctuations. This unexpected result is explained by larger model variances which were observed as the number of training samples increased. In other words, as the number of training samples is increased, the variance of each feature grows, thus allowing more imposters to be accepted because the model is more spread out. This problem can be minimized by using better temporal filtering methods. In this work, a fixed 500ms temporal filter was used to remove extreme outlier values from the keystroke patterns. A better approach would be to use a specific filter for each user based on feature variance and mean. For example, for a good typist that exhibits small keystroke times, a 500ms filter is too large and leads to large model variances and for novice typist that exhibit very large keystroke times a 500ms filter is too small and leads to a strict model. Therefore, temporal filtering methods should be user specific for best results.

d) Comparison of Results

It is difficult to compare these results with a great deal of previous works, since differences in experimental conditions do not allow for legitimate comparison. However, there are selected works that have used similar unconstrained experimental procedures and these are presented in Table 3.2. In Table 3.2 the 'Train' column defines the number of training sample used to train the classifier, 'String' is the type of string used and its length (here the '+' indicates that the string may be longer since the number quoted is the minimum number used), 'Pop' is the population size for the experiment, and the other columns are clearly labeled.

The proposed method has similar performance compared to other methods in Table 3.2. Although the proposed method has higher error rates, it uses a population that is approximately 3-5 times larger than other methods in Table 3.2. This is a critical point since increases in population size tend to increase error rates, as can be seen from the last two entries. Furthermore, in [52], the population is too small to support the reported error rates and number of training samples is very high (7140 sample strings per user); this condition

Source	Method	Feature	Train	String	Pop	Error
Obiadat and Sadoun [52]	Neural Networks	DDKL, KD	7140	User ID (7avg.)	15	FAR=0% FRR=0%
Hocquet et al. [32]	Distance, Rhythm	UDKL	10	Fixed Phrase (25)	15	FAR=1.75% FRR=1.69%
Bleha and Slivinsky[6]	Bayesian Classifier	DDKL	10	User's Name (11+)	10	FAR=3.1% FRR=0.5%
Hosseinzadeh et al. [33]	GMM + LOOM	DDKL, KD	10	User's Name (10+)	8	FAR=3.25% FRR=3.0%
This Work	GMM + LOOM	UUKL, KD, DDKL	30	User's Name (10+)	41	FAR=5.1% FRR=6.5%

Table 3.2: Performance comparison with previous works.

would greatly benefit every technique and is commercially unpractical. The method of [32] uses a 25 character string; longer strings are known to significantly improve classification performance. Therefore, this is very much an unfair advantage since all other methods in Table 3.2 use similar string lengths. Lastly, the method of [6] can be expected to have higher error rate with a similar population size as that used in this work. Therefore, all though the works in Table 3.2 cannot be fairly compared to the proposed method for the reasons mentioned above, this selection of previous works illustrates a good sampling of previous efforts.

e) Error Reporting

All of the results for the combined feature sets meet the requirements imposed by the "Rule of 30" and the "Rule of 3". This implies that the number of tests performed were large enough to justify the reported results. Also, since the data was collected over a long period of time in unsupervised conditions, these results are likely to represent the true performance of the system if it were implemented today in a real setting. Lastly, a point is made about the DET vs. ROC curves which are shown in Figure 3.3 and Figure 3.7, respectively. It is clear that in this case the DET curve provides much more visual information than the ROC curve due to its logarithmic axis which spreads the information. As pointed out earlier,

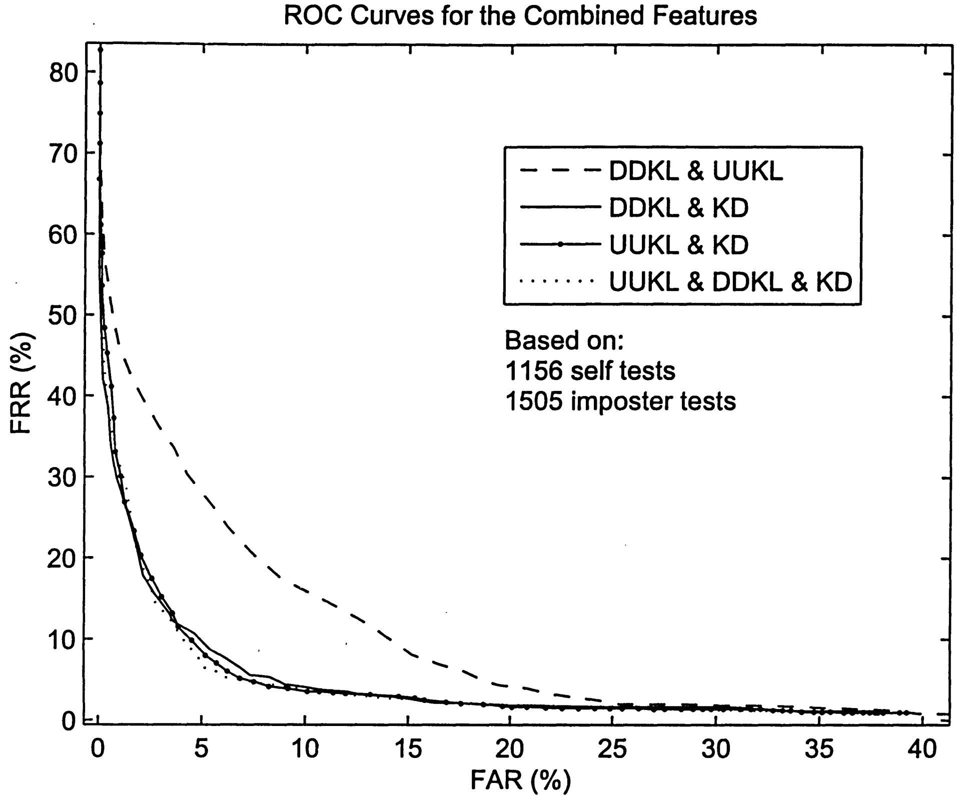


Figure 3.7: Plots of the ROC curves for the combined feature sets.

DET curve are preferable and should be used in most cases to better illustrate a system's performance, especially when comparing multiple systems.

Chapter 4

Speaker Recognition

SPEAKER recognition has many potential applications as a biometric tool since there are many tasks that can be performed remotely using speech. Especially, for telephone based applications such as banking or customer service, there are many costly crimes such as identity theft or fraud that can be prevented by enhanced security protocols. In these applications, the identity of users cannot be verified because there is no direct contact between the user and the service provider. Hence, speaker recognition is the only viable and practical next step for enhanced security. This chapter proposes a cost effective and text-independent GMM based speaker identification scheme that is designed to be used with existing telecommunications infrastructure. This hardware independent method can greatly improve security for remote telephone based applications without significant costs for service providers. The block diagram of the proposed method is shown in Figure 4.1.

GMMs are the most popular statistical tool for speaker recognition because of their ability to accurately capture speech phenomena [64][62][23]. In fact, as already mentioned in Section 2.2.3, some clusters within GMMs can be highly correlated with specific phonemes. And the overall GMM can capture a broad range of phonetic events or acoustic classes within a speaker's utterances. These are very useful characteristics that can lead to very good speaker models if a comprehensive training set is used. A good training set would include multiple instances of a wide range of phoneme combinations.

Since GMMs characterize acoustic classes of speech and not specific words or phrases,

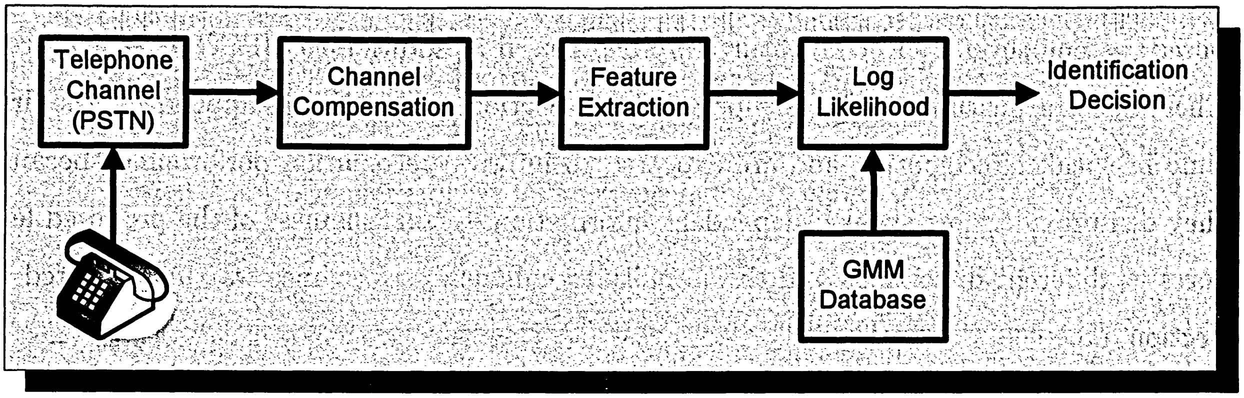


Figure 4.1: Block diagram of the proposed text-independent speaker identification system.

they can be effectively used for text-independent identification. Text-independent systems are much more secure than text-dependent systems because text-independent systems can prompt the user to say any phrase during identification. Conversely, a major drawback of text-dependent speaker recognition systems is that they use predetermined phrases for authentication; so it is possible to use a recorded utterance of a valid user to "fool" the system. This issue is particularly important for telephone based applications since there is no physical contact with the person requesting access and therefore, text-independent systems are required. Additionally, in a text-independent speaker recognition system, speech recognition techniques may be used to verify the contents of the utterance before identifying the speaker; for enhanced security.

Another advantage of GMM based systems is their ability to be resilient to different types of noise. By using noisy training data, GMMs can become robust to different (or expected) environmental conditions [60]. For example, by mixing babble noise with the training data the model can become more robust to background speaker interference or white noise can be used in the training data for better performance in all environments. In fact, by adding distortions to the training set, the model will become robust to other types of distortions not found in the training set [60].

Since MFCC based features are the most popular and successful feature set for speaker recognition [62][23][9], they were used here as well. To enhance the performance of the proposed system, several spectral features are being used in addition to MFCC based features. These spectral features, which are presented in Section 4.1, are being applied for the first time in speaker recognition and are expected to improve recognition performance because they can provide more speaker-dependent information. The robustness of the proposed features to different distortions and their ability to enhance performance was be evaluated in Section 4.5.

The proposed system is also robust in the sense that it can function as an identification system or a verification system. Verification systems have an added advantage because they do not require a predefined threshold to detect unknown imposters (see Section 4.4 for details).

4.1 Feature Selection

It is known that the most user-specific quality of speech is embedded in the vocal tract. The vocal tract acts as a time-varying filter that modifies the input from the vocal cords and is highly user-specific. Research in speaker recognition has mainly focused on how to extract this vocal tract information (filter characteristics) from the output speech signal. Given the speech system model in Figure 2.4, deconvolution is one way of separating this information. Linear prediction coefficients (LPC), linear prediction based cepstral coefficients (LPCC), and auto regressive (AR) modeling are other methods that have been used for extracting information about the vocal tract configuration from speech signals [9][75][2]. However, the most successful features for speaker recognition have been the cepstral based features [79][9].

