

**Biometrics and Forensic Applications: Stress Testing a Multi-Modal Biometric Authentication System**

**James Finglas**

**B00094138**

Department of Informatics

School of Informatics and Engineering

Technological University Dublin

Submitted to Technological University Dublin in Partial Fulfilment of the requirements for the degree of:

Bachelor of Science (Honors) in Computing

Course: Application Security

Lecturer: Arnold

Submission Date: 12/12/2019

Word count (Introduction to Conclusions): 3125

# Project Submission/Declaration Form

Submitted by: James Finglas

Course Coordinator: Stephen O'Shaughnessy

Module Name: Biometrics and Forensics Applications

Course Code: H4016

Lecturer: Arnold Hensman

Submission Date: 12/12/2019

I declare that the work contained in this report which I now submit on the program of study leading to the award of Degree of Honors B.Sc. in Computer Science in Technological University for Dublin is my own work and has not been taken from the work of others. Any sources which have been referenced or analyzed have been cited using the Harvard/IEEE standard within the body of this report. It is entirely my own work except where otherwise stated and has not been submitted for assessment for an academic purpose at this or any other academic institution other than in partial fulfilments of the requirements stated above.

I have read and understood the Technology University Dublin’s policy regarding plagiarism.

X

James Finglas

# Abstract

This report documents my testing of a multimodal biometric system. The objective is to demonstrate our understanding of the requires of the testing phase.

An introduction is concluded to lay out the general outline of the assignment. I began the practical elements by documenting my goals and methodology followed by my results and conclusions.  
  
All supporting material will be present on the day of demonstration and is available for submission upon request.

Contents

[Project Submission/Declaration Form 2](#_Toc27301457)

[Abstract 3](#_Toc27301458)

[Table of figures: 5](#_Toc27301459)

[1.0 Introduction 6](#_Toc27301460)

[2.0 Summary 7](#_Toc27301461)

[2.1 Research Goals 7](#_Toc27301462)

[2.3 Methodology 7](#_Toc27301463)

[3.0 Primary Research Outcomes 16](#_Toc27301464)

[3.1 Training Set 16](#_Toc27301465)

[3.2 FRR 17](#_Toc27301466)

[3.3 FAR 18](#_Toc27301467)

[4.0 Conclusions 20](#_Toc27301468)

[5.0 References 21](#_Toc27301469)

# Table of figures:

[Figure 1 My Attempt to create a custom GUI for my multi-modal biometric system. 8](#_Toc27300626)

[Figure 2 My access database tables with manually entered test data. 9](#_Toc27300627)

[Figure 3 My attempt to add data to the dataset and save it. 9](#_Toc27300628)

[Figure 4 data added to my dataset 10](#_Toc27300629)

[Figure 5 My fatal error which prohibited me continuing on the automation path. 10](#_Toc27300630)

[Figure 6 Successful accusation of license. 11](#_Toc27300631)

[Figure 7 Here we see an attempt to enroll. 11](#_Toc27300632)

[Figure 8 Identifying a user with the SDK by comparing a user template file against a user image file. 12](#_Toc27300633)

[Figure 9 Here we see my pre-prepared Excel spreadsheet. 13](#_Toc27300634)

[Figure 10 A successful enrollment 15](#_Toc27300635)

[Figure 11 A failed enrollment attempt. 16](#_Toc27300636)

[Figure 12 An example of my FRR testing process. 17](#_Toc27300637)

[Figure 13 Sample FAR attempt with a rejection of both face and iris. 18](#_Toc27300638)

# Introduction

Our final assignment of the year was to implement and test a multi-modal biometric authentication system. The implementation phase follows on from our previous assignment which required us to design said multi-modal biometric system and the testing phase.

The implementation phase could be approached in one of two ways.

* Firstly, the implementation could be manual, where by the training set of enrollment data is added by hand and the false rejection rates and the false acceptance / circumvention rates are calculated manually also. This would be a simply, yet time consuming method.
* Secondly, the implementation could be handled by a custom designed automated program, which would involve first custom designing the graphical user interface aspect of the program, before modifying and incorporating this GUI into an already existing system such as the Neurotechnology™ Software Development Kit (SDK) programs that are freely available for downloading and using under trial license [1].

It is important to note however, that the implementation phase is only a means to an end and the testing phase of the implementation is the primary objective in the assignment. The testing phase is nessecary to demonstrate the feasibility of our system.  
  
For example, in the case of my system, I elected to develop and implement an ‘ANDED’ system. Under this system a user would first have to identify with my first biometric, ‘AND’ then have to verify with my second biometric. Failure to identify in the first, negates the need to screen for the second biometric and the user fails to enroll.  
  
However, a user would have to attempt to identify with the first biometric ‘OR’ verify with the second. I believe that an immediately flaw is revealed here.  
  
Whereby under the ‘ANDED’ system, a user must identify and then verify, a high level of security is maintained. Were the user must declare their identity on the system, and then verify they are that person as recorded in the database.

But under the ‘OR’ system a user can simply identify themselves, ‘OR’ verify themselves without identification. This is an inherent security risk. Particularly if one or more biometric is particularly easy to circumvent. I believe this is the case with my system. However, I will attempt to document the Failure to Enroll rate (FTE), the False Rejection rate (FRR) and the False Acceptance rate (FAR) in order to further highlight that my ‘And’ system is objectively better than an ‘OR system’.

