5-2014

A Novel Authentication Method Using Multi-Factor Eye Gaze

Lucas A. Herrera
University of Tennessee - Knoxville, trex769@gmail.com

To the Graduate Council:

I am submitting herewith a thesis written by Lucas A. Herrera entitled "A Novel Authentication Method Using Multi-Factor Eye Gaze." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Engineering.

J. Douglas Birdwell, Major Professor

We have read this thesis and recommend its acceptance:

Tsewei Wang, David Icove

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

5-2014

A Novel Authentication Method Using Multi-Factor Eye Gaze

Lucas A. Herrera
University of Tennessee - Knoxville, lherrer1@utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Lucas A. Herrera entitled "A Novel Authentication Method Using Multi-Factor Eye Gaze." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Engineering.

J. Douglas Birdwell, Major Professor

We have read this thesis and recommend its acceptance:

Tsewei Wang, David Icove

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

A Novel Authentication Method Using Multi-Factor Eye Gaze

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Lucas A. Herrera

May 2014

This work is dedicated to Dr. Juan and Kim Herrera.

Thank you for your unwavering love, constant support, and firm guidance.

Acknowledgements

I would like to express my deepest thanks to my Major Advisor Dr. J Douglas Birdwell for his technical guidance and support of this ambitious research through hours of meetings and discussions that directly contributed to its merit. I must also thank my committee members Dr. Tsewei Wang for her constructive criticism and valuable input to the content of this thesis and Dr. David Icove for his enthusiasm and support of this work.

To my Kaki, thank you for your love and support during this difficult and trying journey. I am forever grateful to my parents for believing in me during the many times they alone have stood behind me. Finally, I thank Jesus, my Confidence and Peace, who will always be the driving force in my life.

An investment in knowledge pays the best interest.

-Benjamin Franklin

Abstract

A method for novel, rapid and robust one-step multi-factor authentication of a user is presented, employing multi-factor eye gaze. The mobile environment presents challenges that render the conventional password model obsolete. The primary goal is to offer an authentication method that competitively replaces the password, while offering improved security and usability. This method and apparatus combine the smooth operation of biometric authentication with the protection of knowledge based authentication to robustly authenticate a user and secure information on a mobile device in a manner that is easily used and requires no external hardware. This work demonstrates a solution comprised of a pupil segmentation algorithm, gaze estimation, and an innovative application that allows a user to authenticate oneself using gaze as the interaction medium.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Advances in mobile computing and hardware platforms have enabled mobile devices to become extensions of their users. The category of mobile devices includes smart phones, tablets, ultrabooks, and other products that combine Internet access with flexibility and mobility. Mobile application and service developers capitalize on these dynamic platforms by providing convenient applications and an interface to the Internet. The trade-off for this high flexibility and mobility is a unique set of security challenges. Cryptographic systems have struggled in several aspects, including ease of use and power consumption, and the user component that these cryptographic systems rely on continues to be the password. Users can now access financial, personal, health, and otherwise confidential information using their mobile devices, but security professionals, at least since 1979, have recognized the need for improved authentication [32].

The fundamental problem facing password implementations lies in the human factor. As O'Gorman alludes to in [34], the strongest vault can be attacked by exploiting a human mistake, just as the strongest encryption algorithm can be attacked by exploiting a weak password. The fact that the user is responsible for the password means that any password-based cryptographic system is a single point of failure once the password is compromised. For this reason, strong password choices

are those that are sufficiently long and complex enough to resist social engineering. Such passwords conflict with the limitations of human memory, and users resort to either writing down their passwords or making shorter, thus weaker, passwords. Many systems have been compromised due to poor security implementations and tenacious attackers, further demonstrating the gravity of the situation.

In order to protect the information that is being stored on mobile devices and web servers, improved authentication steps are needed. Many works have focused on developing stronger authentication processes, but security professionals do not typically focus on usability, and application developers do not focus on security. Bridging this gap has proved difficult [4, 15, 34]. The authentication scheme that will replace the pervasive password must offer improved security with the ease and convenience that passwords offer. Researchers and industry professionals cannot agree on how to improve the authentication process for mobile device users. Many researchers believe that multi-factor authentication is the answer, given the attacks and vulnerabilities associated with password approaches.

However, multi-factor schemes usually require added equipment and are expensive to implement [27]. The two-step approaches that the industry has adopted do not adequately improve security and are rarely used as they are optional for most applications and services [27]. This work seeks to remedy the disadvantages of these approaches by combining the usability of passwords and the security offered by multi-factor authentication through a system using gaze pattern detection and estimation. Using the eyes for human-device interaction, by employing gaze estimation, allows for subtle, inconspicuous movements that are difficult for third parties to detect and intercept. Simultaneously, iris and other identifying biometric information can be gathered from a user's ocular regions. The goal of this work is to demonstrate the improvements of the security associated with accessing information on a mobile device or through a web interface accessed by a mobile device.

1.1 Outline

This work proposes a novel replacement for the password by combining the use of biometric and password security components to provide a highly usable authentication procedure that accomplishes multi-factor authentication in one step. This authentication scheme specifically addresses user authentication to the mobile device, allowing the device to identify the user with greater certainty providing appropriate access. This research will contribute to the overall effort of *user authentication in mobile devices*.

Chapter 2 summarizes the technical and research background surrounding this work including the prior art concerning authentication, feature recognition, and gaze estimation. Chapter 3 describes the design of the system and the evaluation criteria that will be used to describe the performance of the system. Chapter 4 details the development and implementation of a software application used to demonstrate the practicality of the approach. The conclusions and future work are identified in Chapter 5.

Chapter 2

Technology and Literature Review

This chapter gives background of and summarizes the prior art in the related fields of user authentication and eye gaze tracking and estimation.

2.1 Authentication Factors

Validating the identity of a user, commonly referred to as user authentication, can be achieved through the verification of one or more of three questions:

- What does the user know?
- What does the user have?
- What is the user?

For a more in-depth consideration of the three authentication factors, please consult Appendix A.

2.2 Single Factor Authentication

While the front end of most password authentication systems might look the same, sharing an entry box for the username and password in common, implementations of

knowledge-based authentication differ greatly, with some systems having more secure password management and communication than others. Although trust is a major component to any authentication protocol, this is especially true for a knowledge based system, where security of the system relies on the existence and protection of a shared secret.

Since keeping the password secret is the mutual responsibility of the user and the application developer, Almuairifi, Veeraraghvan, and Chilamkurti proposed in [2] a system seeking to minimize the user's responsibility by expanding on a knowledge-based authentication scheme using graphical representations of passwords. This system relies on contextual information from the password and is inherently vulnerable to dictionary attacks, since an attacker observing the systems responses to multiple attacks could discern the context and make well-educated attempts. Over-the-shoulder, dictionary, and other social engineering attacks exploit the dependence of knowledge-based authentication on human factors. In addition, considering the web-enhanced nature of mobile devices, more frequent authentications of users to their devices lays the groundwork of a convincing case for the development of stronger, user-friendly authentication. As Skracic, Pale, and Jeren discuss in [37], dynamically changing passwords may offer increased security; however, simple knowledge-based authentication schemes continue to offer users and designers the best combination of usability, scalability, and security in spite of their inherent vulnerabilities.

Defigueiredo sums up the need for a mobile two factor authentication solution in [7], by explaining that mobile device authentication provides a unique set of design constraints, which expose problems never addressed by desktop authentication systems, such as device loss and phishing. For a desktop system, loss is unlikely and phishing risk can be reduced by securing access at a software level, through an operating system or third-party application. Some laptops have integrated fingerprint scanners and smartcard readers, but widespread use has not been achieved, as these components offer little additional functionality and increase manufacturing costs. Since the vast majority of mobile applications require web access and some form

of authentication, mobile device users are bombarded with authentication requests from the device or the web service, preventing the current security solutions from being ideal for mobile device applications.

2.2.1 Industry Deployed Approaches

Today, little is known about the specifications, but Apple Inc. and Google have developed secure hardware components that allow embedded sensors to access information that is completely secure from other hardware components. Security is improved through exclusive bus lines that only communicate between a secure cache element on the CPU and a sensor on the device. Google has implemented this technology associated with a Near Field Communication chip that has secure bus lines to the CPU to communicate secure information.

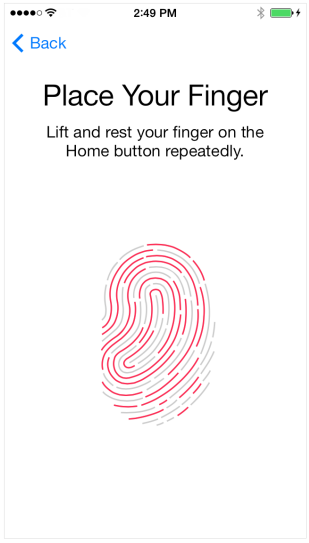


Figure 2.1: The training screen for Apple Inc.'s TouchID, a fingerprint-based authentication system that debuted on the iPhone 5s in October 2013.

Apple has implemented this technology using a special imaging device to capture unique points on a user's finger. This technology has been marketed as TouchID and has been integrated into the iPhone 5S. Figure 2.1 displays a feedback screen shown to a user during a TouchID authentication attempt. The imaging chip has secure bus lines and dedicated cache elements to securely handle the information.

Unfortunately for users of TouchID, fingerprints can be replicated using household materials or by lifting a fingerprint from the devices case or glass screen [2, 24]. Despite the potential security risks, TouchID debuted as the most popular biometric authentication implementation. This widespread disregard of risk can be most likely attributed to the consumer loyalty and trust in the Apple Inc. brand.

2.3 Multi-factor Authentication Implementations

A multi-factor authentication (MFA) procedure requires challenging at least two of the authentication factors from Table 2.1. This requirement is essential in providing improved security over a one or two password system. Combining two factors provides added security, but more often than not, this comes at a high cost to the usability or scalability of the authentication scheme.

Table 2.1: Authentication factors

Factor	Description	Practical Applications
Knowledge	Something only the user knows	password, PIN
Possession	Something only the user has	key, ID card, token
Inherence	Something only the user is	biometrics (fingerprints, iris, voice)

To security professionals, the shortcomings of user-dependent passwords more than demonstrate the need for a viable alternative, but the reluctance of businesses and users to embrace more secure alternatives proves that the benefits of upgraded security do not yet outweigh the costs of reduced usability, increased complexity, and complete overhaul of the existing authentication system [27]. For this reason, Mao, Florencio, and Herley, professionals from the technology industry, have

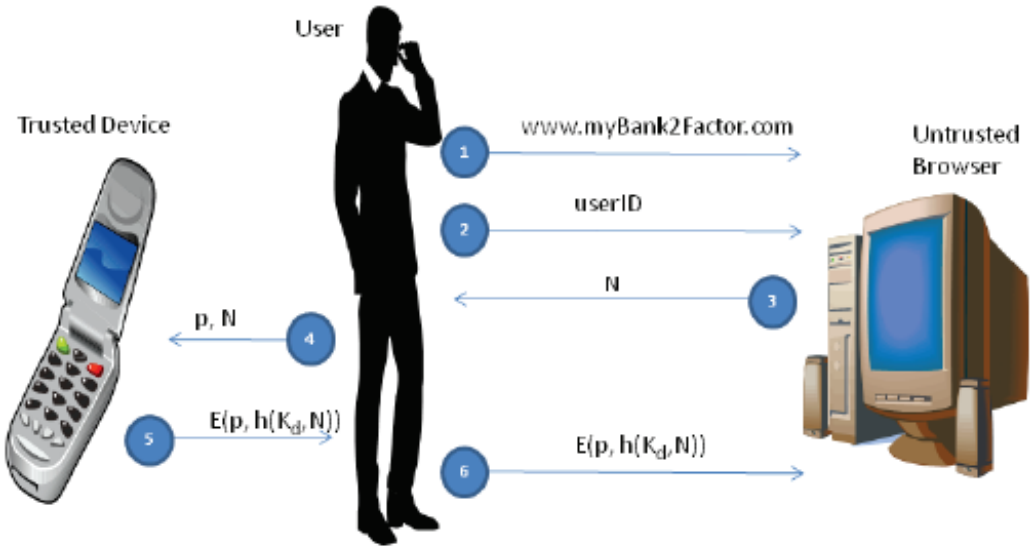


Figure 2.2: The protocol outline for the user experience of a system using the authentication method described in [27].

collaborated to propose the method in [27], which mitigates the disruptive change concerning an upgrade to a multi-factor system by adding a possession-based layer on top of a password system in the form of an additional server that verifies the possession of a trusted device. The possession verification via a secure PIN communicated from the additional server to an authenticating device, usually a mobile phone, uses a messaging medium such as voicemail, text, or email. All of this communication is triggered by a successful authentication through the existing framework. Figure 2.2 illustrates the messages that are sent between the devices involved in an authentication attempt using the Mao et al. method.