4.1.1 Cepstral Features

Cepstral based features are widely used for the task of speaker recognition because they accurately characterize speaker-dependent features from speech signals. Cepstral features are obtained using a homomorphic deconvolution operator and therefore, in the cepstral domain the excitation component of speech is separated from the filter component. Mathematically,

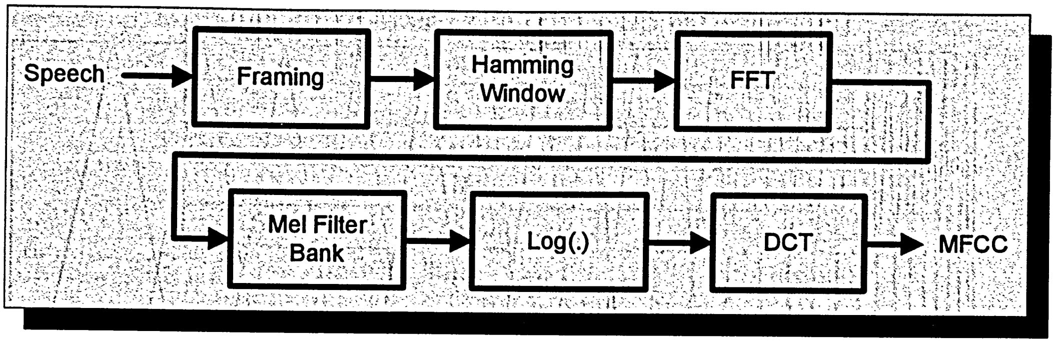


Figure 4.2: Block diagram for generating MFCCs.

this separation is expressed in Equation 2.4.

It has been shown that the first several cepstral coefficients from the cepstral domain represent the anatomical structure of a speaker's vocal tract and as such, these coefficients are highly speaker-dependent [53]. Since the vocal tract can be considered stationary over short periods of time (of approximately 10-30 ms) [28][9][79], cepstral coefficients should be calculated frequently and with different sounds so that the speaker (or more accurately, the speaker vocal tract configuration) can be accurately characterized. In this work, all processing is performed on 20 ms frames with 10 ms of overlap between adjacent frames.

MFCCs are a special form of cepstral coefficients that have been shown to be more effective for speaker recognition; this feature is calculated using the procedure shown in Figure 4.2 [79]. This modified cepstral feature is based on the Mel frequency scale which approximates the non-linear way that humans perceive sounds by emphasizing the lower frequencies more than the higher frequencies [18]. This perceptual masking is achieved by using the Mel filter bank shown in Figure 4.3. Here, the triangular filters are linearly spaced and have a smaller bandwidths at lower frequencies, while at higher frequencies the filters are logarithmically spaced and have larger bandwidths. Typically, the first 12 to 14 MFCCs have been used in previous speaker recognition works [28][29][64]. In this work, the first 14 MFCCs were used in order to obtain a good estimate of the speaker's vocal tract configuration.

The first derivative of the MFCCs (ΔMFCCs), can also capture speaker-dependent

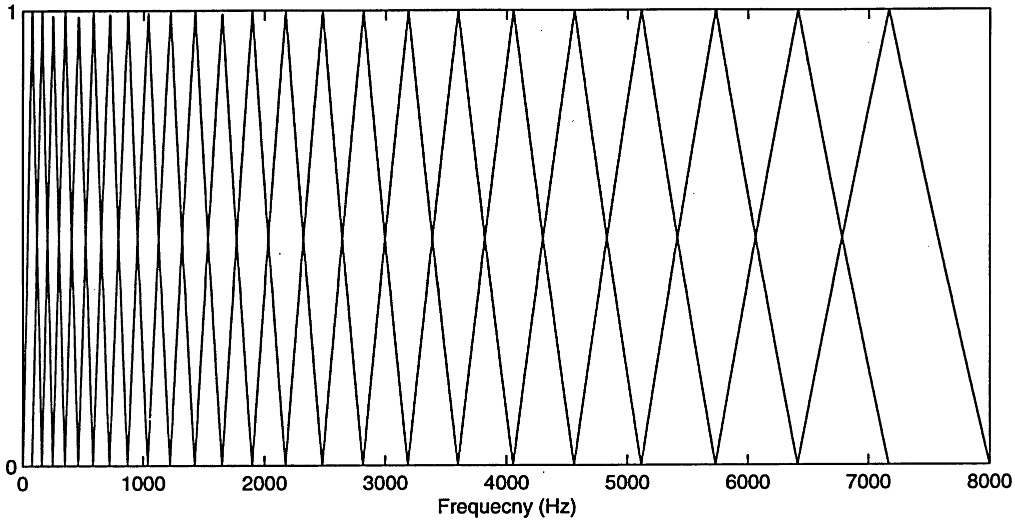


Figure 4.3: The perceptually motivated Mel filter bank before normalization [18].

characteristics. This feature is largely uncorrelated with the MFCC feature and therefore, it can enhance recognition performance. Since these two features have been previously used to obtain good speaker recognition performance, they were also used in this work. In addition to these features, several novel spectral features are also introduced for speaker recognition. These new features were examined and compared with the traditional MFCC based features in an attempt to improve the performance of the recognition system by gathering more speaker-dependent information.

4.1.2 Spectral Features

Using spectral information is a logical area to explore since they describe the frequency content of speech. As a result, spectral features can be expected to improve the performance of MFCC based systems since they can provide some information about the excitation component of the speech signal [23][49]. For example, pitch information, energy distribution, or bandwidth of the speech spectrum contains some speaker-dependent information that is not captured by MFCCs; since MFCCs discard all information about the excitation component

Subband	Lower Edge (Hz)	Upper Edge (Hz)
1	300	550
2	550	750
3	750	1000
4	1000	2000
5	2000	3400

Table 4.1: Subband allocation used to calculated spectral features.

of speech. Therefore, MFCC based features combined with spectral features can describe the complete speech system including the vocal tract configuration and the frequency content of the excitation produced by the vocal cords, respectively.

Similar to MFCCs, spectral information is also obtained from short-time frames of 20 ms in length with 50% of overlap between adjacent frames. By extracting both the spectral features and the MFCC features from the same frames, a better description of the speech signal can be created, as discussed above. This synchronization is important for achieving enhanced performance with spectral features.

In addition, the spectral features were extracted from multiple subbands within the telephone channel's bandwidth, so to minimize errors in a practical implementation. These subbands, which are shown in Table 4.1, will provide better discrimination between different speakers because the trend for a given feature can be captured from the spectrum. This is better than obtaining one global value from the spectrum, which is not likely to show speaker-dependent characteristics. Furthermore, the subband allocation reflects the fact that most of the speech energy is located in the lower subbands, by using narrowly defined subbands in the lower frequencies. This is also consistent with the non-linearities of human auditory perception, which shows more sensitivity to lower frequencies than higher frequencies. This non-linearity has been shown to be important for cepstral based features such as the MFCC feature [18].

Spectral features are extracted from framed speech segments as follows. Let $s_i[n]$ for $n \in [0, N]$, represents the i^{th} speech frame and $S_i[f]$ represents the spectrum of this frame.

Then, $S_i[f]$ can be divided into M non-overlapping subbands where, each subband (b) is defined by a lower frequency edge (l_b) and a upper frequency edge (u_b). Now, each of the seven spectral features can be calculated from $S_i[f]$ as shown below.

- **Spectral Centroid (SC)** - SC as given below is the weighted average frequency for a given subband, where the weights are the normalized energy of each frequency component in that subband. Since this measure captures the center of gravity of each subband, it describes the approximate location of the large peak in each subband. These peaks correspond to the approximate location of formants [54] or pitch frequencies.

$$SC_{i,b} = \frac{\sum_{f=l_b}^{u_b} f|S_i[f]|^2}{\sum_{f=l_b}^{u_b} |S_i[f]|^2} \tag{4.1}$$

- **Spectral Bandwidth (SBW)** - SBW as given below is the weighted average distance from each frequency component in a subband to the spectral centroid of that subband. Here, the weights are the normalized energy of each frequency component in that subband. This measure quantifies the relative spread of each subband for a given sound and therefore, it might characterize some speaker-dependent information.

$$SBW_{i,b} = \frac{\sum_{f=l_b}^{u_b} (f - SC_{i,b})^2 \, |S_i[f]|^2}{\sum_{f=l_b}^{u_b} |S_i[f]|^2} \tag{4.2}$$

- **Spectral Band Energy (SBE)** - SBE as given below is the energy of each subband normalized with the combined energy of the spectrum. The SBE gives the trend of energy distribution for a given sound and therefore, it contains some speaker-dependent information.

$$SBE_{i,b} = \frac{\sum_{f=l_b}^{u_b} |S_i[f]|^2}{\sum_{f,b} |S(f)|^2} \tag{4.3}$$

- **Spectral Flatness Measure (SFM)** - SFM as given below is a measure of the flatness of the spectrum, where white noise has a perfectly flat spectrum. This measure can be useful for discriminating between voiced and un-voiced components of speech [81].

$$SFM_{i,b} = \frac{\left[\prod_{f=l_b}^{u_b} |S_i[f]|^2\right]^{\frac{1}{u_b-l_b+1}}}{\frac{1}{u_b-l_b+1}\sum_{f=l_b}^{u_b} |S_i[f]|^2} \tag{4.4}$$

- **Spectral Crest Factor (SCF)** - SCF as given below provides a measure for quantifying the tonality of the signal. This measure is useful for discriminating between wideband and narrowband signals since it describes the relative strength of the largest peak in a subband. These peaks correspond to the most dominant pitch frequency in each subband.

$$SCF_{i,b} = \frac{max(|S_i[f]|^2)}{\frac{1}{u_b-l_b+1}\sum_{f=l_b}^{u_b} |S_i[f]|^2} \tag{4.5}$$

- **Renyi Entropy (RE)** - RE as given below is an information theoretic measure that quantifies the randomness of the subband. Here, the normalized energy of the subband can be treated as a probability distribution for calculating entropy and α is set to 3, as commonly found in literature [24][5]. This RE trend is useful for detecting the voiced and unvoiced components of speech.

$$RE_{i,b} = \frac{1}{1-\alpha}\log_2\left(\sum_{f=l_b}^{u_b}\left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right|^{\alpha}\right) \tag{4.6}$$

- **Shannon Entropy (SE)** - SE as given below is also an information theoretic measure that quantifies the randomness of the subband. Here, the normalized energy of the subband can be treated as a probability distribution for calculating entropy. Similar to the RE trend, the SE trend is also useful for detecting the voiced and unvoiced components of speech.

$$SE_{i,b} = -\sum_{f=l_b}^{u_b}\left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right| \cdot \log_2\left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right| \tag{4.7}$$

All of these features are being used for the first time in speaker recognition. These spectral features along with the MFCC and ΔMFCC features were extracted from each speech frame and appended together to form a combined feature matrix for the speech signal. These vectors can then be modeled and used for speaker identification. Equation 4.8 shows the feature matrix that can be extracted based on only one spectral feature, say the SC feature, from i frames; where the bracketed number is the length of the feature. It should be noted that any other spectral feature can be substituted in for the SC feature in the feature matrix, which is used to train the GMM for each speaker.

$$\vec{\mathcal{F}} = \begin{bmatrix} MFCC_1(14) & \Delta MFCC_1(14) & SC_1(5) \\ \vdots & \vdots & \vdots \\ MFCC_i(14) & \Delta MFCC_i(14) & SC_i(5) \end{bmatrix} \tag{4.8}$$

The spectral features are expected to be uncorrelated with the MFCC based features because the spectral features capture information about the frequency content of speech, whereas the MFCC features capture information about the vocal tract configuration. Among the spectral features, there is some correlation between the SC and the SCF features because they both characterize the location of peaks (or locations of energy concentration) in each subband. The difference is that the SCF feature describes the largest peak in each subband while the SC feature describes the center of gravity of each subband. Therefore, these features will produce similar results if the largest peak in a given subband is much larger than all other peaks in that subband.