# Summary

## 2.1 Research Goals

The goal of the research undertaken in this assignment is firstly demonstrate our understanding of, and ability to calculate the FTE, FRR and FAR rates of a biometric system. This is absolutely vital in the development of any biometric system for comparison with other systems and for determining if the system is practically secure, acceptable in common use and performs reasonably with minimal false rejections.

Along-side this it allows to demonstrate the feasibility of our design, and its superiority over other potential implementations.

## 2.3 Methodology

I initially elected to attempt a fully programmed automated system. To achieve this, I used Visual Studio community edition version 2019. Within this platform I added the following packages:

* + - Visual Studio core editor
    - .NET desktop Development
    - Desktop development with C++\*
    - Individual components
      * + NuGet package manager
        + .NET Framework 4.6.1 targeting pack
        + C# and Visual Basic Roslyn compilers
        + C# and Visual Basic
        + .NET Framework 4.7.2 targeting pack
        + MSBuild
        + .NET Framework 4.5.2 targeting pack
        + .NET Framework 4.5 targeting pack
        + .NET Framework 4 targeting pack
        + .NET Framework 4.5.1 targeting pack
        + .NET Framework 4.6 targeting pack
        + .NET profiling tools
        + Advanced ASP.NET features
        + .NET Framework 4.6.2 targeting pack
        + .NET Framework 4.7 targeting pack
        + .NET Framework 4.7.1 targeting pack
        + .NET Portable Library targeting pack
        + C++ ATL for latest v142 build tools (x86 & x64)
        + C++ MFC for latest v142 build tools (x86 & x64)
        + Windows Universal CRT SDK
        + MSVC v140 - VS 2015 C++ build tools (v14.00)
        + .NET Native
        + .NET Framework 3.5 development tools
        + Visual Studio SDK
        + .NET Compiler Platform SDK
        + .NET Framework 4.6.1 SDK
        + .NET Framework 4.6.2 SDK
        + .NET Framework 4.7 SDK
        + .NET Framework 4.7.1 SDK
        + .NET Framework 4.7.2 SDK

Along-side Visual Studios I used Microsoft™ Access Database. This was to be the database software I would use to store my enrollment, FRR and FAR related data. The use of Microsoft™ Excel will also be considered ins cope should manual data entry and saving be required.

For the capturing of biometrics data, I used both the Neurotechnology™ SDK version 11.2 suite and the Neurotechnology™ demo programs, also version 11.2.

I began my attempt to automate the data acquisition process by designing a custom GUI based on work we had done in our labs.