By requiring no removal of the legacy authentication framework and placing emphasis on *ease of integration*, this approach is the most widespread implementation of MFA. Businesses have embraced this system, since its perceived security and low implementation overhead offers the least cost for an improved system, and further places emphasis on the need of a system to consider implementation costs as a priority. This system compromises potential multi-factor authentication (MFA) security by relying on possession of a trusted device and prioritizing integration over usability by

authenticating in two distinct steps, as opposed to one. Combine the typical shoulder surfing attacks that plague mobile devices with the fact that trusted mobile devices are not always in the possession of their trusted users, and the result is a set of feasible and crippling attacks that exploit this system and the essence of mobile devices.

Other common authentication procedures claiming to achieve MFA employ only one authentication factor with multiple challenges, such as asking for a password and the answer to a challenge question. Authenticating in this fashion only validates the knowledge component of a user's identity. A procedure such as this is more accurately termed strong authentication, and the security offered does not fully benefit from a true MFA scheme.

Some MFA designs, such as that proposed by Liou, Egan, Patel, and Bhashyam in [24], use a combination of the knowledge (password) and possession (cell phone or token) authentication factors. These designs have a two-phase identification process where the knowledge component and possession components are challenged not only independently, but also separately. This leads to a more cumbersome authentication step than a single factor design, and deters users not wanting to sacrifice the convenience of their mobile device for the security of their information. Fundamentally, this added step in two-step strategies does not lend itself well to the desirable trait of authentication systems to disappear into the use of a service.

The focus of Tiwari, Sud. Sanyal, Abraham, Knapskog, and Sug. Sanyal in [39] is to use a Transaction Identification Code and the Short Message Service of mobile devices to realize a multi-factor scheme. The system is designed to support mobile transactions and highly secure communication between banking servers, mobile devices, and Point-of-Sale (POS) machines. Similarly, the method implemented by Vipin, Sarad, and Sankar [44] relies on a knowledge secret and a secret generated by a possessed token to authenticate users in a mobile commerce environment. Similar to the systems discussed earlier, these two systems require multiple steps to fully authenticate, and do not offer users the necessary convenience to replace passwords.

More importantly, possession factors have been shown to be difficult for users to manage and should be used conservatively for mobile device authentication.

Phiri, Zhao, and Agbinya [35] introduce a novel approach to compose one authentication factor through the fusion of biometrics (fingerprint), device metrics, and pseudo metrics. The authors employ a combinatorial neural network that is trained to implement the authentication by reaching the activation potential when a certainty level has been achieved. Unfortunately, biometric data introduce a level of uncertainty that must be managed, but compounding that uncertainty in a fusion approach may greatly increase the probability of a false-positive. Generally, given the adaptive thresholding steps, neural networks are ill-suited for robust authentication systems. Sun, Li, Jiang, and Kot implement MFA in [38] by sending two or three images of a user's face to an image database, computing a 40 digit hash from the user's face, and combining that with an image-based password. This approach indeed prevents over-the-shoulder attacks and allows for greater security; however, using a server for authenticating a user to a mobile device presents an obstacle for users attempting to access devices not connected to a wireless network, excluding this method from competition with the traditional password.

The comprehensive scheme proposed by Huang, Xiang, Chonka, Zhou, and Deng [17] authenticates users based on verified password knowledge, smart card possession, and abstract biometric characteristics. Many concerns are addressed in this true three-factor approach, yet it omits the description of how biometric data are acquired. Fan and Lin [9] also propose a three factor system, combining a password with a smart card and fingerprint. Just as Huang, et al.[17] lack an adequate mobile variety or feasibility for mobile platforms, this work, too, suffers from the use of an authentication server.

Ocular multi-factor approaches have been proposed by Millan, Perez-Cabre, and Javidi [30] via a system using retinal images in response to specific images stored on an ID token or card to authenticate users by imaging the user's retina *in situ*. While this approach does offer a high degree of security, it requires expensive external

imaging equipment and computational workloads difficult to integrate into current mobile platforms. None of the current authentication technology sufficiently combines authentication factors in such a way that enables the usability of passwords and the security of MFA in a system that is practical for mobile devices.

The approach of this work proves the feasibility of a novel authentication method between a user and a mobile device. The method estimates the user's gaze point– where the user is looking on the mobile device's screen. Before the user's gaze may be estimated, the face and eye regions of the user must be detected. The following sections describe methods to detect the face and eye regions, as well as estimate the user's gaze.

2.4 Viola-Jones Feature Detection

Using the embedded user-facing camera available on the majority of mobile devices, biometric information can be collected and used for authentication. Biometric information extracted from an image exhibits unique characteristics that can be abstracted as features within an image. Biometric features used for authentication must be identified in a deterministic fashion. This section describes the facial feature detection method used in this work.

Haar features are two-dimensional image features extracted by evaluating the integral of the image, or the sum of the image intensities, within rectangular sections of varying scales within the image. The rectangular image sections are selected by rectangular filters that seek out areas of dark intensity near areas of light intensity oriented in the same fashion as the filters. Viola and Jones designed a set of optimizations, known as Viola-Jones feature detection, that allow multiple filter stages to be developed and cascaded using AdaBoost training and tuning to provide real-time facial feature detection [42].

Viola-Jones utilizes a set of classifiers that are ordered to allow for rapid and robust object detection. The ordering is done by offline training using AdaBoost, a machine

learning algorithm. AdaBoost weights several imprecise classifiers that combine to provide a rapid and robust classifier, termed a *cascade* by Viola and Jones. The classifiers are organized by weight according to the false acceptance rate measured by AdaBoost measured on a training set, with the first filters allowing the greatest false positive rate and the last filters allowing a minimal false positive rate. The goal of this ordering is to optimize the performance of feature detection for speed and robustness. The first stage selects all candidates, while the succeeding stages progressively cull through the candidate set, rejecting ill-fitting members at each stage. This method will be used to detect the face and eyes of users during the process of authentication.

2.5 Gaze Estimation via Face Detection and Eye Tracking

Recently, gaze tracking has garnered considerable attention in the fields of human-computer interaction (HCI) and biometrics. Eye-gaze has been identified for several applications, such as gaming interactions in [5], fatigued driver recognition in [29], paraplegic assistance in [40]. Achieving reliable gaze estimation relies on the combination of face detection, eye region detection, and pupil or iris tracking. Haar cascades created by the Viola-Jones method can be used for both the face and the eye region detection.

Face Detection

Work done by Ephraim, Himmelman, and Siddiqi in [8] attests to the fast performance of Viola-Jones detection algorithm. The authors were able to embed the algorithm using a slow scripting language in a web browser. The performance of the facial detection algorithm suffered little degradation in the scripting environment, and detection rates hovered above 90%. The work done by Mei, Liu, Li, and Yang in [29], Udayashankar, Kowski, Chandramouli, and Prashanth in [40], and Jiang, Lu,

Tang, and Goto in [18], all incorporate Haar features to quickly detect faces in real-time video. In all three approaches, Viola-Jones face detection algorithm is used successfully, followed by tracking of the eyes using image processing techniques.

Pupil Tracking

Yan, Gao, and Zhang [45] use glint detection to track the pupil, and obtain accurate, though not precise, results (78% accuracy) using the Hough transform to locate the circle representing the pupil. Several other works, including the work of C. Yang, Sun, J. Liu, X. Yang, Wang, and W. Liu [46, 48] use glint detection under adequate lighting conditions to simplify tracking the pupil. Without special lighting conditions, the glint is not reflected in a deterministic fashion and cannot be relied upon to track the pupil region of the eye, so a novel method must be used to accommodate natural lighting conditions.

Gaze Estimation

The efforts of Hennessey and Lawrence, with contributions from Noureddin [12, 13, 14] estimate the point-of-gaze (POG, the subject's focal point in 3-D space) via off-axis infrared light sources and image processing of the corneal reflections those light sources produce. The results of this system support the most accurate POG estimation of all those considered, but the system requires an abundance of light sources mounted at the correct angles. Just as with the pupil detection methods described earlier, the directed lighting required for this method renders it infeasible for consideration in this algorithm.

Accurate gaze estimation that is head-movement tolerant and mobile platform friendly has yet to be developed. For this reason, a major goal of the work reported herein is the development of accurate gaze estimation to accommodate mobile devices with neither external hardware nor special lighting conditions.

2.6 Gaze-based Authentication

Bednarik, Kinnunen, Mihalia, and Franti [3] studied gaze-tracking as an authentication factor for desktop computers. Using acceleration and gaze velocity of pupil movement, as well as its size, the authors observed 60 percent accuracy, but the eye-movements were tracked using infrared light and algorithms that were intolerant of blinking and head movement. Later, Liang, Tan, and Chi [23] expanded on the work in [3] by measuring acceleration, geometry, and muscle information of the ocular region, to provide 34 features to a classifier. The classifier method discriminates users based on the commonality of the transient response of the eyes to a stimulus video. Maeder and Fookes [25] use the stimulation from a specific visual scene to measure the unique response of a user's gaze– the points in the images where the user focuses on– and iris width for identification purposes. These methods require high resolution, continuous images of the eyes to accurately measure fine eye movements, known as saccades, cannot be maintained by mobile devices.

De Luca, Weiss, and Drewes [6] proposed a novel method of PIN-entry using eye-gaze. The motivation of the work is to mitigate the widely accepted risk of an attack known as shoulder surfing, whereby an active observer memorizes the PIN during a user's traditional and viewable keypad entry. Kumar, Garfinkel, Boneh, and Winograd [22] and Kasprowski and Ober [21] address this issue as well, with each effort showing, respectively through user studies, that gaze-based entry methods are preferred by a majority of users to protect against shoulder surfing attacks.

The method of [6] presents a drawing pad to a user where, drawing with their eyes, they are able to enter their password. Although an eye-centric interface is the goal of the work described herein, the method in [6] requires large and expensive equipment, as well as a stationary device, such as an ATM, and does not directly or indirectly represent a feasible solution for the mobile environment. Similarly, the gaze-based password entry system proposed in [22] requires a stationary camera, is designed for desktop use, and does not provide a feasible basis for mobile devices. Additionally,

none of the previously observed gaze-based methods provide a multi-factor approach to mobile device authentication.

Iris Scanning Techniques

Although an intriguing possibility as high resolution imaging continues to advance, iris scanning techniques are not in the scope of this work, as they do not lend themselves to a mobile platform without embedding specialized hardware.

2.7 Summary

The prior methods developed for gaze-based authentication or multi-factor authentication do not present feasible authentication options for use in mobile devices. The factor limiting the use of any previously developed methods is gaze estimation under natural lighting conditions running on a mobile device. The following chapter will describe the development and implementation of a gaze estimation algorithm that is expected to facilitate a stable and reliable human-computer interface to provide a novel method of user-device authentication. The design of the gaze estimation algorithm will leverage Viola-Jones feature detection to first detect the face and eyes of a user. The existing research on pupil tracking and gaze estimation in mobile devices must be furthered to develop a feasible solution subject to the constraints of the mobile environment. The next chapter details the design of a system that performs real-time image processing to estimate the gaze point of a user combined with a minimally observable interface to offer users an innovative authentication process.

Chapter 3

Design

This chapter details the design of an application that attempts to deliver gaze-based multi-factor authentication and to satisfy the constraints presented by the mobile environment.

3.1 Background

As standard computers become more portable, mobile devices continue to permeate all aspects of life, and corporations continue to amass more and more data, the need for strong, robust security increases. Current security measures are always under duress, and they are breaking. The security of the everyday user must be seamlessly integrated into the usage model, so that the user is encouraged to embrace security as opposed to spurning it through disabling the features, or effectively circumventing them by using simple or trivial passwords. The current password model will be very difficult to replace, but the time has come.

Just like the computer it protects, a password is an invention that modern life has embraced to the point that the associated obstacles are effortlessly overcome. People no longer wonder why their hard drive has died. It is just accepted that hard drives will fail, and people may be proactive by backing up their data to prevent and mitigate the damage. Similarly, people appear to know that every online and computer account

they have will need a username and password. It is accepted that upon creating any account, at a minimum, an email address, or other identifier, and a password will be needed. Unfortunately, by settling for passwords as the industry standard, people have acquiesced to an authentication model that is no longer sufficient to secure their information, and perhaps never was.