The RE and SE features are also correlated since they are both entropy measures. However, the RE feature is more sensitive to small changes in the spectrum because of the exponent term α. Therefore, although these features quantify the same type of information, their performance may be different for speech signals.

4.2 Channel Compensation Techniques

The proposed method is intended to work over existing communication channels. Since these channels are often bandpass in nature and are subject to additive distortions such as additive white Gaussian noise (AWGN), some compensation techniques are required to ensure that the proposed method remains reliable in practice.

Telephone channels are often approximated to have a bandpass range of 300-3400 Hz. Therefore, when extracting features from telephone speech two types of compensation techniques can be used [64]: either features should be extracted from the bandpass range of the channel or frequency warping methods should be used to map the linear frequency axis (f) to modified axis (f'). The simple and commonly used warping function given below stretches (or spreads) f so that it encompasses the desired frequency range; where the desired frequency range is often half of the sampling frequency (f_s) and f_{min} and f_{max} describe the bandpass range of the channel.

$$f' = \frac{f - f_{min}}{f_{max} - f_{min}} \cdot \frac{f_s}{2} \tag{4.9}$$

This warping function allows normal processing to be performed on telephone speech by stretching the distorted spectrum to fill empty gaps. However, this technique also introduces some distortions and will lead to degraded performance. Hence, in this work, warping was not used and all features were extracted from the available bandwidth.

The shape of the magnitude spectrum of the telephone channel ($h[n]$) also affects the extracted features. Especially for the MFCC feature, which has the best recognition performance among all speech based features, linear convolution results in additive distortion. It has been shown that if $h[n]$ is relatively smooth, then the channel will have an additive effect on the output cepstral coefficients [63], given by:

$$\vec{Y} = \vec{S} + \vec{H} \tag{4.10}$$

where, $\vec{Y}$ is the output cepstral coefficient vector, $\vec{S}$ is the input cepstral coefficient vector, and $\vec{H}$ is the cepstral coefficients of the channel derived from $h[n]$. There are two commonly used methods to overcome this additive effect: cepstral mean normalization (CMN) and RASTA processing [64][30].

The CMN method is easily implemented by estimating the mean of the cepstral vector and removing it from the output cepstral coefficients, as shown below.

$$\vec{Y}_{CMN} = \vec{Y} - \frac{1}{N}\sum_{i=1}^{N}\vec{Y_i} \tag{4.11}$$

where, $\vec{Y}_{CMN}$ is the normalized cepstral vector and $\vec{Y}_i$ is the i^{th} element of $\vec{Y}$. This method has the advantage that it removes any global side effects imposed by different channels but also, it removes any intra-speaker biases introduced over different sessions from the intensity of the spoken speech (i.e. how loud it is) [64]. However, the assumption of the CMN technique is that the channel is time-invariant [9] and therefore, it cannot be used with all channels. RASTA processing is another very common technique that can be used with time-varying channels and it too, is easily implemented using the infinite impulse response (IIR) filter given by:

$$\vec{Y}_{RASTA}[n] = \vec{Y}[n] - \vec{Y}[n-1] + 0.97\vec{Y}_{RASTA}[n-1] \tag{4.12}$$

Research has shown that RASTA processing and CMN have comparable performance for single speaker identification with telephone quality speech [62][29]. The only advantage of RASTA processing is that it is able to better overcome the effects of mismatched microphones (from using different telephones), which are a part of the overall channel distortion [62]. Nonetheless, CMN was used in this work because it can better cope with intra-speaker variabilities and it provides similar performance as RASTA processing. Removing intra-speaker variabilities improves speaker recognition performance because it minimizes the variability introduced from different sessions due to the intensity of speech and also, it minimizes spectral shaping from different telephone channels [64].

RASTA processing and CMN are post processing techniques that attempt to compensate for channel distortions. Another approach for minimizing channel effects is to design robust channel invariant features. For example, cepstral difference coefficients such as the ΔMFCC are less affected by channel distortions because they rely on the difference between samples and not on the absolute value of the samples. Therefore, the difference coefficients are not significantly affected by time-invariant channel distortions [64]. Despite this advantage, the difference cepstral features do not perform as well as the regular cepstral features for speaker recognition, but the difference cepstral features can enhance the recognition performance of regular cepstral features since the two types of features are uncorrelated [64][9]. Therefore, the ΔMFCC features are also used to improve recognition performance in this work.

4.3 Training and GMM Estimation

The performance of GMMs are affected by several factors including the quality of training data, amount of training data, model order etc. These factors will be discussed here.

The expectation maximization (EM) algorithm was used to estimate the parameters of the GMM model as described in Section 3.2.2 and Appendix A. Although the EM algorithm is an unsupervised clustering algorithm, it cannot estimate the model order and it requires an initial grouping of the data. In previous speaker recognition works the question for model order has been decisively answered. Although model orders of 8 to 32 are common, most have concluded that a model order of 16 (or a 16 component GMM) is sufficient for speaker recognition [64][29]. It has also been shown that the initial grouping of data does not significantly affect the performance of GMM based recognition systems [64]. Hence, in this work, model order of 16 was used with the k-means algorithm which provided the initial estimate for each of the 16 clusters.

Diagonal covariance matrices were used to estimate the variances of each cluster in the models since it is well known that diagonal covariance matrices are much more computationally efficient than full covariance matrices. Furthermore, diagonal covariance matrices can provide the same level of performance as full covariance matrices because they can capture the correlation among the components of the feature vector if a larger model order is used. For these reasons, diagonal covariance matrices have almost been exclusively used in previous speaker recognition works. Each element of these matrices is limited to a minimum value of 0.01 during the EM estimation process to prevent singularities in the matrix, as recommended by [64].

4.4 Authentication Scheme

The proposed method is an identification system by design since the log-likelihood function is used to find the model (Λ) that best matches a given utterance ($\vec{x}$), as shown below.

$$\mathcal{L}(\vec{x}) = \max_{i}\{\log p(\vec{x}|\Lambda_i)\} \tag{4.13}$$

where, $p(.)$ denotes the probability function. In such an identification system, the user model that produces the best likelihood result for a given utterance will be selected as the correct user. This means, that no matter who provides a sample, the system will select the best match for that sample and declare the corresponding user as identified. This is problematic because if an unknown user provides the test sample, than the system will still pick the model that provides the best likelihood value; even though this user is not a valid user because he or she is not known to the system. To prevent this scenario, most identification systems compare the likelihood value of identified sample with a threshold before accepting that user. Therefore, a user specific threshold is required so that the identification system can reject unknown users from being incorrectly identified as a valid user.

Since the proposed system is designed for telephone based applications there is always an implicit need for an identity claim. Users should always identify themselves to the service provider so that the appropriate account can be serviced over the telephone. Therefore, the identification system should always return the same identity as the claimed identity, thus the proposed system does not require a threshold for this application.

4.5 Experimental Results

This section presents the experimental conditions as well as the results. Section 4.5.1 explains the details of the experimental procedures and the data collection procedures, while Section 4.5.2 provides a detailed discussion about the results.

4.5.1 Experimental Conditions

All speech samples used in these experiments were obtained from the well known TIMIT speech corpus [50]. 100 users were randomly chosen from this database which has speakers from 8 different dialect regions in the United States. Each user provided 10 recordings which have a wide range of phonetic sounds suitable for training the classifier. However, the recordings are made in an acoustically quiet environment using a high quality microphone and therefore, distortions were added to simulate a practical telephone channel.

The distortions used include bandpass filtering (from 300-3400 Hz) to simulate the characteristics of a telephone channel, babble noise to simulate background speakers that might be found in some environments, and AWGN to simulate normal background noise found in many environments. Each GMM was trained with 20 seconds of silence removed clean speech and separately, with 20 seconds of noisy speech. The noisy speech was obtained by first adding AWGN to the clean speech signal to obtain an SNR of 25 decibels (dB) and then to this signal, babble noise was added to obtain an SNR of 15 dB. This results in a total SNR of nearly 15 dB for the noisy training data. Bandpass filtering was also applied to the noisy training data to simulate the bandpass range of the telephone channel. The remaining speech was segmented into 5 second slices and used to test the two different models under noisy and noise free conditions.

Since the TIMIT database has a sampling frequency of 16kHz, the signals were down sampled to 8kHz. This not only suits telephone applications better but also does not degrade the quality of speech significantly. Features were extracted from 20 ms long frames with 10 ms of overlap with the previous frames and a Hamming window was applied to each frame to ensure a smooth frequency transition between frames. From each frame, the feature matrix ($\vec{\mathcal{F}}$) extracted was a concatenation of a 14 dimensional MFCC vector, a 14 dimensional ΔMFCC, and a 5 dimensional spectral feature vector as shown in Equation 4.8. In cases where multiple spectral features are used, all features are appended together to form the feature matrix as shown in the example below.

$$\vec{\mathcal{F}} = \begin{bmatrix} MFCC_1(14) & \Delta MFCC_1(14) & SC_i(5) & SCF_i(5) & SBE_i(5) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ MFCC_i(14) & \Delta MFCC_i(14) & SC_i(5) & SCF_i(5) & SBE_i(5) \end{bmatrix} \tag{4.14}$$

where, i represents the frame number and the bracketed number represents the length of the feature. The feature matrix was used to train a 16-component GMM for each speaker.

4.5.2 Results and Discussions

These experiments aim to demonstrate the effectiveness of the proposed speaker recognition system under practical circumstances. However, before presenting these results, an analysis is performed on the proposed novel spectral features.

Spectral features are expected to increase identification accuracy of MFCC based systems because they provide some information about the source of speech signals. MFCC based features discard all information about the excitation component and only characterize the anatomical structure of the vocal tract. Although this is the main reason for the success of the MFCC based features, it is not a complete description of the speaker's speech system. Since, there is some speaker-dependent information embedded in the excitation component of speech, seven spectral features were introduced to capture this information.