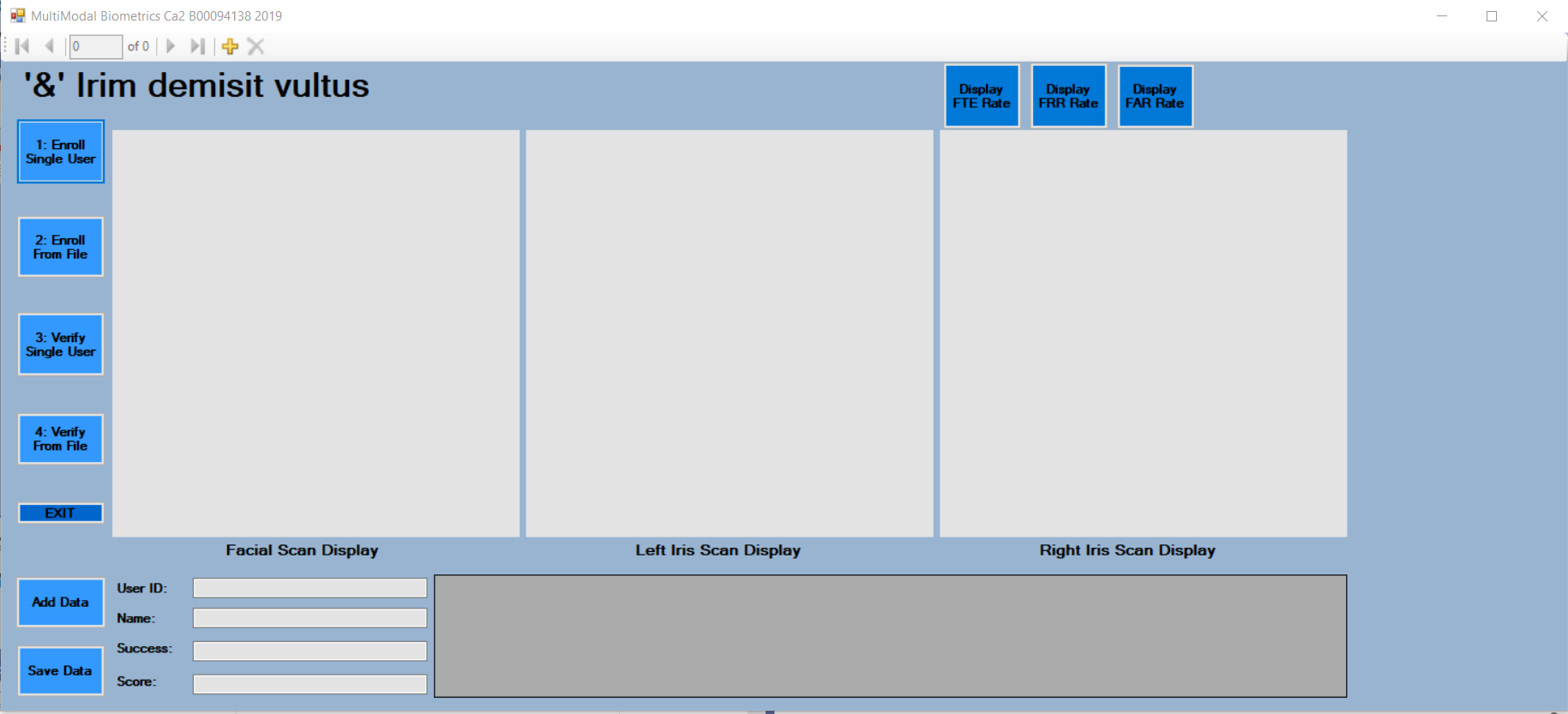


Figure My Attempt to create a custom GUI for my multi-modal biometric system.

As we can see above in figure 1, I successfully completed my GUI implementation. However, this was simply the first step of the process of automating the data accusation process. The next step was to successful get my code to record data to the programs database. And this is where I ran into an unsurmountable problem.

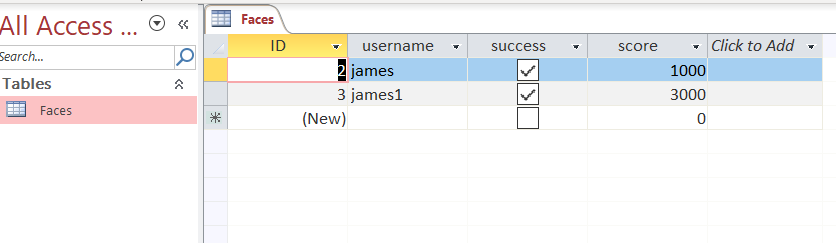


Figure My access database tables with manually entered test data.

Above in figure 2, we can see my access database table with some manually entered entries for testing purposes. However, as we can see in figure one, this test data is not returned and displayed within my program. Furthermore, while I can add data to the dataset, this dataset data will not transfer to the table as I will now demonstrate.

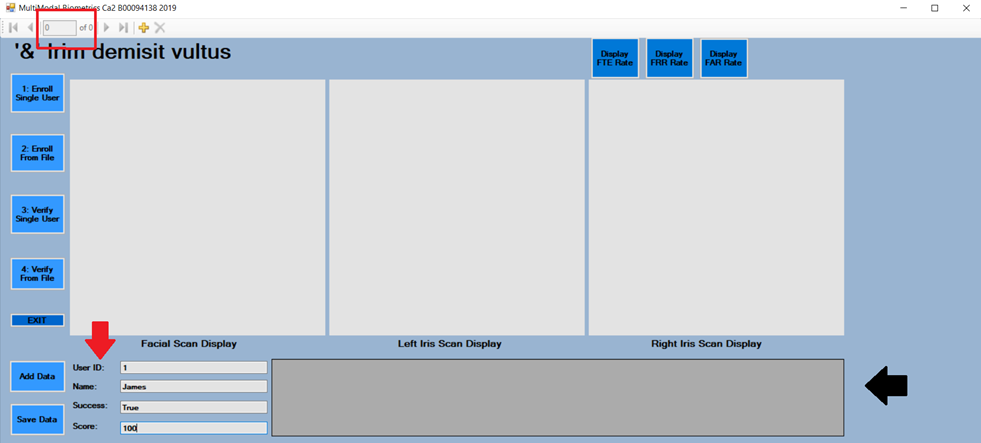


Figure My attempt to add data to the dataset and save it.

In figure 3 above, you can see my attempt to add data to my dataset and save it to my database. Please not the black arrow highlighting the fact that my manually added test data is not displayed indicating a minor communication issue between my program and database. The red arrow indicates the data I am trying to add. The red box highlights that there is currently 0 of 0 items in my dataset, yet again highlight a communications issue between my dataset and database.