Corporations need to have strong authentication as the first step to secure the data of their users, but more security often implies a cumbersome and slow authentication process. This characteristic permeates authentication systems, where stronger authentication generally means more time to authenticate. Today's users want fast authentication more than secure authentication, and complicated authentication steps encourage users to compromise the security potential of the authentication process, such as by choosing easy passwords or writing them down. The average technology user assumes that the simple use of a password is enough to protect their data, and anything more than that is an unjustified hassle. This demonstrates the lack of understanding that users have of the vulnerabilities of their technologies, and this gap prevents extra authentication steps from being justified in the eyes of the casual user. As computers, tablets, smart phones, and other mobile devices become more portable and functional, new uses, such as electronic medical records and personal financial data, are arising that necessitate stronger authentication schemes. This work proposes an authentication system that embraces the current mobile environment and allows users to securely authenticate themselves to their devices.

For mobile applications, where security is not the only consideration, users are very concerned with aesthetics and usability [1]. Eliminating adoption obstacles for this technology further supports the goal of this system to adequately replace the popular, yet insufficient, password approaches currently employed on mobile devices. The design must support the primary goal of the system to authenticate users using a one-step multi-factor approach. Every aspect must accommodate the limitations of mobile deployed applications, while taking into account user expectations of convenience and effortless functionality. The details of the system should provide a reliable user

experience by avoiding prohibitive properties such as errors (false-negatives and false-positives) and confusing interfaces. Corporate security measures are proving to be inadequate deterrents and defenses to persistent attackers. The major contribution of this system leverages a two-factor scheme that provides the usability and experience of a one-step password scheme, thus reducing the vulnerabilities of previous systems. As developers increasingly disguise web services as mobile applications, the user authentication steps must uphold a level of security that can protect the accounts of the system and entire user populations. The largest consideration of this design is that the authentication achieve two-factor confidence in one seamless step– either granting or prohibiting access after one fluid motion.

Endpoint or *device authentication* is a promising method to secure the backend servers and data warehouses. Interacting with the device through touch allows neighboring people to view the exact authentication actions of the user. Device authentication requires an authentication interaction with the device, and for this reason, a new mode of interaction is needed. A user's gaze does not allow neighbors to view the authentication interactions. The proposed system will offer a novel user interface designed to protect the interactions of the user, and will offer a medium to provide the security of MFA. The proposed authentication scheme combines knowledge and inherence factors to authenticate users using eye movements to enter a password. The usability and security of this authentication method hinges on the user interface.

3.2 User Experience Considerations

To promote adoption of this method, the experience of existing password interfaces will be preserved to the utmost, with the exception of the interaction medium. The user selects a personal identification number (PIN) composed of any number (above a specified minimum) of digits (0-9), and an integrated camera provides images to a gaze estimation algorithm. Once the gaze point is established, an estimation algorithm

projects the gaze point onto the device's screen, enabling the user to interact with the device and enter the PIN expressed as a sequence of blocks occupying specific positions corresponding to the symbols of a normal keypad layout. As an added layer of security, random input feedback is given to the user until authentication is complete. The random input feedback is provided through colored blocks that shuffle on the screen when an input is received. Using this approach, the user must rely on the phone to accurately estimate the gaze position, but the vulnerability of a malicious user observing the password is all but eliminated. This would require remote estimation of gaze point on the screens. Furthermore, this method capitalizes on the advantages of combining knowledge and biometric factors and mitigates many of the disadvantages of using either knowledge or inherence factors exclusively.

In providing a competitive replacement method of user authentication to a mobile device, a crucial design consideration is optimized implementation with respect to authentication accuracy, battery consumption, and duration for existing mobile platforms. All of the algorithms must be performed by a mobile optimized processor, limit unnecessary battery use, and operate using the integrated camera. Additionally, the flexibility of mobile devices allows a user to be in any environment. Ideally, the user would always be in the exact same environment with the same lighting conditions as those used for training the algorithm, however, this is not a reasonable assumption, so the detection and tracking algorithms must also address real-time issues such as non-static devices, inadequate lighting, and background image noise presented by ultrabooks, tablets, and smartphones.

3.3 Gaze Estimation Algorithm

Real-time video images from the device's integrated camera are to be processed for the extraction of images of the eyes, which are passed to the gaze estimation and recognition phases. Figure 3.1 describes the flow of the algorithm. The initial hurdle to establishing ocular movements as a viable method for users to interface with their

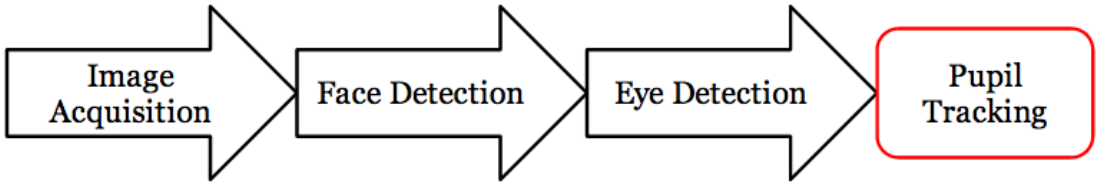


Figure 3.1: A flowchart of the application's flow that allows the mobile device to authenticate the user's identity using the multi-factor method described.

mobile devices centers on reliable detection of not only the ocular region, but the finer details of the region as well. Existing methods of gaze estimation rely on high resolution images and an infrared light source, but this work aims to use the existing cameras integrated into mobile devices at the time of this writing, namely the user-facing cameras found in mobile devices. As these cameras are designed for transmitting video for video chat applications, the design emphasis of these cameras is the capture of low resolution images with a large field of view.

Haar cascades have been used in training and are used for detecting the user's face and eyes. In over 1000 runs during the development of this work, these cascades allows the face and eyes to be detected rapidly and reliably. The detection time using the cascades is directly proportional to the number of pixels in, or size of, the image, so reducing the image size that is passed to the feature detection algorithm greatly reduces the detection time. Optimizations are made so that the size of the image is reduced whenever possible before it is passed onto the subsequent processing stages. Vertical face alignment is assumed throughout the authentication process to standardize the feature detection. Figure 3.2 illustrates an image of a user with the face and an eye image detected using the respective Haar cascade files.

3.3.1 Facial Feature Detection

First, a scaled image (640x480p) along with the face cascade is passed to the feature detection algorithm to detect the face. The feature detection algorithm returns the rectangles containing any areas in the image identified by the cascaded face filter. The

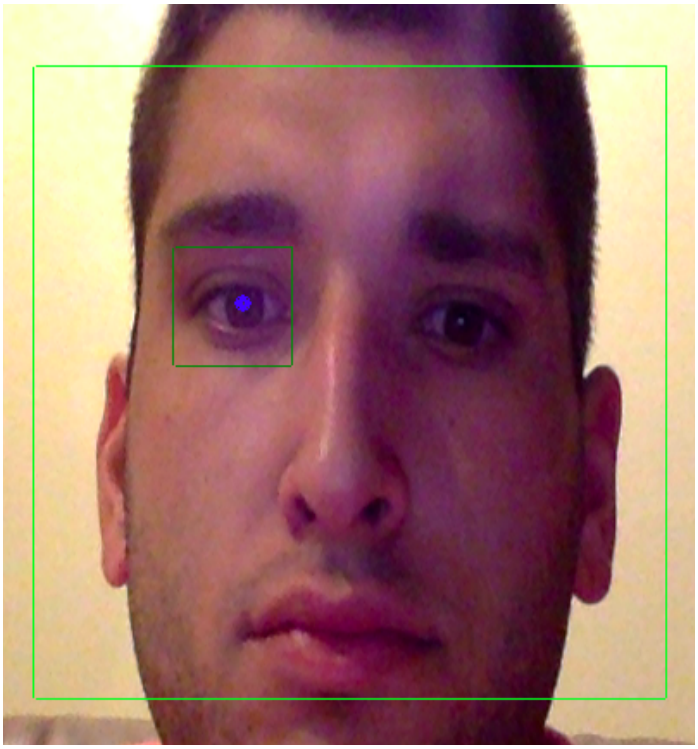


Figure 3.2: An example of the initial result of the gaze estimation algorithm, showing an image with the face and eye regions detected and marked by the green boxes, and the center of the pupil calculated and indicated by the blue dot.

best match is selected according to appropriate size and matching confidence. The user's face is the first feature to be detected by the algorithm. If there are multiple faces in the image, the largest face in the image will be selected, with the assumption that the device will be decisively closer to the user who is trying to authenticate and would have the largest face in the images. The face detection portion of the algorithm returns the four points that form the corners of a rectangle which marks the face region. This face image can still be relatively large (400x300p), so, in order to speed up the eye detection, a mask is placed on the face image. The mask is created by sub-sampling the face image by half vertically and horizontally. Then this half-sized image is passed to the eye detection step of the algorithm. Optimizing for the human anatomy, the eyes are found above the horizontal mid-line of the face, and one eye is located on either side of the vertical mid-line of the face. The same

matching algorithm that yields the face image is again used to apply the eye cascade to the input image.

Similar to the face detection algorithm, a special eye Haar cascade is used to detect the specific eye region in the image. The top left corner of this region is particularly important, because it is used in the gaze estimation portion of the algorithm as well. While the algorithm detects an eye within the subdivided face region, the algorithm continues using the same face region. This step also greatly reduces time between image grabs, allowing the method to execute at standard video frame rates (20-30 fps). This region of the image is cropped and passed to the pupil tracking portion of the algorithm.

Before the pupil can be tracked, it must first be segmented from the image. In image processing, segmentation refers to the separation or identification of all pixels corresponding to a specific object, in this case, the pupil. This will be accomplished through rudimentary image processing operations, in the hopes of keeping the computation time as low as possible. Many extraction algorithms were explored during the initial stages of this work to establish the optimal segmentation method.

3.4 Summary

The user authentication of this system achieves multi-factor authentication by challenging two identifying factors, knowledge and inherence. The primary obstacles facing the implementation of either function are mitigated through the complementary arrangement of the algorithm's flow. The knowledge factor allows the user to maintain the security of a password, and the biometric factor reduces the possible attacks that plague password systems.

Although the system operates and functions as a one-step system, several algorithms operate simultaneously to carry out the two factor procedure. The algorithm should be trained or calibrated to only acknowledge the user's eyes, and

in this way, only the user's inputs will be received. This provides an extra level of security not present in current MFA approaches. This extra security, implemented in a fashion appealing to users, will be essential to fulfill the principal goal of this work– replace the password. The following chapter describes the work undertaken to implement one-step multi-factor authentication on a mobile device and its results.

Chapter 4

Development and Implementation

A comprehensive explanation of the application development and algorithm implementation is presented in this chapter.

4.1 Android Development

The Android operating system was chosen as a starting place to begin developing gaze-based multi-factor authentication mobile devices. The Android Software Development Kit (SDK) uses the Eclipse development environment with the Android Developer Tools plug-in installed. OpenCV, a library of programming functions written in C/C++ aimed at real-time computer vision, has been ported to the Android platform and implemented by the OpenCV4Android library. OpenCV4Android is released under a BSD license and gives Android applications access to the OpenCV API by linking to the C library at runtime.

4.1.1 OpenCV4Android Application

Initial work was targeted at demonstrating feasibility of accurate detection of a face, eyes, and eye details of a user with the mobile device within an arm's length. The OpenCV4Android library supports Haar cascade feature detection, and, as such, lends

itself well to the purposes of this work. The application developed for the Android platform used the OpenCV Feature Detection Library to detect features in images based on a Haar cascade file. The images are acquired through the smartphone's forward facing camera. This integrated camera usually is designed with video chat applications in mind, and has a lower resolution imager better suited for real-time processing.

The application is straightforward and is designed to evaluate the feasibility of a smartphones hardware to implement a real-time feature detection application. The application triggers the smartphone's camera to capture an image, and the image is passed to two feature detection steps. Using the method outlined in the previous section, the first step calls a feature detection function that uses the face Haar cascade and returns an array of rectangles that contain facial components. The largest face rectangle is chosen, and the image is cropped to the rectangle of that face. This cropped and subsampled image is then passed to the eye detection function.

Along with the face subimage, an eye Haar cascade file is passed to the function. As before, the function returns an array of rectangular regions that correspond to rectangles that bound the components of the eye. After the best eye region has been returned, all of the rectangles are drawn on the screen, a new image is captured, and the detection process is repeated.