In order to demonstrate the effectiveness of the spectral features, each of the spectral features was combined with the MFCC based features to create an enhanced system. Then the performance of the enhanced system was compared to the baseline system, which is a GMM classifier, as discussed in Section 4.3, trained with the MFCC and ΔMFCC features.

a) Robustness to Undistorted Speech

Table 4.2 demonstrates the identification error of the system when using spectral features in addition to MFCC based features with undistorted speech sampled at 8kHz. This error rate represents the percentage of tests that were incorrectly identified by the system, as shown below.

$$\text{Error} = \frac{\text{Samples Incorrectly Identified}}{\text{Total Number of Samples}} \tag{4.15}$$

Feature	Error (%)
MFCC & ΔMFCC (Baseline System)	7.31
MFCC & ΔMFCC & SC	3.66
MFCC & ΔMFCC & SCF	3.66
MFCC & ΔMFCC & SBE	4.87
MFCC & ΔMFCC & SBW	2.43
MFCC & ΔMFCC & SFM	29.27
MFCC & ΔMFCC & SE	7.32
MFCC & ΔMFCC & RE	6.10
MFCC & ΔMFCC & SC & SCF & SBE	**1.22**

Table 4.2: Experimental results: based on 83 tests with 5-sec of undistorted speech.

It is evident from these results that there is some speaker-dependent information captured by the SC, SCF, SBW, SBE, and RE features as they improved error rates when combined with the standard MFCC based features. In fact, when three of the best performing spectral features (SC, SCF, and SBE) were simultaneously combined with the MFCC based features, an identification error of 1.22% was achieved, which represents a 6% improvement when compared with using MFCC based features alone. These results suggest that the spectral features provide enough speaker-dependent information about the speaker's vocal cord activity to enhance the performance of MFCC based features.

As noted in Section 2.2.2, the most dominant structures in the speech spectrum are pitch, energy distribution, and formant locations. Formants are large humps in the envelope of the speech spectrum that represent the resonant frequencies of the vocal tract. And pitch information represents the frequency of the vocal cords and is often seen in the speech spectrum through periodic spikes. As can be seen from Figure 4.4(a) and Figure 4.4(b), pitch information can be easily detected by calculating the SC and SCF features.

From Figure 4.4(a) it can be seen that the SC feature (given in Equation 4.1) can approximately detect the location of the largest spike in each subband. This location corresponds to a particular pitch frequency or the approximate location of a formant (in cases where several peaks fall in the same subband). However, the SCF feature (given in Equation 4.5)

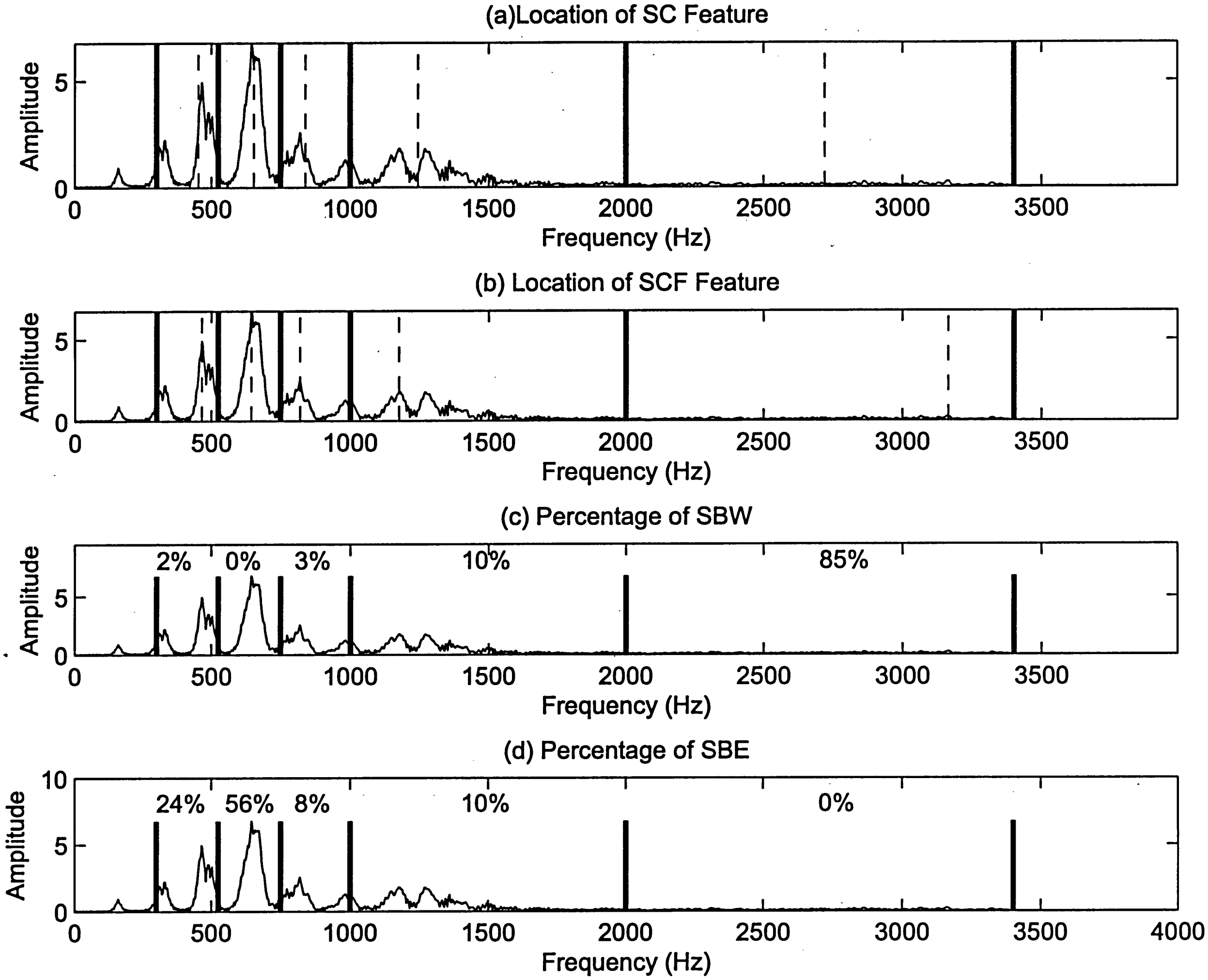


Figure 4.4: Plot illustrates four of the spectral features. Subband boundaries are indicated with dark solid lines. (a) Shows the location of the SC in each subband with dashed lines. (b) Shows the location of the SCF in each subband with dashed lines. (c) Shows the SBW as a percentage of the five subbands. (d) Shows the SBE as a percentage of the of the whole spectrum.

shown in Figure 4.4(b), describes the exact location of the dominant spike in each subband. As a result, the SC feature tends to capture 'formant like' information and pitch information, while the SCF feature captures the location of the dominant pitch frequency in each subband. Since this information has speaker-dependent characteristics, as evident from the identification results of Table 4.2, the performance of the system was improved by more than

50% when either of these spectral features were used with the MFCC based features. The use of narrowly defined subbands in the lower frequency range also helped this performance because most of the energy of the speech spectrum is contained in the lower frequencies. Therefore, high frequency subbands cannot be relied upon for much information; although Figure 4.4 is only one example, it can illustrate this point.

The SBE feature (given in Equation 4.3) also performed well because it captures the normalized energy (relative strength) of each subband. This feature provides the trend for the amount of energy in each subband as a percentage of the energy in the entire spectrum, as shown in Figure 4.4(d). Additionally, since the SBE feature is a normalized energy measure, it is not affected by the intensity (or relative loudness) of speech from different sessions. Therefore, the SBE trend for a given sound seems to be a unique speaker-dependent feature that can improve the performance of the MFCC based features. This improvement, shown in Table 4.2, also suggests that for a given vocal tract configuration (given by the MFCC features) the SBE trend is predictable.

The SBW feature (given in Equation 4.2) is largely dependent on the SC feature and the energy distribution of each subband therefore, it has also performed well for the reasons mentioned above. This feature showed the best performance among all of the proposed features, an improvement of 70% over the MFCC based features. This is because the SBW feature is effectively based on two of the best performing spectral features (the SBE and SC features). Figure 4.4(c) shows the SBW for of each subband as a percentage of the 5 subbands.

The features which did not perform well all have similar weaknesses because they quantify characteristics that are not well defined in speech signals. For example, the SFM feature (given in Equation 4.4) measures the tonality of the subband, a characteristic that is difficult to define in the speech spectrum since its energy is distributed across many frequencies. And although the entropy features provided some additional information, they did not perform much better than the MFCC features. This may be because the speech spectrum has a lot of sample-wise variations. Especially in the lower subbands where most of the energy is

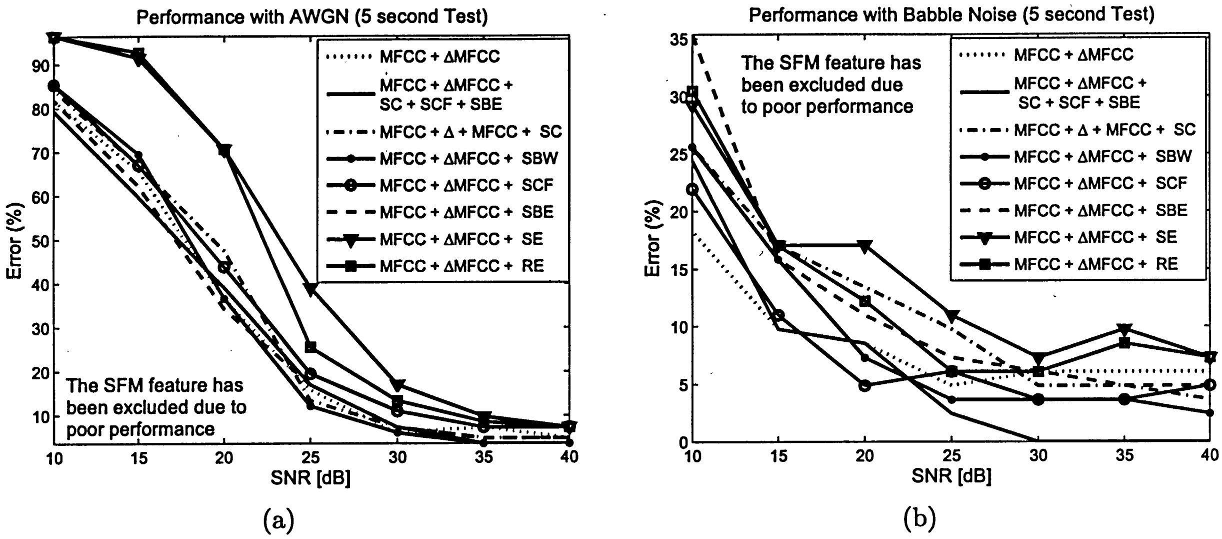


Figure 4.5: Performance of spectral features with noise with 5 second test utterance. (a) with AWGN, (b) with babble noise.

contained, these variations are quite large and without any particular pattern. Therefore, no distinct pattern was captured by the entropy features.

b) Robustness to Distortions

Figure 4.5 shows the performance of the spectral features with AWGN and babble noise. It can be seen that the proposed features are robust to these types of noise since their performance does not degrade significantly when compared with the MFCC features. For AWGN, the SC and SBW combined with the MFCC features provides a slightly better performance than the MFCC features for mid to high signal to noise ratio (SNR) values.