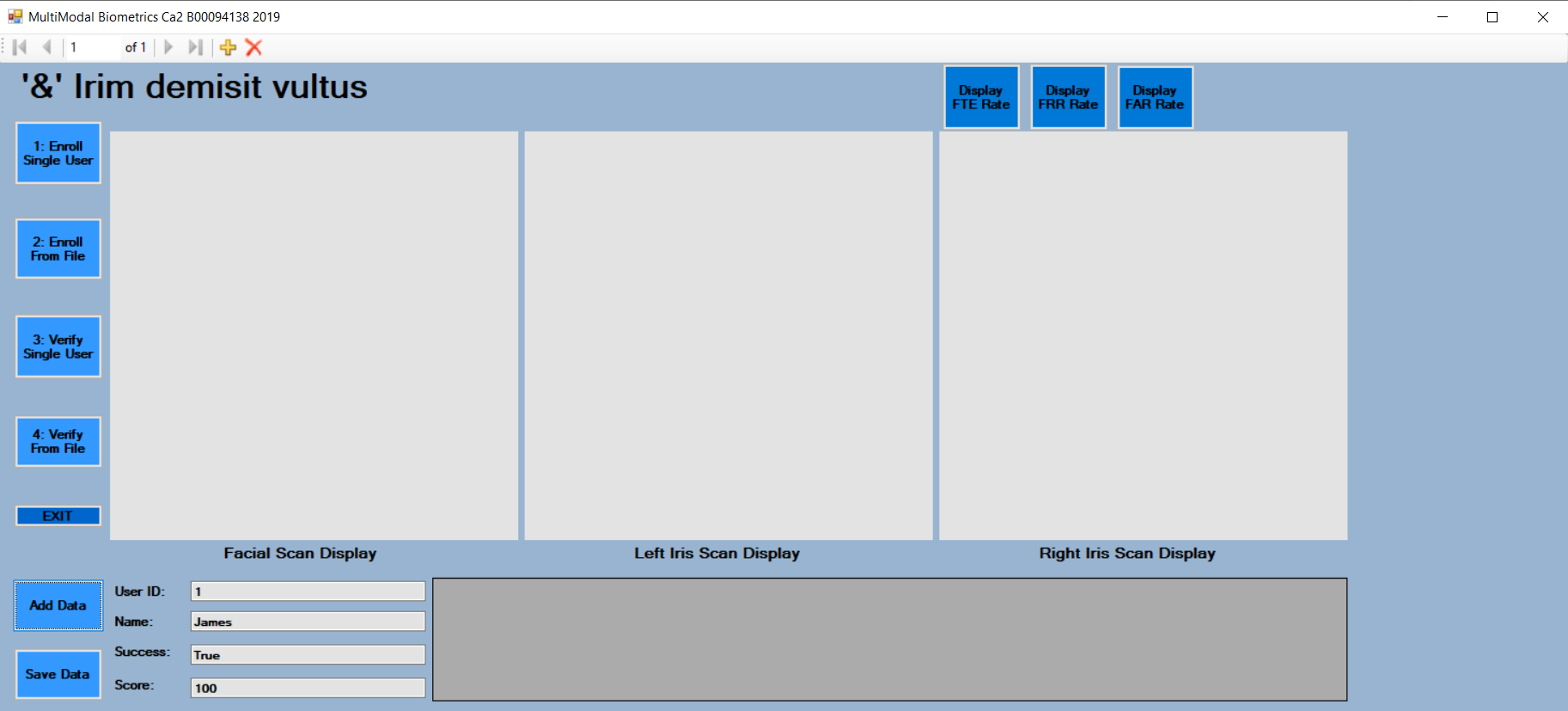


Figure data added to my dataset

Here in figure 4, we can see top left; 1 of 1 rows of data added to my dataset (red arrow). However, when I attempt to save this data to my database, I am struck with an error I could not diagnose due to time constraints and ack of experience with this language, platform and a lack of available information that I could access within the time I had.

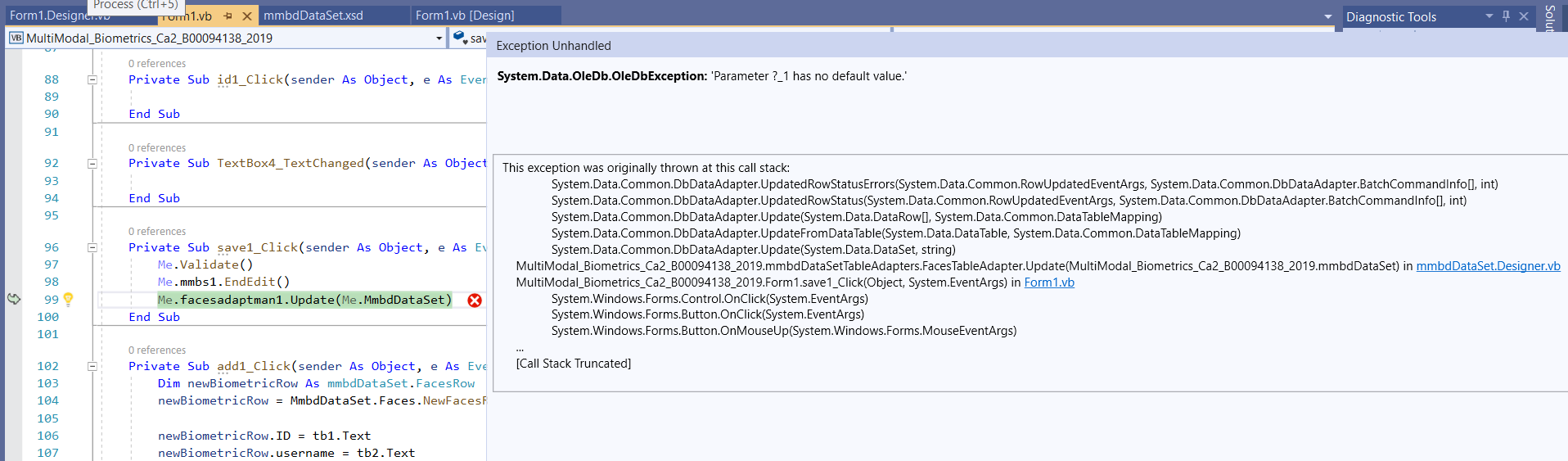


Figure My fatal error which prohibited me continuing on the automation path.