4.1.2 Android Application Results

Figure 4.1 shows the rectangles that are detected from the application and displayed on the smartphone's screen. With the favorable conditions of a static person, static device, and lighting conditions providing ideal contrast for feature detection and gaze processing, the detection algorithm yields relatively accurate results but lacks sensitivity to minor changes in lighting and mobile device position with respect to the user. Unfortunately, the Android API must support an abstraction from the camera software and does not have a highly developed functional interaction capability for the

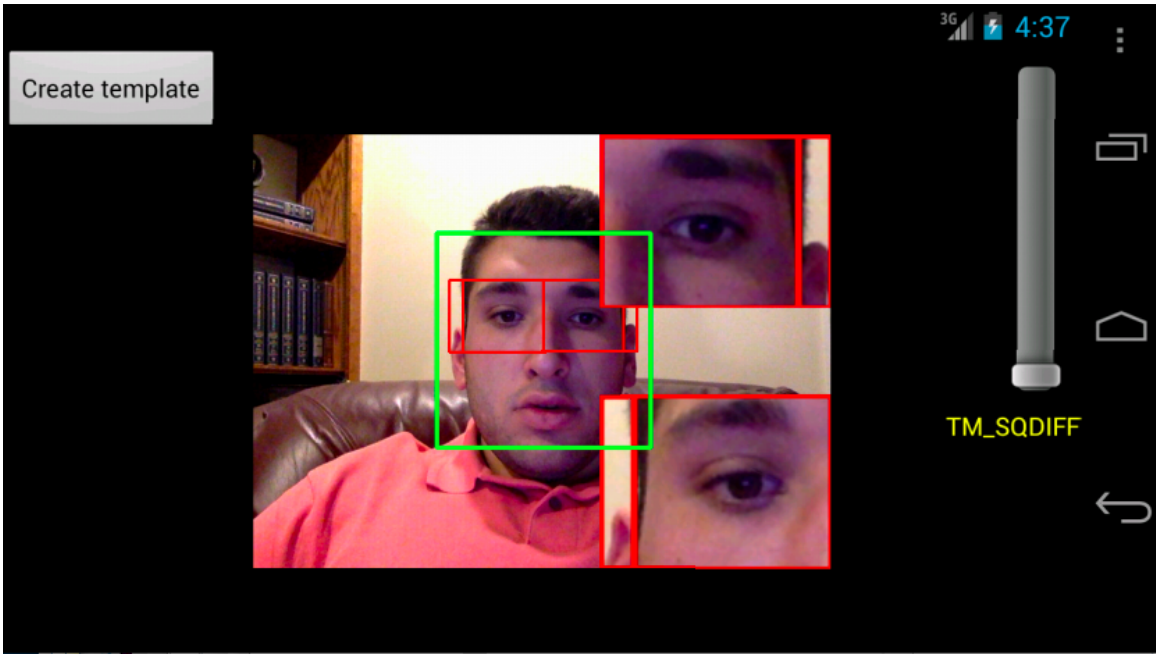


Figure 4.1: A screenshot of the Android application, where the face is outlined in green and the eyes are outlined in red.

camera. Furthermore, simply detecting the face features represents a small portion of the overall design. Reliable and real-time eye processing is needed to know exactly where the eye is looking.

By the time any processing could be done on a detected eye region, the image resolution was so low that no information could be extracted, and there was no way to modify and test in an efficient manner. Due to the demands of a mobile environment, the non-static nature of the device made development of an algorithm extremely difficult, and slow processor speeds made viewing rudimentary image processing in real-time infeasible. Given these challenges, but the overwhelming success of the feature detection step, it was determined that developing the segmentation and detail extraction of the eye area would be better achieved by moving away from the constraints of the smartphone. Since the face and eyes could be reliably tested, the next phase of the early research focused on extracting detailed eye information, namely the position of the subject's pupil, from low resolution eye

images, implemented using a laptop with a front-facing camera (Apple MacBook Air).

4.2 Eye Image Database

Moving away from the smartphone platform allows for streamlined evaluation of the available methods to process the eye region. The goal in processing the eye region is to yield enough detail to authenticate the user and estimate the user's gaze direction and sequence. The first step in the evaluation is to compile a database of images that represent a diverse user population. An image acquisition script, written in SimpleCV, was created to automatically save the images that are generated by the eye detection algorithms.

SimpleCV employs the functionality of the OpenCV libraries using Python wrappers to give developers a way to rapidly prototype image processing applications. Along with current image processing support, the SimpleCV libraries also have webcam support, which allows real-time applications to be developed on computers without much initial setup overhead. Unfortunately these features come at the cost of execution time, which increases proportionately to the resolution of the images being captured. However, video frame rates can still be achieved with optimized and resourceful coding.

Using scripts to automate the image acquisition process, a diverse eye database was established to allow processing techniques to be developed that would extract the pupil location from the eye of the diverse images. Since the SimpleCV library and OpenCV4Android library both link to the OpenCV library, this allows an ultrabook to capture images comparable to what can be achieved by the smartphone. The ultrabook used for this work is an Apple® MacBook Air, with a dual-core 1.7 GHz Intel® Core i7 processor and 8 GB of Random Access Memory.

The same feature extraction algorithm is employed from the Android application, but this time, the cropped images of the detected face and eyes are saved as files, so

that any language can interact with them. The eye image files were loaded into a database, since the organization of the images is important to determine the results of each segmentation method. Figure 4.2 shows an example of the image quality and resolution of the eyes contained in the database.

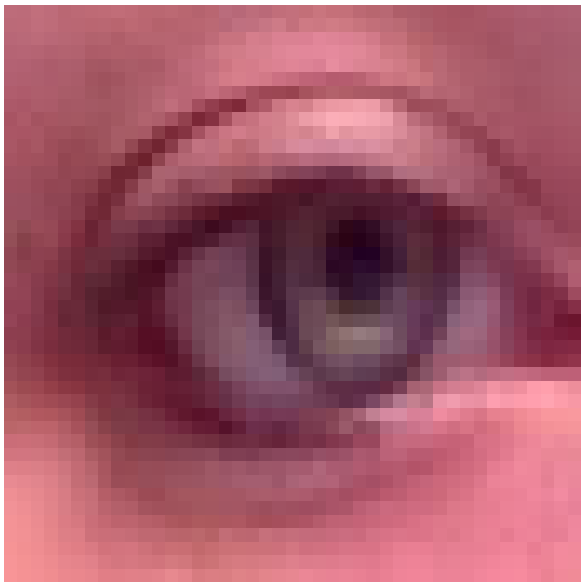


Figure 4.2: An example of an eye image saved to the database.

To ensure that the database represents a substantial number of eye presentations, over 325 eye images were collected from ten different subjects in five independent lighting conditions. The images are stored according to the subject and lighting fields in the database. No groundtruth information for the images is stored in the database. Although iris color is a relatively unique attribute, users were chosen based on distinctness of iris color. Lighting conditions were chosen based on type of lighting (incandescent, flourescent, sunlight, etc.) and lighting angle (overhead, ambient, structured, etc.). Organizing this database by iris color and lighting, several eye processing techniques could be developed and rapidly tested on images of eyes to identify challenging combinations of iris color and lighting.

4.3 Eye Image Processing

The iris of the eye is segmented from the eye image in order to find the location of the pupil. Given the high contrast edge between the sclera and the iris of the eye, an edge based approach was initially deemed the most favorable. The iris and pupil areas are assumed to be concentric circles. Matlab® was chosen as the development language, since the images from the database can be loaded by any Matlab® programming script.

For the initial implementation, images are processed at 960 by 1280 resolution. This gives the processing algorithms sufficient information to detect facial features and track the pupils, while not inhibiting the experience for the user. The resolution is an important consideration, because a subject's eyes will likely represent a small portion of the pixels in each image, so the highest resolution that can be supported without reducing the frame rate is used.

In order to maximize the frame rate it is important to find the algorithm that presents the greatest potential to accurately and quickly calculate the center of the iris within the eye image. For this work, three methods were evaluated to determine their fitness for pupil segmentation using the sample images in the eye image database: k-Means Clustering, Daugman's Integrodifferential Operator, and Morphological Processing.

4.3.1 k-Means Clustering

Clustering techniques are commonly used in image processing and computer vision applications to group pixels in an image based on similar features, usually color or intensity. In k-means clustering, k optimal clusters result, and the pixels of an image are classified to a cluster with respect to the minimum distance in color between each pixel and the average color of the closest, most similar cluster. This method was chosen because of the perceived distinctness in color of the different components of an eye image– skin, iris/pupil, sclera/whites.

The purpose of the k-means color-based segmentation method is the extraction of the colored iris region, containing both the iris and the pupil from the eye image. Before applying k-means, the colorspace of the image is transformed, allowing a stronger and more perceptual representation of the color content in the image. The eye image is first converted from the Red-Green-Blue (RGB) colorspace to the Lightness-Alpha-Beta (LAB) colorspace, where the alpha channel loosely corresponds to the red-green axis and the beta channel loosely corresponds to the blue-yellow axis. The alpha and beta channels are then clustered using k-means.

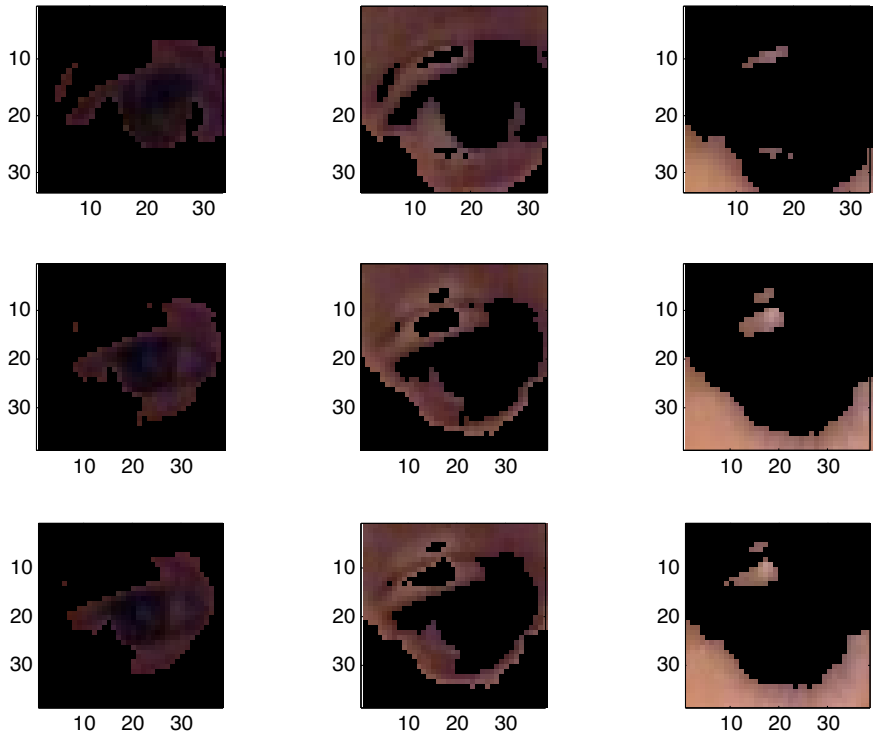


Figure 4.3: k-means applied to 3 images with $k = 3$

In this algorithm, the pixels of the eye image are grouped into k different components according to Euclidean distance between pixels, clustering the pixels that have the most similar color composition. This method operates under the assumption that three distinct color combination regions will be found (skin,

iris/pupil, sclera/whites). Due to this, the eye images were clustered using k equal to three, and the pixels of the iris and pupil are found in the cluster with the lowest magnitude. Acceptable results can be expected in specimens where the skin is a noticeably lighter hue than the iris and pupil, but there are certainly cases where the iris and pupil may be lighter than the skin, due to either lighting or biology. Figure 4.3 shows results of the method with $k = 3$ using representative images from the database. The performance, judged manually by observing the segmentation with the eye, was poor.

Given the anticipated color groups in the eye images, an intuitive assumption for $k = 3$ is that the skin, iris and pupil, and eye whites each have their own respective cluster, but that is not the case. The tone of the skin for the specimen has enough variation to require two clusters, and the whites of the eyes do not actually appear that white. After applying the k-means color-based segmentation method to several eye images with $k = 3, 4, 5$, acceptable results were never reliably achieved, and a more discriminating approach based on physiological assumptions was chosen.