The main reason for the better performance of the SC feature is that it captures the center of gravity of each subband, which is near the peak of the subband. Therefore, as long as this peak is larger than the noise, which is to be expected, the SC feature is not significantly affected by AWGN. The SBW feature is also based on the SC feature so an accurate SC will provide an accurate SBW as well. Furthermore, AWGN will simply raise the energy of each frequency component equally therefore, it does not significantly effect the SBW calculation.

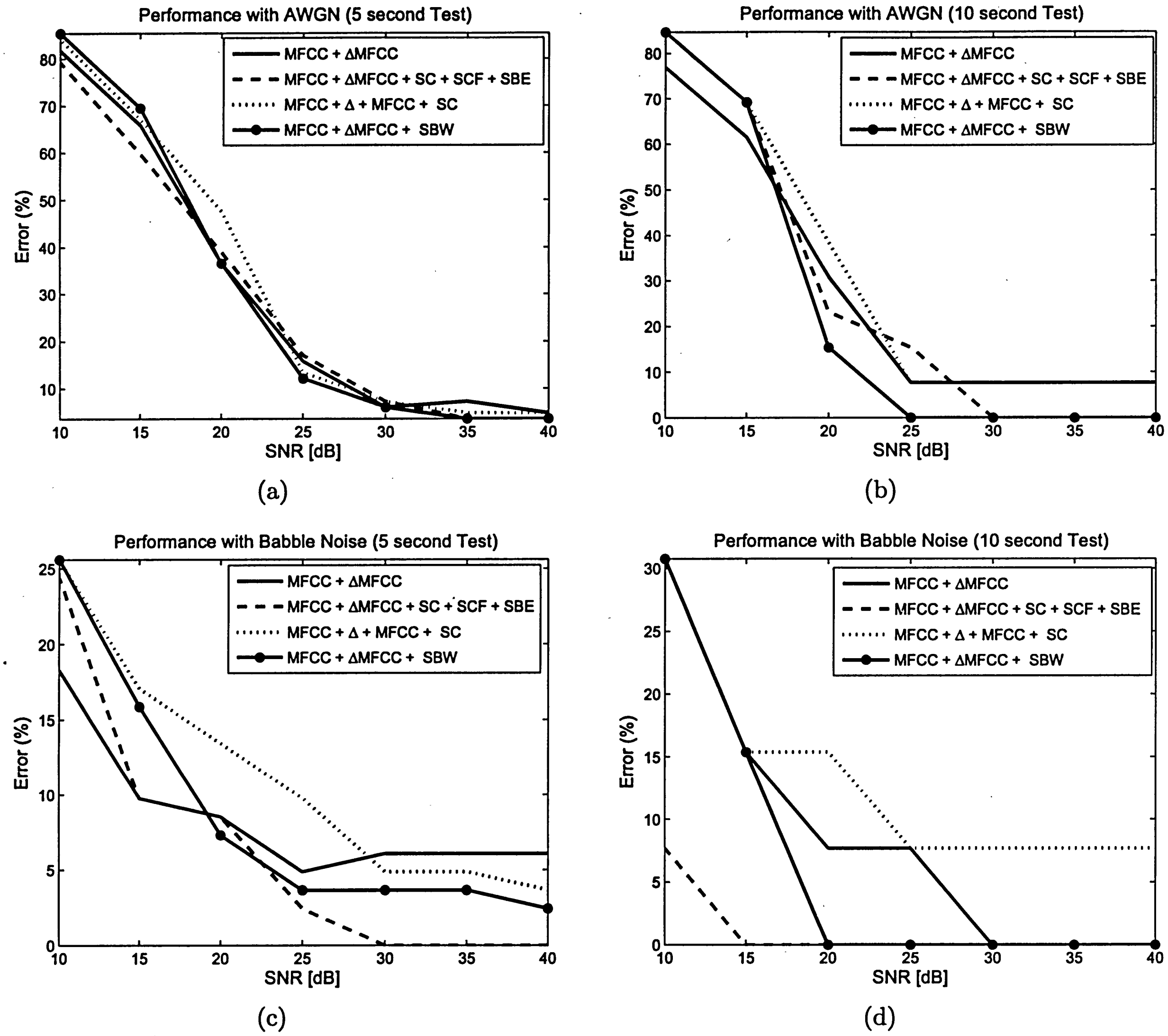
The SBW feature also performed well in the case of babble noise. It improved performance for mid to high SNR values compared with the MFCC based features. However, the combined features of SC, SBE, SCF, and the MFCC based features performed even better than the SBW feature combination under babble noise. This feature combination results in a 0% identification error for SNR values greater than 29 dB. The SCF feature combination also produced better results than the MFCC based features under babble noise.

c) Effects of Test Length

As evident from Figure 4.6, the spectral features greatly enhance identification performance as the length of the test utterance increases. With a 10 second long test, the system's performance is much better with spectral features than with MFCC features. While the trends are roughly the same as the tests above, some of the specific outcomes are highlighted below.

In the case of babble noise, a 10 second test utterance produced a 0% error rate for the SC, SBE, SCF, and MFCC feature combination at SNR values greater than 15 dB. The SBW and MFCC feature combination achieved a 0% error rate at SNR values greater than 20 dB. Although these feature combinations consistently perform better than MFCC based features, the results show that the performance gain is very high with babble noise and longer test utterances. These two feature combinations also obtained the best results under AWGN, especially with the 10 second test utterance, as can be seen from Figure 4.6.

It should be noted that "toll quality" speech (or acceptable quality of speech on the telephone networks) is generally perceptually defined. This means that many users are asked to listen and judge the quality of telephone speech based on clarity. In general, the speech quality used by many telephone companies is very good; i.e. no significant distortion is audible in the speech signal. Therefore, the results presented here, which show a 0% error rate for SNR values as low as 15 dB is a significant achievement. Low SNR conditions are not common because they would result in low quality speech. Therefore, these conditions will only be encountered in noisy environments and not from the telephone channel.



Performance with AWGN (5 second Test)

Performance with AWGN (10 second Test)

Performance with Babble Noise (5 second Test)

Performance with Babble Noise (10 second Test)

(a)

(b)

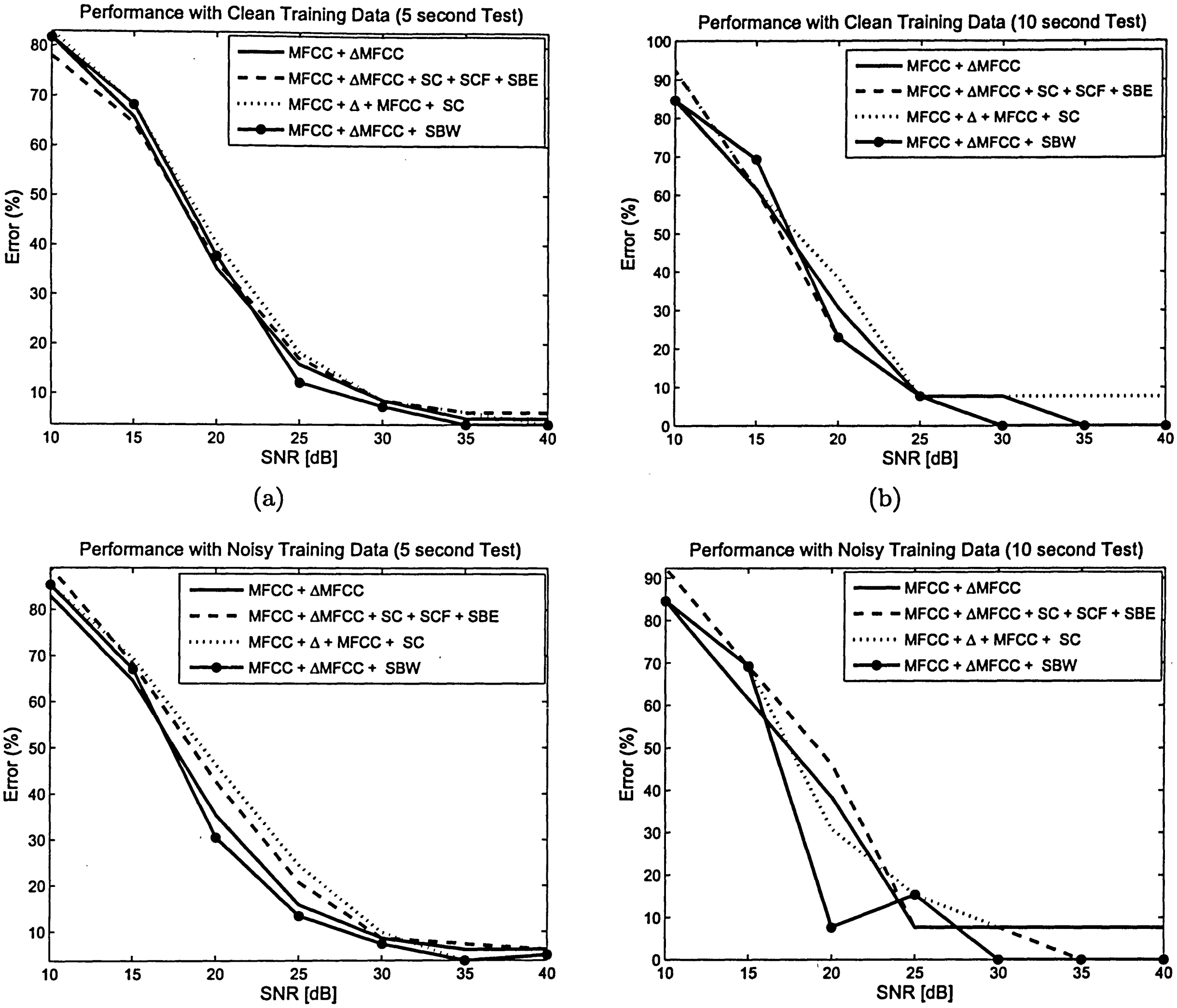
(c)

(d)

Figure 4.6: Performance of spectral features with noise. (a) AWGN with 5 second test utterance, (b) AWGN with 10 second test utterance, (c) Babble noise with 5 second test utterance, (d) Babble noise with 10 second test utterance.

d) Robustness to Channel Effects

Lastly, Figure 4.7 shows the effect of using distorted training data on identification performance. Here, the distortions used in the training data were described in Section 4.5.1 and all of these distortions (AWGN, babble noise, and bandpass filtering) were applied in the testing sessions as well to observe the benefits of using distorted data in the training set. It can be seen that using distorted data in the training set can improve performance especially if



(a)

(b)

(c)

(d)

Figure 4.7: Performance of spectral features for distorted training data and clean training data. Both models were tested with distorted test speech. The distortions used were AWGN, babble noise and bandpass filtering. (a) Clean training data with 5 second tests data (b) Clean training data with 10 second test data (c) Noisy training data with 5 second test data (d) Noisy training data with 10 second test data

longer test sequences are used. For example, in the case where 10 second test utterances was used, the SBW feature and the SC & SCF & SBE feature combination provided significant improvements when the training was performed with distorted data. However, in both cases (noisy and clean training) the SBW feature provided better performance when compared to the baseline system.