Figure 6 documents the error I suffered upon attempting to save the data contained within my dataset to my database. This was the last thing I needed to fix before transplanting my GUI into the Neurotechnology™ SDK.  
  
While there is no guarantee that I would have successfully implemented the modifications within the SDK, I must confess that I found myself feeling very defeated by this failure and I have committed myself to rectifying this and achieving the full automation of this assignment beyond the scope of this project. However, this did end my attempts to automate this assignment.

Next, I moved onto to the fallback option of manually entering the data. At this point I ran into two further issues. Firstly, due to circumstances beyond my control, access to biometric hardware (in my case, an iris scanner) was extremely limited. In fact, at the time I began to document my report, I had yet to gain access to the hardware. Nevertheless, I proceeded with my planning. At this point I came across another issue.  
  
I began my testing of the SDK by acquiring licenses as seen below.

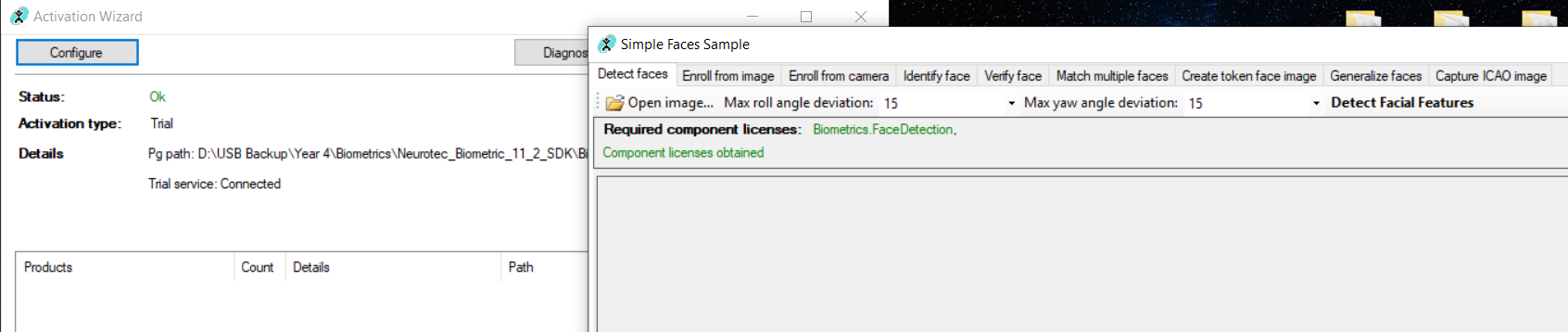


Figure Successful accusation of license.

With my license acquired I began testing the SDK software and I immediately noticed several problems.