4.3.2 Daugman's Integrodifferential Operator

Observing the physiology of the human eye, the edge of the iris can be seen as a circular area of dark iris pixels bounded by an area of lighter pixels of the whites creates an edge. The goal of Daugman's Integrodifferential Operator is to fit a circle to the boundary of the darker circular area of the iris, yielding a center and radius of the circle. After the boundary information is obtained, the iris can be easily segmented as all pixels inside the circular boundary with the associated center and radius.

Daugman's Integrodifferential Operator is an exhaustive search algorithm that finds the boundary between the iris and whites of the eyes. The operator searches over circles of all radii at each given center for the maximum average intensity gradient from across each concentric circle boundary to the next, along the radius and the

center of the circles. The operator is applied throughout the region of interest (ROI), and a Gaussian blur may be applied to smooth out any outlier noise that may cause erroneous results. The complexity of the algorithm is quite high since every pixel is observed R times, where R represents the number of radii to be processed, or once for every radius in the range between the minimum and the maximum radius. For every radius in the specified range, the normalized sum of the intensities of all circumferential pixel values is calculated for every pixel acting as a center. For every radius increase, the difference between the normalized sums of pixel intensity values of the adjacent circles is stored. After processing the entire range of radii, the center of the circle yielding the greatest edge is stated to be the center pixel of the iris, the boundary of which has the greatest change in circumferential pixels. Radman, Juwari, and Zainal present the algorithm in [36] implemented for this application. According to [36], the operator is governed by the following equation:

$$G_\sigma * max_{r,x_0,y_0} \left(\frac{\partial}{\partial r} \oint \frac{I(x,y)}{2\pi r} ds \right) \tag{4.1}$$

where, $I(x, y)$ is the intensity at coordinates x, y; r is the radius of the circular region with the center at x_0, y_0; σ, held constant at 2, is the standard deviation of the Gaussian distribution; s is the contour of the circle given by (r, x_0, y_0) governed by the equation of the circle.

Since every pixel in the image is a potential center candidate, preprocessing steps can help mitigate long processing times. In fact, several assumptions are valid, which can greatly reduce the candidate locations for the pupil center. It is assumed that the center of the pupil will be dark (intensity value less than 50). This means that the only pixels passed to the algorithm will be those above a specified threshold intensity value. Unfortunately, lighting conditions can create a glint reflection off the eye, creating the potential where the center of the eye may not be passed to the algorithm as a result of the center pixel being left out of the algorithm. For this reason, any glints caused by incident or directed lighting of the cornea are filled

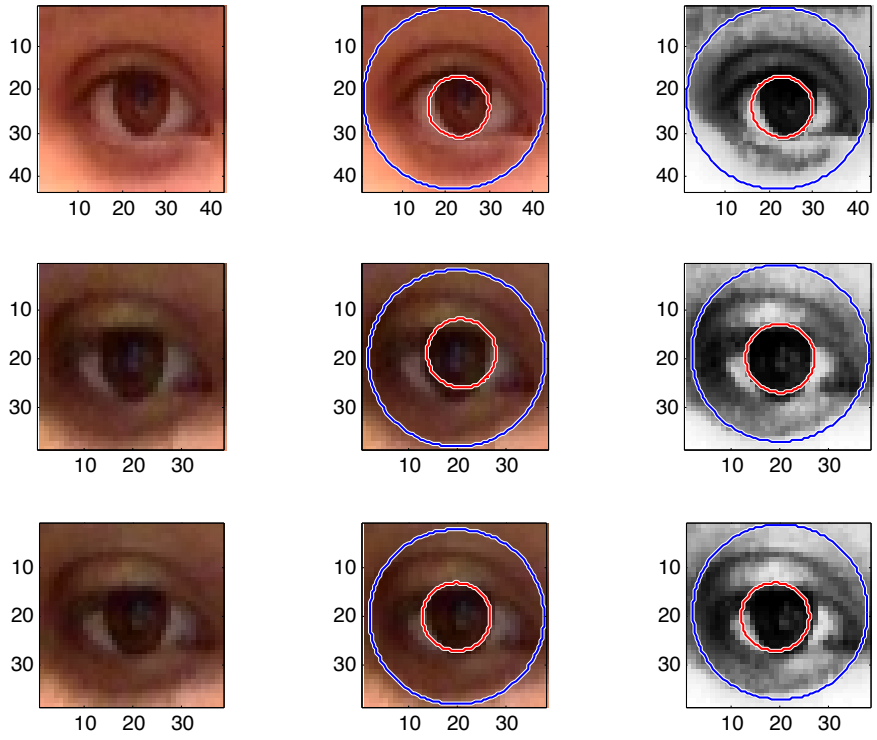


Figure 4.4: An example of Daugman's Integrodifferential Operator when it performs well. The first column is the original eye image, the second column is the result of Daugman's operator applied to the color image, and the third column is the fame algorithm applied to the monochrome image. The blue circle indicates an approximation of the corners of the eyes, and the red circle indicates the detected edge of the iris.

before any thresholding is applied. Additionally, some mathematical operations, such as division, can be avoided if the neighbors of the dark pixels are observed to ensure that only the darkest pixels in the neighborhood are passed to the algorithm. Finally, it is assumed that the pupil is reasonably centered within the image, such that the best circle fitting the iris will never go outside the bounds of the image.

As shown in Figure 4.4, the algorithm is able to detect the edge of the iris, as indicated by the red circle. The added dimensionality of the color images adds more information to the edges and allows the algorithm to detect the true iris edge

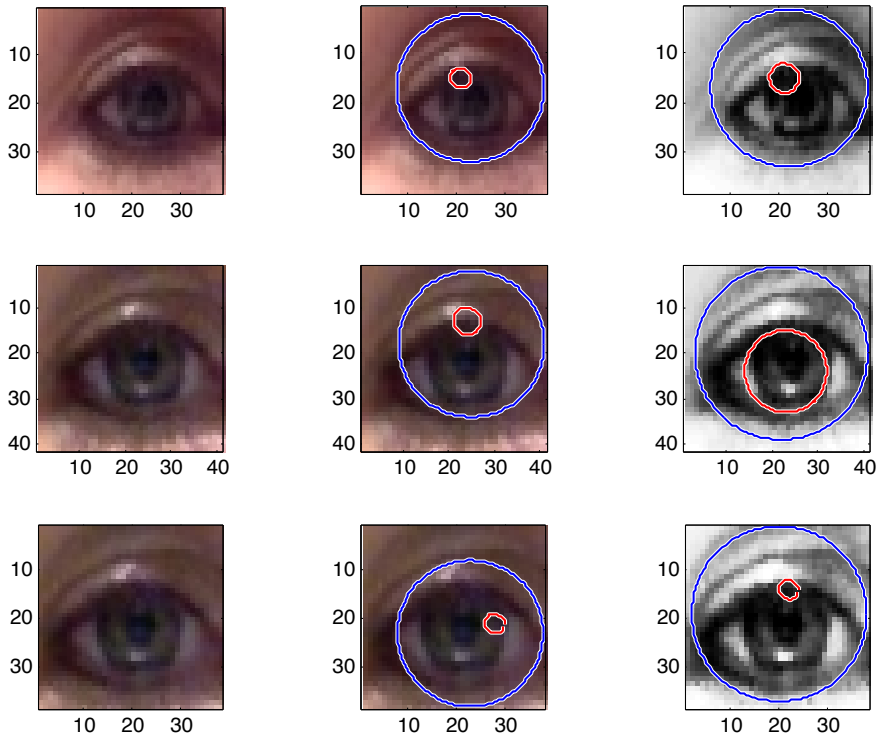


Figure 4.5: An example of Daugman's Integrodifferential Operator performing poorly. The color and shadows of the eyelashes create stronger edges than the color of the iris, which produces poor results. The images are organized just as Figure 4.4.

more accurately. However, this advantage illustrates the sensitivity of the algorithm's performance to color.

Figure 4.5 shows the erratic behavior of the algorithm when the iris is not in high contrast to the whites of the eyes. In spite of the strong analytical validity of Daugmans Integrodifferential Operator, this method does not achieve the appropriate results due to the averaging in the integration part of the algorithm. As is evident by the results shown, the intensity values of noise in the skin can create an average differential that mimics the average differential of an eye edge. Additionally, the eye images do not present favorable data to the algorithm. The computational load

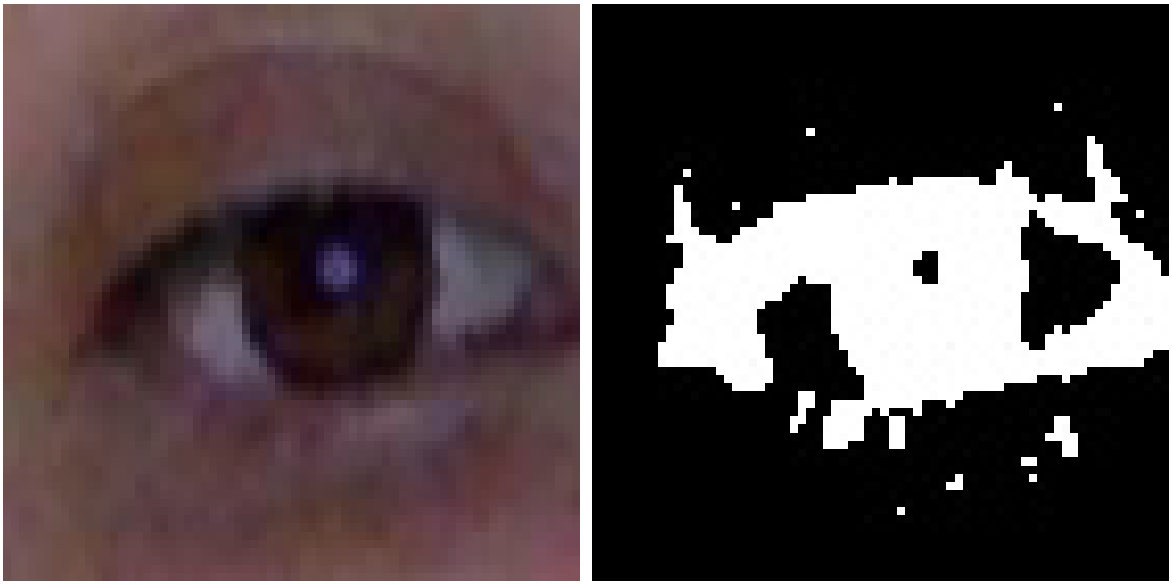
that the operation requires is not well-suited for low performance processing in real-time environments. Due to the real-time operational requirements of the solution and the low-power processor, image resolution and ambient lighting present very real challenges to the implementation of pupil detection in natural light settings. An overlooked aspect of the eye image is the eyelashes, and occasionally the eyebrow, that are sometimes included in an eye image. Daugman's operator does not properly handle partially occluded irises due to the eyelashes. Eyelashes can cause a difficult situation where the irises are no longer detected, as the eyebrows may be in higher contrast to the whites than the iris.

After testing on several other specimens with similarly light colored irises, the dependence on color contrast, along with the prohibitive computational times that come with exhaustive search algorithms, prove to be the limiting factors and exclude it from consideration as a suitable algorithm for real-time applications, such as that described in this work. Refocusing on algorithms that fulfill real-time constraints associated with image processing, points to a solution employing rudimentary methods that have been coded and optimized in the SimpleCV library.

4.3.3 Morphological Segmentation

Given the need for deterministic performance when extracting biometric information, a method with strong analytical integrity was initially sought out. After encountering obstacles with two deterministic approaches to the iris segmentation, developing a real-time segmentation approach became the main priority. Morphological segmentation uses nonlinear image filters, such as thresholding, dilation, and erosion. For this application, filters are selected that remove almost all information in the image except those pixels in the image representing the iris and pupil. Although this approach offers no theoretical guarantees regarding optimal segmentation, it successfully segments the iris area in real-time a high percentage of the time. This method is comprised of three simple processing techniques, implemented on every

image that is taken, usually accurately yielding the center of the user's pupil when performed in sequence. The techniques described in this section are implemented using the same SimpleCV library that provides the feature detection. This allows the techniques to be seamlessly incorporated into one cohesive application that carries out the entire iris segmentation process, from image capture to identifying the center of the iris, whereas the previous methods would require intensive porting efforts.