The difference between these results and the results of Figure 4.6 is that here, bandpass distortion has been added to simulate the telephone channel and babble noise and AWGN have also been added in equal amounts to the test utterances. Therefore, the performance results in Figure 4.7 are the most convincing because three of the most common distortions have been simultaneously added to simulate the telephone channel and the speaker's environment. As a result, it can be concluded that the SBW feature is the best spectral feature under practical situations, especially for longer test sequences. For test sequences of 10 second, it provides 0% error rate for SNR values greater than 30 dB. This is much better than MFCC based features, which cannot achieve 0% error rates even with no distortion, as shown in Table 4.2.

Often times, modern telephone networks use speech coders before transmitting speech signals over the communication link. Although this aspect was not simulated in the presented results, it is well known that the MFCC features can withstand these distortions and still obtain good results [64][83]. The spectral features are also expected to withstand distortions from speech coders since these features have performed well in the area of audio watermarking, where attacks such as compression with Motion Pictures Experts Group 1 Layer 3 (MP3), Advanced Audio Coding (AAC), and Windows Media Audio (WMA) as well as other distortions such as amplitude distortion, frequency distortion, change in pitch, resampling, and echo addition are common [60].

Chapter 5

Conclusions

THE lack of biometric based user recognition systems has become a costly and difficult problem in recent years. Losses as a result of crimes such as fraud and identity theft, which are often incurred via remote transactions, have reached staggering amounts. It is estimated that in 2003 the losses of individuals and businesses worldwide due to identity theft and fraud was US $221 billion and growing rapidly. These losses occur because the identity of people cannot be remotely verified on the Internet or over the telephone. And with the increasing popularity of both of these technologies, the problem seems to be getting worse.

Biometric solutions are a good way to counter the costly side effects of doing business over the Internet and the telephone. Biometrics, such as fingerprints, iris patterns, retinal patterns, and hand geometry have reached a mature state and have very good accuracy. However, these technologies require expensive hardware sensors to detect the biometric signature. Therefore, they cannot be used for remote authentication since it is not possible to deploy them on a large scale, even though it is well known that large scale deployment of biometric technologies can eliminate the vast majority of all crimes related to remote transaction fraud.

Although there is no shortage of off-the-shelf biometric technologies, they is certainly a shortage of ubiquitous technologies.

ubiquitous - 'present, appearing, or found everywhere.'

- *Compact Oxford English Dictionary of Current English*

Ubiquitous technologies are those that can function with existing hardware and infrastructure. In other words, users or service providers should not have to invest in any additional hardware other than what they need to establish a remote connection. For example, speech is good example of a ubiquitous biometric technology because telephones and telecommunications infrastructure can be found anywhere in the world and therefore, any transaction performed via the telephone can use speaker recognition technology. The only other truly ubiquitous biometric technology is keystroke recognition. This behavioral biometric examines a computer user's typing pattern, which has been shown to be a unique biometric signature. This biometric has the advantage that it can be used with *any* computer system for user recognition since it is completely hardware independent. This hardware independent characteristic is supported by research which has shown that keystroke patterns are produced in the brain before they are reproduced by the fingers, as described in Section 2.1.1. As a result of this hardware independency, keystroke patterns can significantly enhance security for any computer based authentication application; whether it be for Internet applications, personal computer security, or computer network security.

5.1 Keystroke Verification

The proposed keystroke recognition method was a GMM based text-dependent verification system. The main advantages of text-dependent keystroke recognition is that it can work with a much shorter training session and string length than text-independent systems. Therefore, text-dependent systems are much more practical for use with everyday security protocols such as the username and password model. In fact, the proposed technique can be seamlessly integrated with any computer based application because it simply monitors the users keystroke activity during the normal login process. So, in addition to a secret password, which is the most common security technique today, a hidden layer of biometric security can be added by examining the user's keystroke pattern.

Three different features were used for this verification task. Two of these features, which are KD and DDKL features, have been commonly used in previous works. However, the UUKL feature was used for the first time in this work. In terms of latency features, experimental results indicate that the UUKL feature has a better FAR and FRR performance than the standard DDKL feature. Also, the UUKL feature provided the best FAR among all three features used. Nevertheless, neither of these features performed well individually but by combining these features much better performance was achieved. In general, when using multiple features, the FRR can be significantly improved without a significant increase in the FAR. This is because valid users are more likely to pass these tests than imposters. Using all three features, a FAR of 5.1%, a FRR of 6.5%, and a EER of 5.8% were achieved. Other characteristics and advantages of the proposed system are given below:

- LOOM Thresholding - The LOOM provides a least biased, adaptive, and user-dependent threshold. This is advantageous because the threshold value is not database dependent and can change with users over time.

- Two Stage Authentication - By giving each user a second chance at authentication if they fail the first attempt, system performance was significantly improved. Results show that a two stage authentication system decreased the FRR by 14.5% while increasing the FAR by only 2.1%.

- Enrollment - A small number of samples is needed from the user to train the system which is conducive for wide spread use by any computer user. Also, experimental results showed that an increase in the number of training samples has an exponentially decaying effect on the FRR, without a significant effect on the FAR.

Keystroke Protocol

A new keystroke protocol was proposed because there is a great deal of variability in the experimental conditions of previous keystroke recognition works. These conditions often introduce large biases in the experiment which cannot be neglected. For example, collecting the training data and test data in one, two, or three sessions is not likely to provide a true indication of the variabilities in a user's keystroke pattern. These patterns are affected by many physical and psychological factors such as illness, fatigue, mood, etc., and therefore, data should be collected over a long period of time so that the natural variabilities in a user's keystroke pattern can be captured. Otherwise, there can be little confidence in the reported classification results.

The proposed protocol also makes a number of recommendations about how to choose a suitable string for authentication, including the types of strings that should be used and the string's length. For example, familiar strings should be used because they tend to produce stable patterns. Otherwise, the user should practice the chosen string several times before the training session. String length is another important factor. Strings of at least 10 characters should be used so that the classifier can extract enough features, since it is well known that the recognition performance increases with string length.

The proposed keystroke protocol makes a number of other recommendations regarding the minimum number of tests that should be performed before reporting results, based on the "Rule of 30" and the "Rule of 3". Best practices for reporting results such as providing ROC or DET curves with clearly defined EER are also discussed. And lastly, the timing resolution for capturing keystroke timing data should be between 0.1-1 ms.

This protocol was intended to highlight common issues for keystroke recognition systems and provide some guidelines on how to deal with them. However, it also provides some basic knowledge about the characteristics of this biometric, which is useful for creating good experiments and also for comparing different techniques.

5.2 Speaker Identification

Speaker identification can be very useful for telephone-based customer service applications. Here, legitimate account holders and imposters cannot be distinguished since there is no direct contact between the user and the service provider. Speaker recognition can easily resolve this problem with minimal costs to the user or the service provider. Furthermore, this technology is truly ubiquitous since telephones are available practically anywhere in the world.

A GMM based text-independent speaker identification method was used in this work. GMMs are well known to capture acoustic classes from speech signals, which sometimes correspond to phonemes. As a result, GMMs have been commonly used for speaker recognition in recent years. Text-independence is a very important criteria for such systems because it prevents invalid users from using recordings of valid users to "fool" the system. This functionality is especially important for remote authentication applications.

Speaker identification is traditionally performed by extracting MFCC features from speech. These features characterize the anatomical configuration of the vocal tract and therefore, they are highly speaker-dependent. However, these features do not capture any information about the source or frequency content of the speech signal. Since the speech spectrum is known to contain some speaker-dependent information such as pitch and energy distribution, capturing some of this information can improve the performance of MFCC based features. To capture additional speaker-dependent information, several spectral features were proposed which are being used for the first time in speaker recognition. These features include SC, SCF, SBW, SBE, SFM, RE, and SE.

Experimental results show that the spectral features improve the performance of MFCC based features. In particular, the SBW feature combined with the MFCC and the ΔMFCC features consistently outperformed all other feature combinations. Other spectral features such as the SC, SBE, SCF, and RE also improved the performance of MFCC based features but with varying levels of success. Much of the success of the SC and SCF features is because they tend to capture 'formant-like' and pitch information from each subband, while the SBE and RE features tend to capture the trend of energy distribution among subbands. Based on 100 users from the TIMIT database, these features achieved an identification error of 1.22% (for clean speech) by incorporating information about the source of the speech signal. This represents a 6% improvement over the MFCC based features.

The spectral features are also robust to different distortions. AWGN, babble noise, and bandpass filtering (300-3400 Hz) were individually and simultaneously applied to the speech signals to simulate the identification rate of the proposed features for a practical telephone channel. The results indicate that better performance can be achieved in all cases when using spectral features with MFCC based features. Especially, when the length of the test utterance is increased to 10 second, the spectral features perform much better than the MFCC based features. For example, the SBW feature combined with the MFCC based features resulted in a 0% error rate when bandpass filtering, 30 dB of AWGN, and 30 dB of babble noise were simultaneously applied to the speech signal. Furthermore, because all of the spectral features are energy normalized measures, they are robust to intra-speaker biases stemming from the effort or intensity of speech in different sessions (i.e. how loud it is). Overall, the spectral features, and in particular the SBW feature, are worthwhile and should be used with MFCC features for enhanced performance.

Spectral features improved the overall identification performance because they complement the MFCC based features. Since the MFCC features *only* capture information about the anatomical configuration of the vocal tract, all information about the source of the speech signal is lost. Therefore, spectral features can provide additional information about the source of speech signal, which leads to a more accurate description of the speaker's speech system.

5.3 Future Work

Both of the methods presented are biometric technologies that are affected by behavioral characteristics. In particular, keystroke recognition is a behavioral biometric and therefore, it is subject to a lot of variability. To combat this problem, effective user-dependent temporal filters should be designed to remove extreme outliers from keystroke patterns. This will help to create a more defined model with a smaller variance which will produce better FARs. Temporal filtering can even produce better FRRs if the system can intelligently replace or recapture the outlier values so that test samples can become "noise free". Also, the proposed technique should be implemented with a larger number of users to study aspects related to performance and scalability.

The good performance of spectral features for speaker recognition in this simple speaker identification system is very promising. These features should also produce good results if used with more sophisticated speaker recognition techniques such as universal background model (UBM) based approaches. Furthermore, in this work, the spectral features were extracted from several subbands assigned in a similar fashion as the MFCC filter bank. Optimum subband allocation was not investigated and in future works this area could be further developed for better results.