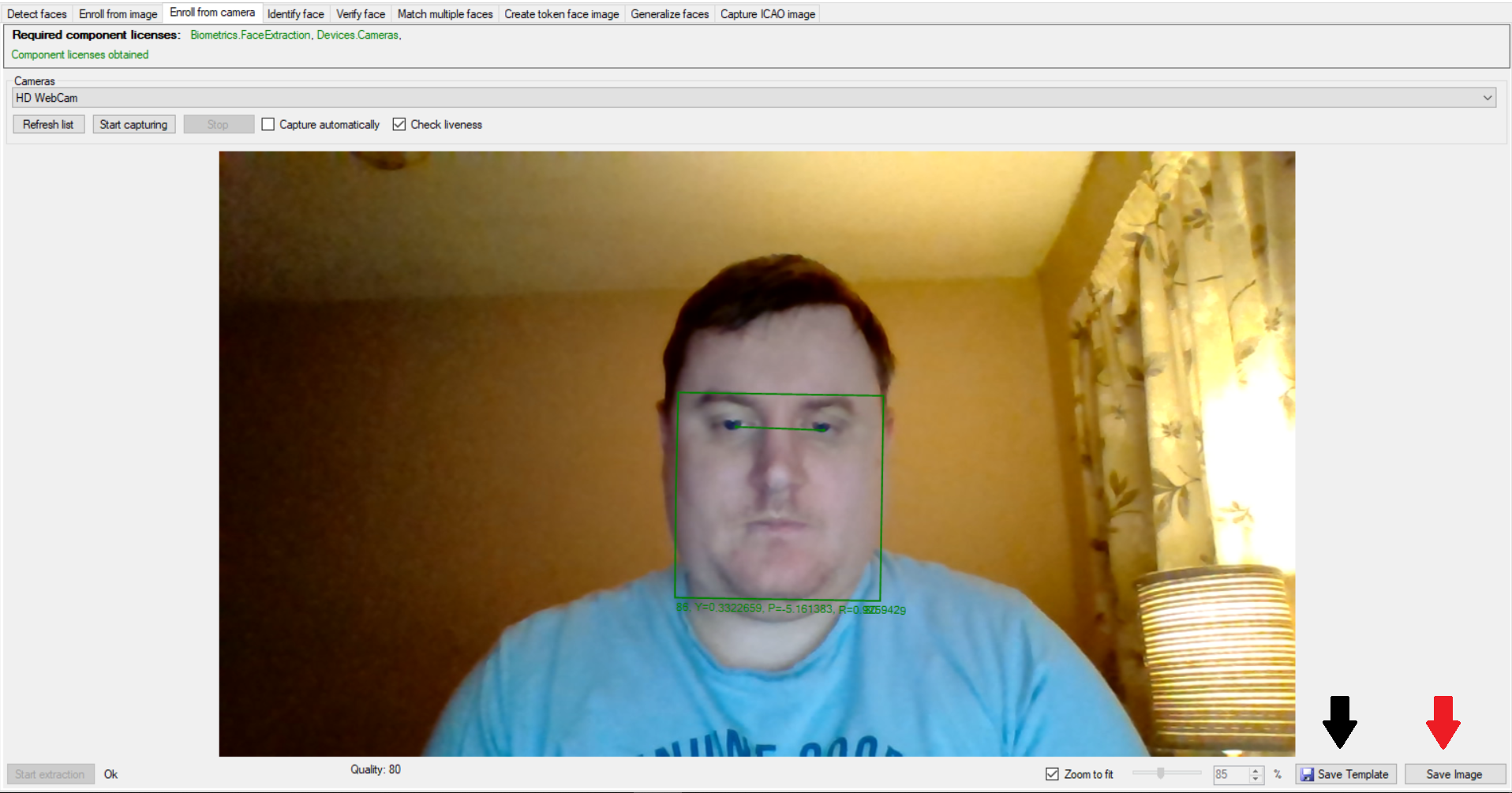


Figure Here we see an attempt to enroll.

As we can see above in figure 7, I attempted to enroll and discovered I could not save my enrollment to a database. Instead I could only save the capture as a template (black arrow)

or as an image file (red arrow). I chose to save it as a file under the name James1, and as a template under the name James2. Next, I tried to identify myself from the template file. The process here involved selecting the template image manually, and the image manually; in order to compare the template against the image file as seen below in figure 8. This process repeats for verification.

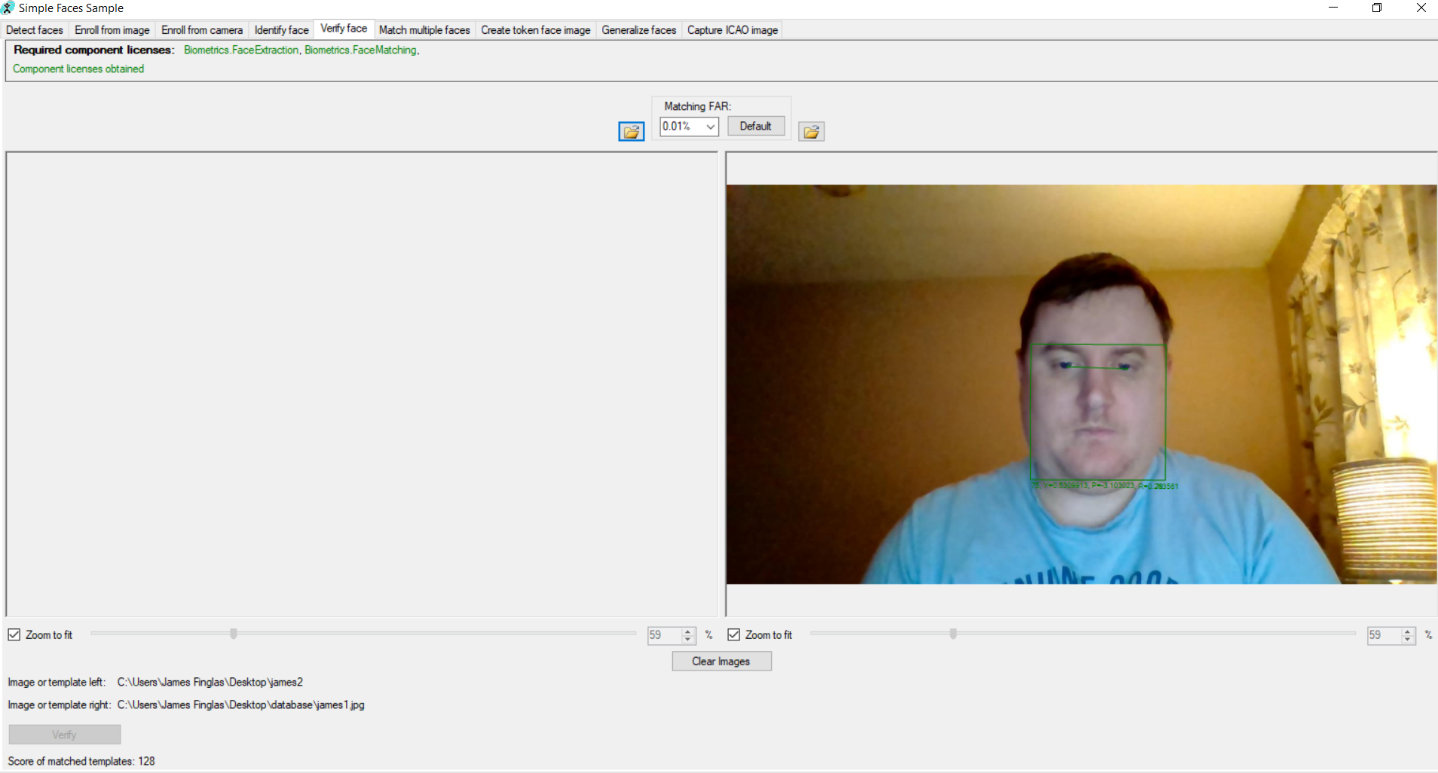


Figure Identifying a user with the SDK by comparing a user template file against a user image file.