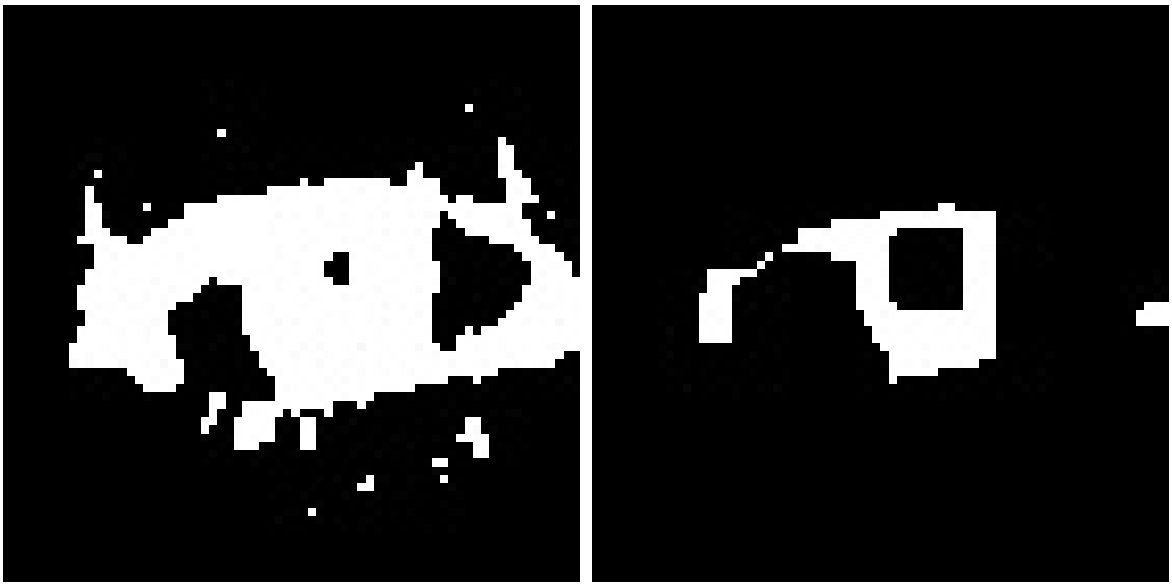


(a) Original **(b)** Threshold Image

Figure 4.6: Eye image and the results of thresholding with $threshold = 60$

The first step in segmenting the iris area is reducing the eye image to a binary representation using an adaptive threshold. Since the pupil should be the darkest region in the image, this binary representation separates the image into two categories: (1) pixels of intensity above the threshold and (2) pixels with intensity below the threshold. The threshold must be calibrated by the user from observed lighting conditions in the given setting to provide accurate results. Future may be undertaken to develop a method for automated threshold selection. In the binary representation, the pixels that are below the threshold are classified with value 1, with all other pixels

being ignored and classified with value 0. The output of the thresholding is shown in Figure 4.6.

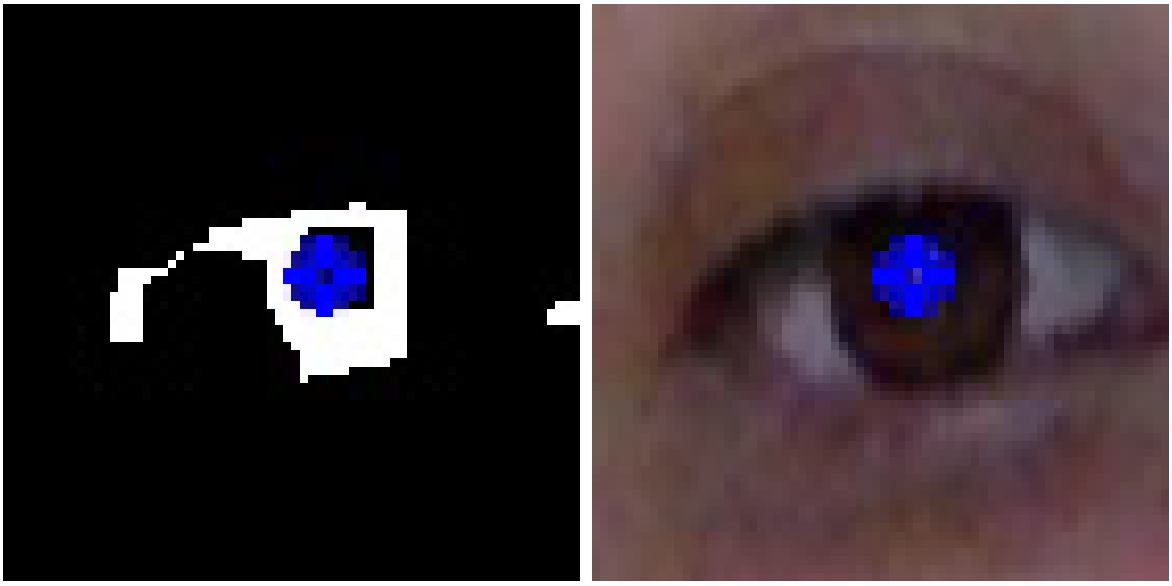


(a) Threshold Image **(b)** Morphological Image

Figure 4.7: Resulting image after all morphological processing with Threshold Image for comparison

After the thresholding, the binary image contains several binary regions comprised of the dark pixels from the image, including several noise artifacts that must be removed before the center of the iris can be calculated. To remove the remaining noise regions, a morphological erosion filter is applied to the image, removing sporadic noise elements of the skin and glares or glints in the eyes. The erosion operator removes pixels or regions of the binary image that do not have sufficient area to be the iris. The erosion operator is applied with a 3x3 mask, dictating that the minimum area of the iris region must be greater than nine pixels. All binary regions with less than seven neighbors are eliminated from the image. Since the edge pixels of the iris satisfy the elimination criteria for the erosion operator, those pixels must be restored after the erosion by applying a dilation filter to grow the areas. Dilation reconstitutes the regions of the image that still remain, and attempts to grow connected regions of the image. The results following both morphological processing steps are illustrated in

Figure 4.7(b). After the noise has been filtered out and the legitimately dark regions of the image are restored using morphological processing, the largest remaining region in the binary image should correspond to the iris region in the original color image.



(a) Binary Image **(b)** Original Image

Figure 4.8: Results of the blob detection and centroid calculation, with center position indicated by blue target. Point is also displayed on the original color image, displaying notably accurate performance.

The final stage in segmenting the iris area from the eye image is calculating the center of the iris area to be used in estimating the user's gaze. The SimpleCV blob detection operator is used to calculate the center of the largest connected component in the image. The blob detection method returns a list of regions in descending order of area, so the first region, i.e. the region of largest area, is chosen. The method also provides the centroids of all of the regions. Figure 4.8 depicts the result of the eye image processing, and indicates the center of the iris area with a blue target.

Using rudimentary image processing, real-time segmentation of the pupil can be achieved. While the method suffers from sensitivity to light and requires tuning, the performance of this simple algorithm is notably superior to the previous, more

complex methods. The next development stage centered on the creation of the application visible to the user during authentication.

Table 4.1: Summary of Performance of Iris Segmentation Algorithms

Variable	***k*-Means**	**DIDO**	**Morphological**
Dark Iris	Good	Best	Best
Light Iris	Poor	Poor	Good
Dark Skin	Poor	Good	Good
Light Skin	Good	Good	Best
Overhead	Good	Poor	Best
Sunlight	Poor	Good	Good
Lamp	Poor	Poor	Good
Bright Directed	Poor	Good	Good
Dim Light	Poor	Poor	Good

Table 4.1 summarizes the performance of the three implemented algorithms in the presence of varying conditions. The variables concern user features and lighting conditions of the eye subimages in the database. All performance measurements are judged manually by the eye ten to fifteen samples, and are indicated by the Poor, Good, and Best states. Poor performance indicates less than three successful segmentation attempts averaged over ten attempts. Good performance is achieved with more than five successful segmentation attempts, while Best performance status is noted when more than eight of ten attempts are successful on average.

Comparing the performance of the iris segmentation algorithms, the results show that the morphological processing approach performed most favorably. The thresholding step allows the method to adjust to varying lighting conditions. Even so, lamp and dim lighting are the harshest conditions for all of the methods. These lighting conditions do not provide the necessary illumination to allow confident segmentation of the eye images. Interestingly, overhead lighting casts a shadow on the user's eyes and causes a loss of contrast, reducing the accuracy of Daugman's algorithm. Light irises also pose a harsh challenge for the methods to deal with, due to the lack of contrast between the iris and the whites of the eyes. To mitigate this effect, the morphological processing approach is still able to segment the pupil area

as this will usually be a dark area, with the exception of bright directed lighting. An intense glint may be reflected in the presence of bright directed lighting, causing the pupil region to have high intensity values instead of low intensity values. The perfect user environment would be a user with light skin and dark irises in an overhead lighting condition. This situation consistently provides the best results when applying morphological processing to the eye image database and during real-time operation.

4.4 Application for User-Device Authentication

This section describes the application that has been developed to carry out the novel authentication method using multi-factor eye gaze. The application performs its tasks in three phases. The first phase implements the calibration needed for the application to deliver accurate performance. The second phase requires the user to establish a PIN of user selected length for use in all subsequent authentication attempts. The third phase of the application allows users to securely enter their PINs using multi-factor eye gaze.

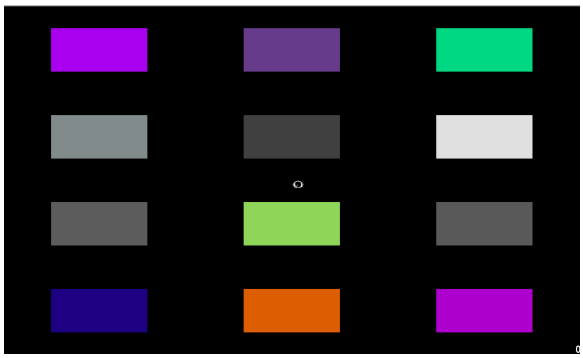


Figure 4.9: A screenshot of the authentication application.

Figure 4.9 shows the application's user interface, which was developed according to the design and requirements detailed in the previous chapter. To enter the password, the user must interact using eye gaze. Twelve colored blocks are presented to the user, arranged in four rows and three columns. Other arrangements can be used and are equivalent. Each block plays the role of a symbol, corresponding to its position,

in a PIN. As the user's gaze settles on a chosen block for a sufficient amount of time (approximately 400ms), an input to the device is triggered. The colors of all of the blocks randomly change when an input is recognized, indicating to the user that an input has been received and the device is ready for the next input.

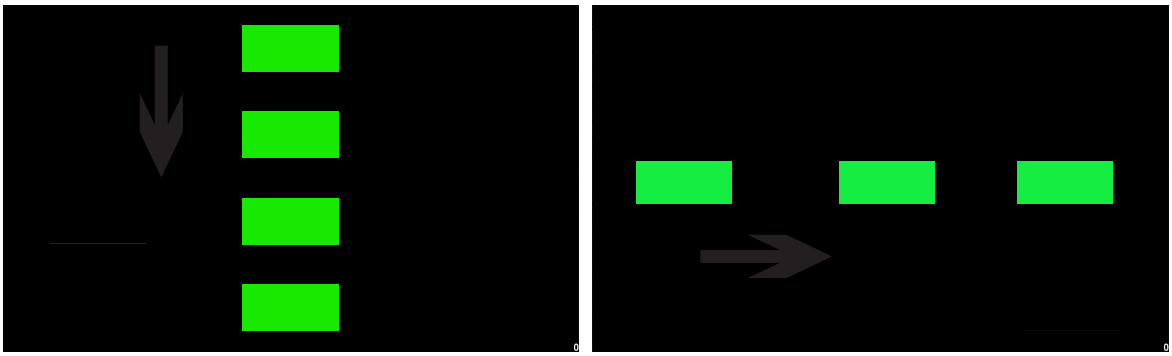
The application was developed using PyGame, the Python gaming library. It supports basic interface capabilities that were integrated into the existing pupil segmentation application. The pupil segmentation application outputs the center of the pupil, the user interface provides feedback to the user, and the gaze estimation framework translates the pupil center into a gaze point on the screen for the user interface.

4.4.1 Gaze Estimation via Measurement and Projection

Before implementation, the center of the eye region is first established to serve as the reference point of a neutral gaze. This point will be utilized during an authentication attempt to assess the direction of the user's gaze. If the subject maintains a neutral gaze by looking at the middle of the mobile device, the center of the pupil and the center of the eye region should be roughly the same. To estimate the point in space on the screen where the subject is gazing, the two-dimensional difference (Δx, Δy) between the reference center of the eye region and the pupil center is used. The output of the pupil segmentation step is the center of the extracted pupil segment. The center of the eye region can be calculated as the center of the eye subimage. It is important to remember that these points are not identical. For this reason, the center of the eye region is used as the reference point of a neutral gaze, to be taken and stored into the memory of the device.