Appendix A

Review of GMMs

A.1 Gaussian Mixture Models

GMMs are a well known method for modeling the probability distribution of random events. By using K weighted L-dimensional Gaussian functions, it is possible to closely approximate any multi-model distribution [22], provided that enough training data is available. This is particularly useful when a given set of data is from an unknown distribution, as is often the case from real world data.

The complete GMM can be expressed by the mean vector $\vec{\mu_i}$, covariance matrix Σ_i, mixture weights w_i, and number of components K as given below:

$$\Lambda = \{w_i,\ \vec{\mu_i},\ \Sigma_i\}, \quad i = 1, ..., K \tag{A.1}$$

Using the model Λ, we can obtain the likelihood that $\vec{x}$ belongs to the model Λ by

$$p(\vec{x}|\Lambda) = \sum_{i=1}^{K} w_i b_i(\vec{x}), \tag{A.2}$$

where b_i is given by a L-dimensional Gaussian PDF as shown below:

$$b_i(\vec{x}) = \frac{1}{(2\pi)^{L/2}} |\Sigma_i|^{-1/2}\, exp\left\{-\frac{1}{2}(\vec{x}-\vec{\mu_i})^T\, \Sigma_i^{-1}(\vec{x}-\vec{\mu_i})\right\} \tag{A.3}$$

To determine the likelihood that a given feature vector $\vec{x}$ belongs to a model Λ, the logarithm of the associated probability is calculated. This likelihood ($\mathcal{L}$) is expressed by:

$$\mathcal{L} = \log\{p(\vec{x}|\Lambda)\} = \log\left\{\sum_{i=1}^{K} w_i b_i(\vec{x})\right\} \tag{A.4}$$

It is also possible to determine which model, from a set of possible models, provides the best likelihood value; as shown below.

$$\mathcal{L}(\vec{x}) = \max_{i} \{\log p(\vec{x}|\Lambda_i)\} \tag{A.5}$$

Often this process is performed in identification systems in order to select the best model for a given sample.

GMMs can be very effective in modeling the type of distributions found in keystroke patterns, which are shown in Figure 2.1 and in [33]. Additionally, GMMs have been widely used with good success for speaker recognition [64][29].

A.2 Expectation Maximization Algorithm

The expectation maximization (EM) algorithm is often used to estimate the parameters of GMMs [20]. The EM algorithm is a two step iterative algorithm that is guaranteed to converge to a maximum likelihood solution for the parameters of a GMM. However, there is no guarantee that the algorithm converges to a global maxima and therefore, it may converge to any local maxima in the likelihood space. Therefore, the algorithm will converge to the closest local maxima starting from the initial estimate of parameters which must be provided to the algorithm.

The initial estimate of the parameters can lead to different estimates for the same data. Unfortunately, this information is usually not available and therefore it must be estimated. Often the K-Means algorithm with random initialization is used to produce the initial estimate for each GMM cluster. By performing this clustering step several times, a better approximation of the initial cluster can be found. This information can then be given to the EM algorithm.

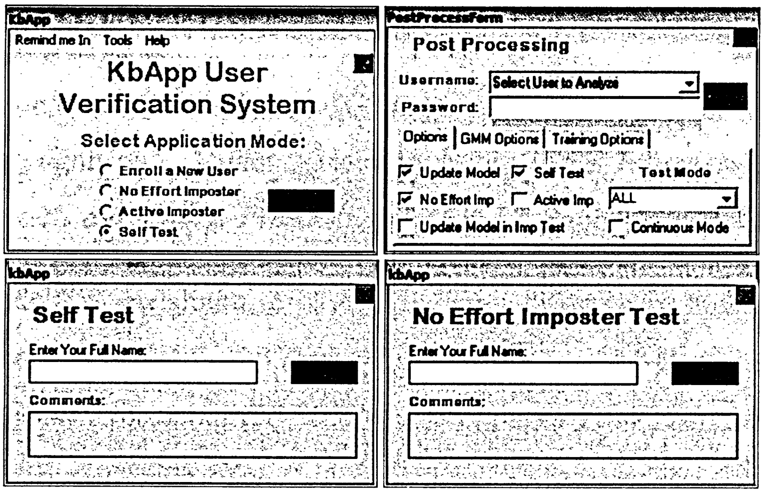
Model order is another important factor that is also not readily known in most applications and therefore, it is often estimated. Rissanen Minimum Description Length (MDL) [66] or the Akaike Information Criteria (AIC) [69] are two popular choices for estimating model order. However, MDL has been shown to be a better estimator of model order for GMMs, performing even better than the AIC when estimating low dimensional models with a small number of samples [46]. Also, in many cases model order has been obtained experimentally.

Appendix B

KbApp Application

KbApp is the especially designed application that was used to collect the experimental data from the users. A few screen shots of *KbApp* are shown in Figure B.1. This application can be installed on any computer with Windows 2000 or later operation system and can gather keystroke timing data directly from the keyboard interrupt handler via a specially designed keyboard driver.

The application has several user modes for gathering test data such as the "Enrollment" mode, "Self Test" mode, "Active Imposter" mode, and "No-Effort Imposter" mode. A



KbApp

Remind me In Tools Help

KbApp User Verification System

Select Application Mode:

○ Enroll a New User
○ No Effort Imposter
○ Active Imposter
● Self Test

PostProcessForm

Post Processing

Username: Select User to Analyze

Password:

Options | GMM Options | Training Options

☑ Update Model ☑ Self Test

Test Mode: ALL

☑ No Effort Imp □ Active Imp

□ Update Model in Imp Test □ Continuous Mode

kbApp

Self Test

Enter Your Full Name:

Comments:

kbApp

No Effort Imposter Test

Enter Your Full Name:

Comments:

Figure B.1: Apperance of *KbApp*, four different windows are shown.

very useful features is the post-processing mode which allows for data to be processed with different parameters. This was especially useful for creating ROC/DET curves.

Appendix C

List of Relevant Publications

Journals

- D. Hosseinzadeh and S. Krishnan, "Keystroke identification as a Biometric". Under review, *IEEE Transactions on Information Forensics and Security*, September 2006.
- D. Hosseinzadeh and S. Krishnan, "On the use of complementary spectral features for speaker recognition". Under review, *EURASIP Journal on Information Security*, September 2006.

Refereed Conferences

- D. Hosseinzadeh and S. Krishnan, "Gaussian mixture modeling of spectral features for speaker recognition". Under review, in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, September 2006.
- D. Hosseinzadeh, S. Krishnan, and A. Khademi, "Keystroke identification based on Gaussian Mixture Models". in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pages 1144-1147, May 2006.

Bibliography

[1] A. Adams and M. A. Sasse, "Users are not the enemy." *Communications of the ACM*, vol. 42, no. 12, pp. 41–46, Dec. 1999.

[2] K. K. Ang and A. C. Kot, "Speaker verification for home security system." in *Proc. IEEE Int'l Symposium on Consumer Electronics (ISCE)*, Dec. 1997, pp. 27–30.

[3] L. C. F. Araújo, L. H. R. Sucupira, M. G. Lizárraga, L. L. Ling, and J. B. T. Yabu-Uti, "User authentication through typing biometrics features." *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 851–855, Feb. 2005.

[4] R. Auckenthaler, E. S. Parris, and M. J. Carey, "Improving a GMM speaker verification system by phonetic weighting." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, Mar. 1999, pp. 313–316.

[5] S. Aviyente and W. J. Williams, "Information bounds for random signals in time-frequency plane." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 6, May 2001, pp. 3549–3552.

[6] S. Bleha, C. Slivinsky, and B. Hussien, "Computer-access security systems using keystroke dynamics." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1217–1222, Dec. 1990.

[7] R. Bose, "Intelligent technologies for managing fraud and identity theft." in *Proc. Third Int'l Conf. Information Technology: New Generations*, Apr. 2006, pp. 446–451.

[8] M. Brown and S. J. Rogers, "User identification via keystroke characteristics of typed names using neural networks." *International Journal of Man-Machine Studies*, vol. 39, no. 6, pp. 999–1014, Dec. 1993.

[9] J. P. Campbell, "Speaker recognition: A tutorial." *Proc. of the IEEE*, vol. 85, no. 9, pp. 1437–1462, Sep. 1997.

[10] W. Chang, "Improving hidden markov models with a similarity histogram for typing pattern biometrics." in *Proc. IEEE Int'l Conf. on Information Reuse and Integration (IRI)*, Aug. 2005, pp. 487–493.

[11] C. Che and Q. Lin, "Speaker recognition using HMM with experiments on the YOHO database." in *Proc. Eurospeech*, Sept. 1995, pp. 625–628.

[12] W. Chen and W. Chang, "Applying hidden markov models to keystroke pattern analysis for password verification." in *Proc. IEEE Int'l Conf. on Information Reuse and Integration (IRI)*, Nov. 2004, pp. 467–474.

[13] N. Clarke, S. Furnell, B. Lines, and P. Reynolds, "Keystroke dynamics on a mobile handset: a feasibility study." *Information Management Computer Security*, vol. 11, no. 4, pp. 161–166, 2003.

[14] J. Colombi, D. Ruck, S. Rogers, M. Oxley, and T. Anderson, "Cohort selection and word grammer effects for speaker recognition." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, May 1996, pp. 85–88.

[15] O. Coltell, J. M. Dabia, and G. Torres, "Biometric identification system based on keyboard filtering." in *Proc. IEEE 33rd Ann. Int'l Carnahan Conf. on Secutrity Technology*, Madrid, Spain, Oct. 1999, pp. 203–209.

[16] W. E. Cooper, Ed., *Cognitive Aspects of Skilled Typewriting.* Springer-Verlag, New York, 1983.

[17] CybèrSource Corp., *Fraudsters Will Take $2.8 Billion out of eCommerce in 2005.* World Wide Web, 2005, available from: http://www.cybersource.com.

[18] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences." *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[19] S. T. de Magalhães, K. Revett, and H. M. D. Santos, "Password secured sites stepping forward with keystroke dynamics." in *Proc. IEEE Int'l Conf. Next Generation Web Services Practices*, Aug. 2005, pp. 224–229.

[20] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[21] G. R. Doddington, M. A. Przybocki, A. F. Martin, and D. A. Reynolds, "The NIST speaker recognition evaluation - overview, methodology, systems, results, perspectives." *Speech Communications*, vol. 31, no. 2-3, pp. 225–254, Jun. 2000.

[22] R. Duda and P. Hart, *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.

[23] M. Faundez-Zanuy and E. Monte-Moreno, "State-of-the-art in speaker recognition." *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 5, pp. 7–12, May 2005.

[24] P. Flandrin, R. G. Baraniuk, and O. Michel, "Time-frequency complexity and information." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, Apr. 1994, pp. 329–332.

[25] K. Fukunaga, *Introduction to Statistical Pattern Recognition (2nd ed.).* Academic Press Professional Inc., San Diego, CA, USA, 1990.