So, the process of enrollment would require the following steps:

* Enroll user from camera
* Save capture as a template file in one location
* Save capture as an image file in another location
* Load template file for identification
* Load Image file for identification
* Identify
* Document success or failure
* Document score
* Load Template file for verification
* Load Image file for Verification
* Verify
* Document success or failure
* Document score
* Document full system success or failure
* Document combined score

Given that I planned to enroll a minimum of 100 users. And test approximately 5% of my training set minimum for both FRR and FAR, this meant a minimum of approximately 100 enrolls and 1000 FRR/FAR scans. This is simply prohibitively too long in terms of the time it would take to carry out this assignment even if I had adequate access to hardware. Therefore, I have deemed the SDK suite, due to its lack of incorporated database; unusable.

This left me with one option. The Neurotechnology™ Demo software. This software is surprisingly robust, functional and usable. Both Verilook and VeriEye contain internal databases. Enrollment is fast, as is both facial identification and iris verification. This is clearly the best way to proceed forward and begin to process of acquiring the data I need to calculate the FTE FRR and FAR rates. Therefore, I elected to use the Demo programs for my assignment.

One final step remained once this decision was made, prior to beginning the data acquisition and documentation phase. Since I have elected to manually scan in pictures using the internal demo database, I will have no access to an automatic database to record my scores. There for I must create an Excel spreadsheet to document my data.

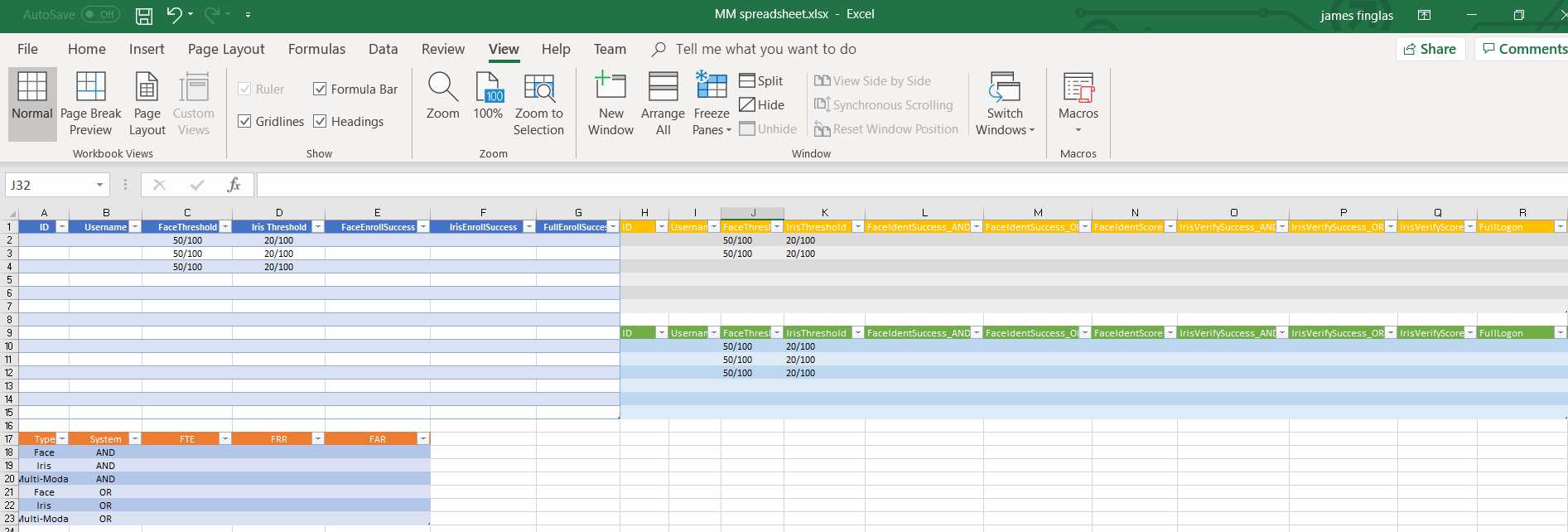


Figure Here we see my pre-prepared Excel spreadsheet.

As we can see above, I created an Excel spreadsheet containing 4 tables. The first documents the Enrollment attempts which will give me my training set. This table contains an ID, Username, my threshold settings, success columns which will document whether the enroll attempts pass or fail via a true/false system; for each biometric type as well as a combined Success column. This table is color coded in blue.