For most subjects, the two centers will not align perfectly, so a translation constant must be calculated to offset the reference point or eye region center. If the subject looks away the eye region center and the pupil center are no longer at the same position, and the distance between these points is measured. The distance

measurement is provided in its horizontal, Δx, and vertical, Δy components as a two-dimensional vector called the gaze vector, (Δx, Δy).



(a) Vertical Calibration **(b)** Horizontal Calibration

Figure 4.10: Illustration of the vertical and horizontal calibration phases

Through calibration steps, centroids for each block on the interface are then established to classify measured differences and represent estimated gaze point as screen coordinates. As opposed to requiring a calibration step for each block independently, resulting in twelve steps, vertical and horizontal centroids are calibrated by two independent calibration steps. The first step establishes the four vertical centroid points by prompting the user to gaze at each of the four central blocks along the vertical axis and averaging the vertical components of the gaze vectors across multiple samples, as shown in Figure 4.10(a). The second step calibrates the three centroids across the horizontal axis, as shown in Figure 4.10(b), using the same method. This results in seven calibration stages rather than twelve, reducing calibration time. After calibration is complete, every gaze vector that is sampled from a new image is classified according to the closest centroid in each direction. Classification of the gaze vector in this manner achieves the gaze estimation, and enables the application to perform the biometric portion of the authentication, but the knowledge factor remains to be established. After the calibration of the gaze estimation is complete, the application enters the next phase.

4.4.2 Establishing the Password

During the second phase of the application, the user establishes the length and value of the PIN to be used in all subsequent authentication attempts. This setup phase is required once. The user first chooses the length of PIN to create, with longer PIN selections providing more security and longer input times. The user selects among the range of lengths from four to seven symbols (more symbols should be used in a fielded system. After the length is chosen, the user creates the PIN that will be used for authentication. Creating the PIN is achieved through eye gaze to acclimate the user to the new interface.

Once the PIN is established, it is important that the PIN be stored into the device's secure and encrypted place in memory to protect it from any malicious memory attacks. This allows the PIN to be used securely during the third application phase until the user decides to manually recreate the PIN.

4.4.3 Entering the Password

While the application is in this phase, multi-factor authentication using eye gaze may be performed seamlessly. The user, when prompted with the authentication screen, gazes at the necessary positions of the blocks in sequence that represent the proper value to enter in the correct PIN established during the second phase. If the biometric features of the user's eyes do not correspond to the calibration established during the first phase, the application will not be able to authenticate successfully. Similarly, if the application recognizes the user's input, but the entered PIN is not the same as the one stored in the encrypted memory, authentication will fail. Only when both the biometric and the knowledge criteria are met will the user be able to successfully authenticate.

Through testing of the application, users other than the user who performed the calibration steps were rejected, and false positives were never encountered. Unfortunately, genuine users attempting to authenticate experienced false negatives.

This indicates a high sensitivity of the biometric recognition portion of the application to something other than the user and the password and has been identified as future work.

Since the gaze estimation of the application is based on the morphological segmentation algorithm, the performance of the application is subject to the same limitations as the morphological segmentation algorithm, namely the lighting conditions. As a result, authentication attempts using the application must be performed under lighting environments similar to the calibration environment. As the application is improved through further development, automatic compensation for lighting conditions will increase the performance of the algorithm. Once the dependence on lighting is addressed, the performance of the application will offer the first viable alternative to the traditional password.

4.5 Summary

The pupil segmentation and user interface components were successfully integrated, and users are able to set a password and authenticate with minimal feedback using gaze as the interface medium. This chapter outlines the development of a novel authentication method using multi-factor eye gaze to compete with an obsolete password model that has outlasted its capabilities. Gaze estimation under natural lighting conditions using an embedded webcam from a mobile device is a difficult problem. Despite this challenge, the developed application successfully demonstrates an initial solution comprised of a pupil segmentation algorithm and gaze estimation that allows a user to authenticate oneself using gaze as the interaction medium. As a result of the work performed in this chapter, several conclusions have been reached and topics have been identified for future work, as discussed in the next chapter.

Chapter 5

Conclusions and Future Work

The feasibility of using a user's eye gaze to authenticate and gain access to a mobile device was explored. An image of the device's user is taken with the front-facing webcam on a mobile device. From that image, the face is first detected and isolated, followed by detection and localization of the eye region. Next, the center of the iris is calculated through a morphological segmentation algorithm. Then, the user gazes at a series of predetermined locations on the mobile device's screen that are indicated with randomly colored blocks. The authentication application running on the mobile device detects when the user has settled on a particular block and triggers an input with the value corresponding to that block's location on a standard digital keypad. Once an input is received by the application, the blocks' colors are randomly chosen to notify the user that an input has been received. The performance of various natural lighting conditions was explored. Results showed that the resolution of the mobile phone's camera, as well as the lack of computing power of the native mobile phone device was not high enough to yield a consistent user experience. Natural lighting conditions were also found to be limiting to the efficient segmentation of the iris region running on a smartphone. The gaze-based authentication algorithm was then implemented on an ultrabook in the hopes that a device with higher resolution and increased computing power would overcome the inadequacy of the smartphones.

Implementation results on the ultrabook indicate sufficient potential for this method to compete with the traditional password model both in security and usability.

An authentication application was developed to provide an experience that leveraged the mobile environment to provide a competitive alternative to the common password model. The application provides obscure feedback that is not readily interpreted by a malicious user. This novel method allows users to authenticate to mobile devices through gaze-based interactions facilitated by a straightforward and mobile-friendly algorithm.

Using the Android environment, the feasibility of the overarching design has been verified, and real-time detection of facial features can be achieved using Haar cascade detection. The center of the eye region is calculated, and image processing techniques extract the center of the pupil. Using the difference between these measurements, the gaze point may be estimated. The approach has large potential for further optimizations regarding ambient conditions.

Before the application may be commercially released, automatic correction methods are needed for lighting, shadows, head pose, device movement, and eye colors. All of these factors have been shown to cause variations in the gaze estimation performance. Once these issues are addressed for natural lighting environments, gaze estimation may be applied for many other applications in mobile devices. Additionally, new applications will certainly arise from the growing wearable devices market, a new frontier of mobile devices.

Bibliography

[1] Adams, A. and Sasse, M. A. (1999). Users are not the enemy. *Commun. ACM*, page 4046. 17

[2] Almuairfi, S., Veeraraghavan, P., and Chilamkurti, N. (2011). IPAS: implicit password authentication system. In *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications (WAINA)*, pages 430–435. 5, 7, 59

[3] Bednarik, R., Kinnunen, T., Mihaila, A., and Frnti, P. (2005). Eye-movements as a biometric. In Kalviainen, H., Parkkinen, J., and Kaarna, A., editors, *Image Analysis*, number 3540 in Lecture Notes in Computer Science, pages 780–789. Springer Berlin Heidelberg. 14

[4] Bonneau, J., Herley, C., van Oorschot, P., and Stajano, F. (2012). The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *2012 IEEE Symposium on Security and Privacy (SP)*, pages 553–567. 2

[5] Corcoran, P., Nanu, F., Petrescu, S., and Bigioi, P. (2012). Real-time eye gaze tracking for gaming design and consumer electronics systems. *IEEE Transactions on Consumer Electronics*, pages 347–355. 12

[6] De Luca, A., Weiss, R., and Drewes, H. (2007). Evaluation of eye-gaze interaction methods for security enhanced PIN-entry. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*, page 199202, New York, NY, USA. ACM. 14

[7] DeFigueiredo, D. (2011). The case for mobile two-factor authentication. *IEEE Security Privacy*, pages 81–85. 5

[8] Ephraim, T., Himmelman, T., and Siddiqi, K. (2009). Real-time viola-jones face detection in a web browser. In *Canadian Conference on Computer and Robot Vision, 2009. CRV '09*, pages 321–328. 12

[9] Fan, C.-I. and Lin, Y.-H. (2009). Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics. *IEEE Transactions on Information Forensics and Security*, pages 933–945. 10

[10] Fini, M., Kashani, M., and Rahmati, M. (2011). Eye detection and tracking in image with complex background. In *2011 3rd International Conference on Electronics Computer Technology (ICECT)*, pages 57–61.

[11] Hansen, D. and Ji, Q. (2010). In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 478–500.

[12] Hennessey, C. and Lawrence, P. (2009a). Improving the accuracy and reliability of remote system-calibration-free eye-gaze tracking. *IEEE Transactions on Biomedical Engineering*, pages 1891–1900. 13

[13] Hennessey, C. and Lawrence, P. (2009b). Noncontact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions. *IEEE Transactions on Biomedical Engineering*, pages 790–799. 13

[14] Hennessey, C., Noureddin, B., and Lawrence, P. (2008). Fixation precision in high-speed noncontact eye-gaze tracking. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, pages 289–298. 13

[15] Herley, C. and Van Oorschot, P. (2012). A research agenda acknowledging the persistence of passwords. *IEEE Security Privacy*, pages 28–36. 2

[16] Huang, S.-H. and Lai, S.-H. (2004). Real-time face detection in color video. In *Multimedia Modelling Conference, 2004. Proceedings. 10th International*, pages 338–345.

[17] Huang, X., Xiang, Y., Chonka, A., Zhou, J., and Deng, R.-H. (2011). A generic framework for three-factor authentication: Preserving security and privacy in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, pages 1390–1397. 10

[18] Jiang, N., Lu, Y., Tang, S., and Goto, S. (2010). Rapid face detection using a multi-mode cascade and separate haar feature. In *2010 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 1–4. 13

[19] Jiang, N., Yu, W., Tang, S., and Goto, S. (2011). A cascade detector for rapid face detection. In *2011 IEEE 7th International Colloquium on Signal Processing and its Applications (CSPA)*, pages 155–158.

[20] Kashani, M., Arani, M., and Fini, M. (2011). Eye detection and tracking in images with using bag of pixels. In *2011 IEEE 3rd International Conference on Communication Software and Networks (ICCSN)*, pages 64–68.

[21] Kasprowski, P. and Ober, J. (2004). Eye movements in biometrics. In Maltoni, D. and Jain, A. K., editors, *Biometric Authentication*, Lecture Notes in Computer Science, pages 248–258. Springer Berlin Heidelberg. 14

[22] Kumar, M., Garfinkel, T., Boneh, D., and Winograd, T. (2007). Reducing shoulder-surfing by using gaze-based password entry. In *Proceedings of the 3rd symposium on Usable privacy and security*, page 1319, New York, NY, USA. ACM. 14

[23] Liang, Z., Tan, F., and Chi, Z. (2012). Video-based biometric identification using eye tracking technique. In *2012 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC)*, pages 728–733. 14

[24] Liou, J.-C., Egan, G., Patel, J., and Bhashyam, S. (2011). A sophisticated RFID application on multi-factor authentication. In *2011 Eighth International Conference on Information Technology: New Generations (ITNG)*, pages 180–185. 7, 9

[25] Maeder, A. J. and Fookes, C. B. (2003). A visual attention approach to personal identification. In *Faculty of Built Environment and Engineering; School of Engineering Systems*, pages 1–7. 14

[26] Majumder, A., Behera, L., and Subramanian, V. (2011). Automatic and robust detection of facial features in frontal face images. In *2011 UkSim 13th International Conference on Computer Modelling and Simulation (UKSim)*, pages 331–336.

[27] Mao, Z., Florencio, D., and Herley, C. (2011). Painless migration from passwords to two factor authentication. In *2011 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. x, 2, 7, 8

[28] Mehrubeoglu, M., Pham, L. M., Le, H. T., Muddu, R., and Ryu, D. (2011). Real-time eye tracking using a smart camera. In *2011 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–7.

[29] Mei, Z., Liu, J., Li, Z., and Yang, L. (2011). Study of the eye-tracking methods based on video. In *2011 Third International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, pages 1–5. 12

[30] Millan, M. S., Perez-Cabre, E., and Javidi, B. (2006). Multifactor authentication reinforces optical security. *Optics Letters*, pages 721–723. 10

[31] Miyazaki, S., Takano, H., and Nakamura, K. (2007). Suitable checkpoints of features surrounding the eye for eye tracking using template matching. In *SICE, 2007 Annual Conference*, pages 356–360.