[26] R. Gaines, W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: Some preliminary results." Rand Corporation, Santa Monica, CA, USA, Tech. Rep. R-256-NSF, May 1980.

[27] M. Gasser, *How Language Works, (edition 3.0).* World Wide Web, 2006, available from: http://www.indiana.edu/~hlw/V3/, Last Accessed: August 17, 2006.

[28] H. Gish and M. Schmidt, "Text-independent speaker identification." *IEEE Signal Processing Magazine*, vol. 11, no. 4, pp. 18–32, Oct. 1994.

[29] J. Gonzalez-Rodriguez, S. Gruz-Llanas, and J. Ortega-Garcia, "Biometric identification through speaker verification over telephone lines." in *Proc. IEEE Int'l Carnahan Conf. on Security Technology*, Oct. 1999, pp. 238–242.

[30] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "Compensation for the effect of the communication channel in auditory-like analysis of speech (RASTA-PLP)." in *Proc. Eurospeech*, Dec. 1991, pp. 1367–1370.

[31] C. Hicks, *Fundamental Concepts in the Design of Experiments.* Ney York: CBS College, 1982.

[32] S. Hocquet, J.-Y. Ramel, and H. Cardot, "Fusion of methods for keystroke dynamic authentication." in *Proc. IEEE Workshop on Automatic Identification Advanced Technologies*, Oct. 2005, pp. 224–229.

[33] D. Hosseinzadeh, S. Krishnan, and A. Khademi, "Keystoke identification based on gaussian mixture models." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, Toulouse, France, May 2006, pp. 1144–1147.

[34] International Biometric Group LLC, *Biometrics Market and Industry Report 2006-2010.* World Wide Web, 2006, available from: http://www.biometricgroup.com.

[35] A. K. Jain and S. Pankanti, "Biometric authentication systems for credit cards could put identity thieves out of business." *IEEE Spectrum*, pp. 22–27, Jul. 2006.

[36] R. Joyce and G. Gupta, "Identity authentication based on keystroke latencies." *Communication of the ACM*, vol. 33, no. 2, pp. 168–176, Feb. 1990.

[37] A. Kale, A. Sundaresan, A. Rajagopalan, N. Cuntoor, A. Roy-Chowdhury, V. Kruger, and R. Chellappa, "Identification of humans using gait." *IEEE Trans. on Image Processing*, vol. 13, no. 9, pp. 1163–1173, Sep. 2004.

[38] J. J. Leggett and G. Williams, "Verifying identity via keystroke characteristics." *International Journal of Man-Machine Studies*, vol. 28, no. 1, pp. 67–76, 1988.

[39] J. J. Leggett, G. Williams, M. Usnick, and M. Longnecker, "Dynamic identity verification via keystroke characteristics." *International Journal of Man-Machine Studies*, vol. 35, no. 6, pp. 859–870, 1991.

[40] D.-T. Lin, "Computer-access authentication with neural network based keystroke identity verification." in *Proc. IEEE Int'l. Conf. on Neural Networks*, vol. 1, Jun. 1997, pp. 174–178.

[41] S. Liu and M. Silverman, "A practical guide to biometric security technology." *IEEE IT Professional*, vol. 3, no. 1, pp. 27–32, Jan. 2001.

[42] D. Mahar, R. Napier, M. Wagner, W. Laverty, R. Henderson, and M. Hiron, "Optimizing digraph-latency based biometric typist verification systems: inter and intra typist differences in digraph latency distributions." *Int'l Journal of Human-Computer. Studies*, vol. 43, no. 4, pp. 579–592, 1995.

[43] S. Mandujano and R. Soto, "Deterring password sharing: User authentication via fuzzy c-means clustering applied to keystroke biometric data." in *5th Mexican Int. Conf. in Computer Science (ENC'04)*, Sept. 2004, pp. 181–187.

[44] A. J. Mansfield and J. L. Wayman, "Best practices in testing and reporting performance of biometric devices." Centre for Mathematics and Scientific Computing, National Physical Laboratory, Middlesex, UK, Tech. Rep. NPL Report CMSC 14/02, Aug. 2002.

[45] K. Markov and S. Nakagawa, "Frame level likelihood normalization for text-independent speaker identification using gaussian mixture models." in *Proc. IEEE Int'l Conf. on Spoken Language Processing(ICSLP)*, vol. 3, Oct. 1996, pp. 1764–1767.

[46] P. McKenzie and M. Alder, "Selecting the optimal number of components for a gaussian mixture model." in *Proc. IEEE Int'l Symposium on Information Theory*, Jun. 1994, p. 393.

[47] F. Monrose and A. Rubin, "Keystroke dynamics as a biometric for authentication." *Future Generation Computer Systems (FGCS)*, vol. 16, no. 4, pp. 351–359, Feb. 2000.

[48] D. S. Moore and G. P. McCabe, *Introduction to the Practice of Statistics (3rd ed.)*. W.H. Freeman and Company, New York, USA, 1999.

[49] J. M. Naik, "Speaker verification: A tutorial." *IEEE Communications Magazine*, vol. 28, no. 1, pp. 42–48, Jan. 1990.

[50] National Institute of Standards and Technology (NIST), "The DARPA TIMIT acoustic-phonetic continuous speech corpus." Oct. 1990, speech Disc CD1-1.1.

[51] M. N. Nazar, "Speaker identification using cepstral analysis." in *Proc. IEEE on Students Conference (ISCON)*, vol. 1, Aug. 2002, pp. 139–143.

[52] M. Obaidat and B. Sadoun, "Verification of computer users using keystroke dynamics." *IEEE Transactions on Systems, Man and Cybernetics - Part B*, vol. 27, no. 2, pp. 261–269, Apr. 1997.

[53] D. O'Shaughnessy, *Speech Communication: Human and Machine.* Addison-Wesley, Reading, MA, 1987.

[54] K. K. Paliwal, "Spectral subband centroid features for speech recognition." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, May 1998, pp. 617–620.

[55] A. Peacock, X. Ke, and M. Wilkerson, "Typing patterns: A key to user identification." *IEEE Security & Privacy Magazie*, vol. 2, no. 5, pp. 40–47, Oct. 2004.

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

[57] S. Prabhakar, S. Pankanti, and A. K. Jain, "Biometric recognition: Security and privacy concerns." *IEEE Security & Privacy Magazine*, vol. 1, no. 2, pp. 33–42, Mar. 2003.

[58] M. A. Przybocki and A. F. Martin, "NIST speaker recognition evaluation." in *Proc. Speaker Recognition and its Commercial and Forensic Applications (RLA2C))*, Oct. 1996, pp. 1764–1767.

[59] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition.* Prentice Hall, Englewood Cliffs, New Jersey, 1993.

[60] A. Ramalingam, "Signal processing techniques for multimedia information security." Master's thesis, Ryerson Univeristy, Dept. of Electrical and Computer Engineering, Toronto, Canada, 2005.

[61] D. A. Reynolds, "Automatic speaker recognition using gaussian mixture speaker models." *Lincoln Lab. Journal*, vol. 8, no. 2, p. 173192, 1995.

[62] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixtures models." *Digital Signal Processing*, vol. 10, pp. 19–41, Jan. 2000.

[63] D. A. Reynolds and R. Rose, "An integrated speech-background model for robust speaker identification." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, Mar. 1992, pp. 185–188.

[64] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models." *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, Jan. 1995.

[65] D. R. Richards, "Rules of thumb for biometric systems." *Security Management*, Oct. 1995.

[66] J. Rissanen, *Stochastic Complexity in Statistical Inquiry.* World Scientific Publishing Co. USA, 1989.

[67] A. E. Rosenberg, J. DeLong, C. H. Lee, B. H. Juang, and F. K. Soong, "The use of cohort normalized scores for speaker verification." in *Int'l Conf. on Speech and Language Processing (ICSLP)*, vol. 10, Nov. 1992, pp. 599–602.

[68] J. D. Sachs, *The End of Poverty: Economic Possibilities for Our Time.* The Penguin Press, USA, 2005.

[69] Y. Sakimoto, M. Ishiguro, and G. Kitagawa, *Akaike Information Criterion Statistics.* KTK Scientific Publishers, Tokyo, 1986.

[70] L. H. Shaffer, *Latency Experiments in Transcription*, ser. Attention and Performance, Edited by Sylvan Kornblum. Academic Press, New York, 1973, vol. 4.

[71] L. H. Shaffer and J. Hardwick, "The basis of transcription skill." *Journal of Experimental Psychology*, vol. 84, no. 3, pp. 424–440, 1970.

[72] M. J. Sniffen, *Pentagon anti-terror surveillance system hopes to identify people by the way they walk.* World Wide Web, 2003, available from: http://www.securityfocus.com/news/4909.

[73] F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang, "A vector quantization approach to speaker recognition." in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Apr. 1985, pp. 387–390.

[74] R. Spillane, "Keyboard apparatus for personal identification." *IBM Technical Disclosure Bulletin*, vol. 17, no. 3346, 1975.

[75] A. Teoh, S. A. Samad, and A. Hussain, “An internet based speech biometric verification system.” in *Proc. IEEE 9th Asia-Pacific Conference on Communications (APCC)*, vol. 1, Sept. 2003, pp. 47–51.

[76] E. Thomas and R. Jones, “A model for subjective grouping in typewriting.” *Quarterly Journal of Experimental Psychology*, vol. 22, pp. 354–367, 1970.

[77] N. Z. Tishby, “On the application of mixture ar hidden markov models to text independent speaker recognition.” *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 39, no. 3, pp. 563–570, Mar. 1991.

[78] D. A. Umphress and G. Williams, “Identity verification through keyboard characteristics.” *International Journal of Man-Machine Studies*, vol. 23, no. 3, pp. 263–273, 1985.

[79] R. Vergin, D. O’Shaughnessy, and A. Farhat, “Generalized mel frequency cepstral coefficients for large-vocabulary speaker-independent continuous-speech recognition.” *IEEE Trans. on Speech and Audio Processing*, vol. 7, no. 5, pp. 525–532, Sep. 1999.

[80] J. D. Woodward, “Biometrics: privacy’s foe or privacy’s friend?.” *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1480–1492, Sept. 1997.

[81] R. E. Yantorno, K. R. Krishnamachari, and J. M. Lovekin, “The spectral autocorrelation peak valley ratio (SAPVR) a usable speech measure employed as a co-channel detection system.” in *Proc. IEEE Int’l Workshop on Intelligent Signal Processing (WISP)*, May 2001.

[82] E. Yu and S. Cho, “GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification,” in *Proc. IEEE Int’l Joint Conf. on Neural Networks*, vol. 3, Jul. 2003, pp. 2253–2257.

[83] R. Zheng, S. Zhang, and B. Xu, “Text-independent speaker identification using GMM-UBM and frame level likelihood normalization.” in *Proc. Int’l Symposium on Chinese Spoken Language Processing*, Dec. 2004, pp. 289–292.