[32] Morris, R. and Thompson, K. (1979). Password security: a case history. *Commun. ACM*, page 594597. 1

[33] Nanu, F., Petrescu, S., Corcoran, P., and Bigioi, P. (2011). Face and gaze tracking as input methods for gaming design. In *Games Innovation Conference (IGIC), 2011 IEEE International*, pages 115–116.

[34] O'Gorman, L. (2003). Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, pages 2021–2040. 1, 2

[35] Phiri, J., Zhao, T.-J., and Agbinya, J. (2011). Biometrics, device metrics and pseudo metrics in a multifactor authentication with artificial intelligence. In *2011 6th International Conference on Broadband and Biomedical Communications (IB2Com)*, pages 157–162. 10

[36] Radman, A., Jumari, K., and Zainal, N. (2013). Fast and reliable iris segmentation algorithm. *IET Image Processing*, 7(1):42–49. 32

[37] Skracic, K., Pale, P., and Jeren, B. (2013). Knowledge based authentication requirements. In *2013 36th International Convention on Information Communication Technology Electronics Microelectronics (MIPRO)*, pages 1116–1120. 5

[38] Sun, Q., Li, Z., Jiang, X., and Kot, A. (2008). An interactive and secure user authentication scheme for mobile devices. In *IEEE International Symposium on Circuits and Systems, 2008. ISCAS 2008*, pages 2973–2976. 10

[39] Tiwari, A., Sanyal, S., Abraham, A., Knapskog, S. J., and Sanyal, S. (2011). A multi-factor security protocol for wireless payment - secure web authentication using mobile devices. Technical report, India Institute of Information Technology. 9

[40] Udayashankar, A., Kowshik, A., Chandramouli, S., and Prashanth, H. S. (2012). Assistance for the paralyzed using eye blink detection. In *2012 Fourth International Conference on Digital Home (ICDH)*, pages 104–108. 12

[41] Uludag, U., Pankanti, S., Prabhakar, S., and Jain, A. (2004). Biometric cryptosystems: issues and challenges. *Proceedings of the IEEE*, pages 948–960. xi, 60, 61

[42] Viola, P. and Jones, M. (2001a). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001*, volume 1, pages I–511–I–518 vol.1. 11

[43] Viola, P. and Jones, M. (2001b). Robust real-time face detection. In *Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings*, volume 2, pages 747–747.

[44] Vipin, M., Sarad, A., and Sankar, K. (2008). A multi way tree for token based authentication. In *2008 International Conference on Computer Science and Software Engineering*, pages 1011–1014. 9

[45] Yan, B., Gao, L., and Zhang, X. (2009). Research on feature points positioning in non-contact eye-gaze tracking system. In *9th International Conference on Electronic Measurement Instruments, 2009. ICEMI '09*, pages 1042–1045. 13

[46] Yang, C., Sun, J., Liu, J., Yang, X., Wang, D., and Liu, W. (2010). A gray difference-based pre-processing for gaze tracking. In *2010 IEEE 10th International Conference on Signal Processing (ICSP)*, pages 1293–1296. 13

[47] Yuan, Z. and Kebin, J. (2011). A local and scale integrated feature descriptor in eye-gaze tracking. In *2011 4th International Congress on Image and Signal Processing (CISP)*, pages 465–468.

[48] Zhu, Z. and Ji, Q. (2007). Novel eye gaze tracking techniques under natural head movement. *IEEE Transactions on Biomedical Engineering*, pages 2246–2260. 13

Appendix

Appendix A

User Authentication

In network security, an authentication protocol is the process by which an entity proves their identity. Authentication protocols are usually part of a larger cryptographic system used to secure privileged information from unauthorized entities. The mark of great engineering is an invention or system that disappears from the consciousness of the user, the goal being to provide the most convenient design and solution. Unfortunately for computing technologies, this goal flies in the face of security, where designers of web services and mobile applications have opted for convenience, shying away from implementing bulletproof security protocols. As a result, these technologies have cryptographic systems that operate behind the scenes, hidden from the user with the exception of the authentication portion of the system. Intrinsically, the user authentication steps must be exposed to the requesting entity. Many times this takes the form of a challenge and answer format. Valid authentication challenges can be grouped into three categories: What you know, What you have, What you are.

A.1 Knowledge - What Do You Know?

Knowledge based schemes are associated with the knowledge of some secret, or password, that is verified to validate a user's identity. Knowledge factor

authentication is the most common type of authentication, since all password-centric schemes are based on knowledge-factor authentication. The authenticating system stores the secret and compares any future authentication attempts against this stored secret.

Generally a password is chosen by the user and communicated to the authentication system. For these user-created passwords, the authentication system typically places requirements on the length or content of the password to ensure that it will be sufficiently complex to avoid a brute force attack. Unfortunately an attacker can consider these requirements when developing a brute force attack, and the added complexity of these requirements becomes negligible.

Passwords can also be established mutually, with each entity contributing a portion of the secret and then both portions being combined to form the final secret to be used. A password can be given by the authenticating system, in which case the user is responsible for remembering it. A secret must not only be created, but it must also be known by both parties, and to be known it must be remembered. Remembering the secret is the main issue with this system.

A.1.1 Password Challenges

This scheme requires that both entities, the user and the authentication system, remember the secret. For a user this means either recording or memorizing the secret, and both pose obvious problems and require shortcuts to be taken by the user that jeopardize the potential security of the authentication scheme as a whole.

In order to avoid one password protecting all of their accounts, many users are aware that every account should have its own password. This results in users recording passwords, since unfortunately users cannot usually memorize that many random passwords. Physically or digitally recording their secrets typically takes the form of the user compiling a library of all passwords in a file or note. Most users justify this behavior by the assumption that the contents of a file on their computer or a note

on their desk would never be viewed by a malicious, untrusted user. What is not appreciated is that this behavior reduces all of their passwords down to a single point of failure, that, in most cases, is not even protected. In other words, choosing different passwords is done to prevent the compromise of one password from jeopardizing all accounts, but once the password library is compromised then all of the accounts can be accessed, and in some cases this library is not even secured and can be viewed with little effort.

On the other hand, the users who correctly refuse to record their passwords must choose passwords that are easily remembered, but these easily remembered passwords are usually taken from life contexts like addresses, phone numbers, names, or words, which lend themselves to dictionary attacks and social engineering. An educated attacker who knows these contexts has a high likelihood of guessing the password. These limitations only take into account the user side of the scheme, but the authentication system also add limitations that must be considered.

For the authentication system to know the information, in a computing application, means that the information must be stored in memory. Storing secure information in memory is comparable to writing a password down, so it automatically becomes a point of failure for any authentication system. Basically, the authentication system must record the secret in order to have knowledge of the secret, and this means that the secret must be hidden in order to be kept secret. The password must be encrypted or hashed before it is stored. Once stored, the password must remain hashed or encrypted as a security measure.

The specifics of every knowledge-based authentication vary, but the general model remains the same, where there are two entities and one requires the verification of the identity of the other. The dialog typically takes the form of a challenge and response, where the challenge is what is the username and password, and the response is the username and password. Some security questions may follow but those questions follow the same challenge and response dialog.

A.2 Possession- What Do You Have?

Possession-based authentication places emphasis on simple possession of a physical object or token. A key and lock is an example of a possession-based authentication scheme, where the lock is the authentication system that asks the question, Do you have the key?, and the key is the answer. Whoever possesses the key can successfully open the lock. This model does not translate well to virtual environments as possession is almost impossible to validate in a computing environment. The best approximation is to communicate via a specific device or provide information like a password that lies within that device, which indirectly proves possession of that device. However, solely relying on possession poses many issues.

A.2.1 Possession Challenges

Objects are easily lost or stolen, and lost objects may be found by a malicious user. Objects can also be duplicated, which is more difficult to detect as the duplicated object will be valid for as long as the genuine object is valid. Provisions must be made and precautions must be taken to prevent the use of stolen or duplicated objects.

Once objects are lost, stolen, or duplicated, the credentials of that object must be revoked. Upon revocation, a new object must be chosen or established. Unfortunately for some computing applications, revocation may mean that the entire account is irreparably compromised and must be replaced. This would be analogous to having to replace all the locks because the key was lost or stolen.

A.3 Inherence - What Are You?

This authentication scheme emphasizes the exhibition of a specific characteristic or property. Generally, inherence factors are synonymous with biometric factors, since biometrics are intrinsic human properties. More accurately, biometric methods are a subset of the set of inherence factors that can be used. However, biometric factors

are the most suitable factors considering the requirements of authentication as they provide the finest granularity and should be able to accurately discriminate between similar attributes of different users. These systems can be characterized as having a sensing and differentiating criteria.

Vulnerabilities and obstacles uniquely associated with biometric authentication are false-positives and false-negatives of the matching algorithm, replay attacks, irrevocable credentials, and extra equipment [2].

A.3.1 Real World Example

An illustrative example of an inherence factor authentication system can be found on most touch-sensitive interfaces using capacitance as the differentiating criteria. The touchscreen of a smartphone realizes this authentication method by requiring input from an object that inherently has the capacitance of a finger. If the ability of an object to register touches with the screen is considered a privilege, and human fingers are the only objects privileged to register touches, then every time an object touches the screen, the object must authenticate that it is a finger in order to prove that it is authorized to interact with the device. In authentication terms, in order for the screen to register a touch, the object touching the screen must authenticate that it has the capacitance of a finger.

This complex array of statements demonstrates how using inherence-based authentication exhibits the greatest potential to disappear into the use of a system. A user never has to think twice about whether to use a finger to touch the screen, and the authentication disappears into the use of the screen.

In this way, more complicated biometric-based authentication systems seek to confirm the inherent properties of an entity with minimal conscious effort. For instance, facial recognition validates that a person has the same face, or face metrics, as a previously authorized user without any memorization or ensuring they have a

token with them. For more sensitive implementations, the tolerance of what it takes to be considered becomes more accurate and less permissive.

A.3.2 Biometric Challenges

Returning to the smartphone screen example to demonstrate the tolerance vulnerability of biometric authentication, an object that is not a finger, but merely has the same capacitance as a finger, that touches the screen will still register a touch. Furthermore, this invalid object (not a finger) will always have the privilege of registering touches. Anything else that holds the same (or similar within some tolerance) inherence property, capacitance in this case, will be able to authenticate. This brings up the greatest weakness of using a biometric authentication factor.

Biometric authentication is irrevocable, implying that any person who can authenticate once, will forevermore be able to gain access. This presents obvious challenges to a real-time system where the differentiating criteria is not or cannot be made specific enough. The ideal differentiating criteria must be able to distinguish between the most subtle variations in the data. Further complicating matters, is the competing necessity to convert analog biometric data to digital values.

Obviously the exact representation of a biometric measurement cannot be fully digitized, thus introducing quantization of the data. Uludag, Pankanti, Prabhakar, and Jain offer the principle variations in the presentation of biometric information in [41], which are listed in Figure A.1.

Although this conversion from analog to digital is absolutely necessary, it not only strips valuable information from the data, it also maps multiple analog values to the same digital representation and diminishes variation. At the same time, there will always be noise when dealing with sensor data, so quantization will normalize out some of the variations caused by noise. The risk is that contained in the noise component, some transient data will be the only discriminating information to separate the biometrics of two users.

1. *Inconsistent*- Natural biometric signal is a non-deterministic composition of the physiological trait. Intrinsic variation exists in creating a deterministic representation.

2. *Irreproducible*- Environmental or permanent physiological change can render biometric signals irreproducible and useless.

3. *Imperfect Acquisition*- Given perfect presentation, the signal acquisition may still introduce variation due to hardware interactions with the data, e.g. camera automatically compensating for lighting, thus altering the signal.

Figure A.1: List of the variances found in biometric signal acquisition by Uludag, et al. in [41]

After acquisition of the biometric signal, some data processing is needed to correct variations before determining whether the acquired signal matches the stored representation of the signal. The signals should be aligned by matching keypoints in the signal. This should result in a rotation or shifting translation that will allow the data two distinct presentations requires complex pattern recognition and decision-making algorithms.

Vita

Lucas Herrera was born in the beautiful Knoxville, TN in 1990. In 2012, he received a Bachelor's Degree in Electrical Engineering from the University of Tennessee, Knoxville, where he began his graduate studies. He received his Master's Degree in Computer Engineering in May 2014. His work focuses on developing practical solutions and commercializing competitive technology.