Next, we have my FRR Table which splits the success columns in success in the add system ‘AND’ success in the OR’ for later statistical comparisons. This table is color coded to yellow.

Next, we have my FAR table which follows the same format as my FRR table and is color coded to green.

And Lastly, we have my rates table which will be used to calculate my FTE, FRR and FAR rates for both individual systems and combined systems, as well as ‘AND’ and ‘OR’ systems. This table is color coded to orange.

Ratings formulas:

* FTE rates will be calculated using the following formula: failed enroll attempts \* 100 / training set.

For example: total user enrolled = 100. Training set = 102. Failed enroll attempts = 2.

2 \* 100 / 102 = FTE rate.

* FRR rates will be calculated using the following formula: false rejection total \* 100 / Total attempts to enroll enrolled users

For example: 20 attempts to identify/verify. 5 false rejections.

5 \* 100 / 20 = FRR rate.

* FAR rates will be calculated using the following formula: successful false attempts to identify/verify (circumventions) \* 100 / total attempts to falsely identify/verify.

For example: 20 attempts to circumvent. 1 success.

1 \* 100 / 20 = FAR rate.

# Primary Research Outcomes

## 3.1 Training Set

As previously described in my methodology, I created an Excel spreadsheet for documenting my results. I began by enrolling my Training Set into my enrollment table. While my initial decision was to slightly increase the thresholds of both biometrics, I did some test enrollment and discovered that this led to a serious increase in FTE rates with face. Therefore, I decided to rely on the strength of Iris circumvention difficulty for security and retained the default Threshold levels of 50/100 for face and 20/100 for iris. This remained consistent throughout my testing.

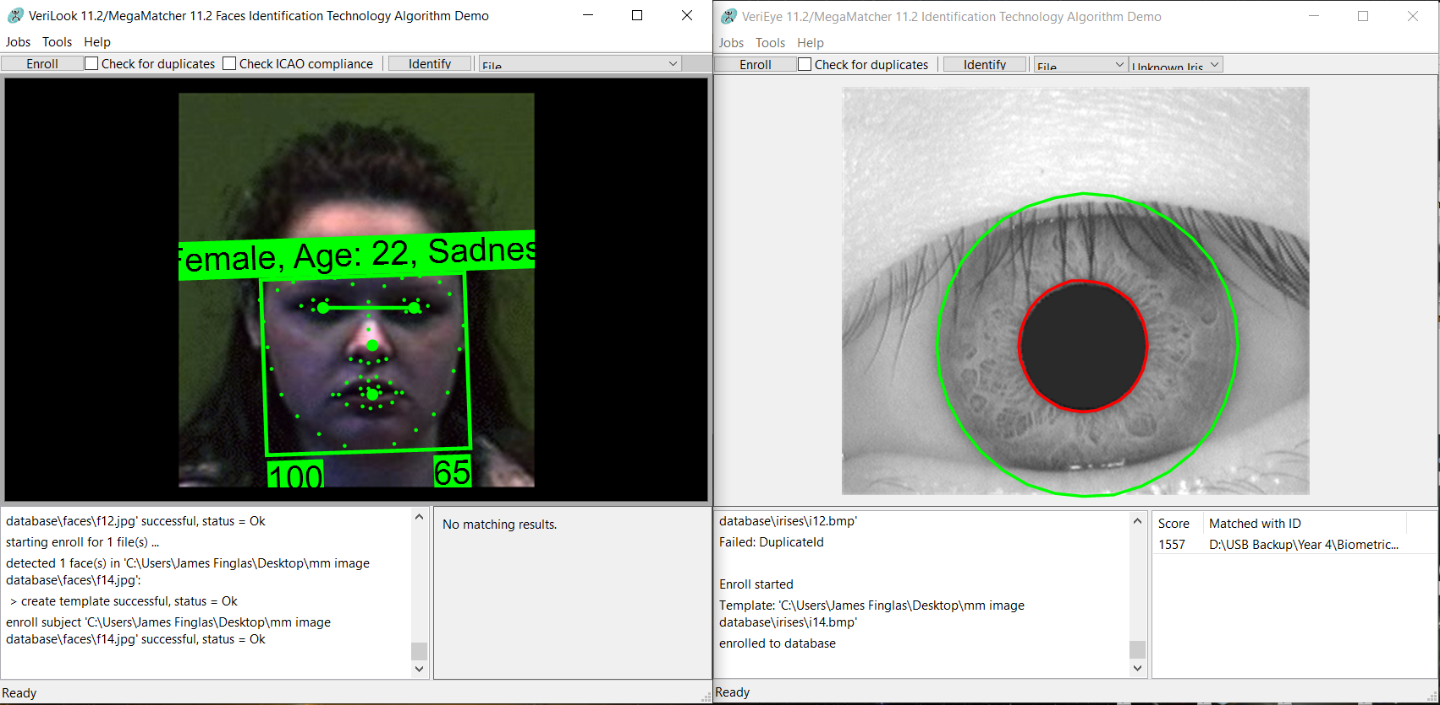


Figure A successful enrollment

Above, in figure 10; we can see a successful enrollment into my training set. However, I also had some failed attempts to enroll, a sample of which can be seen below in figure 11.

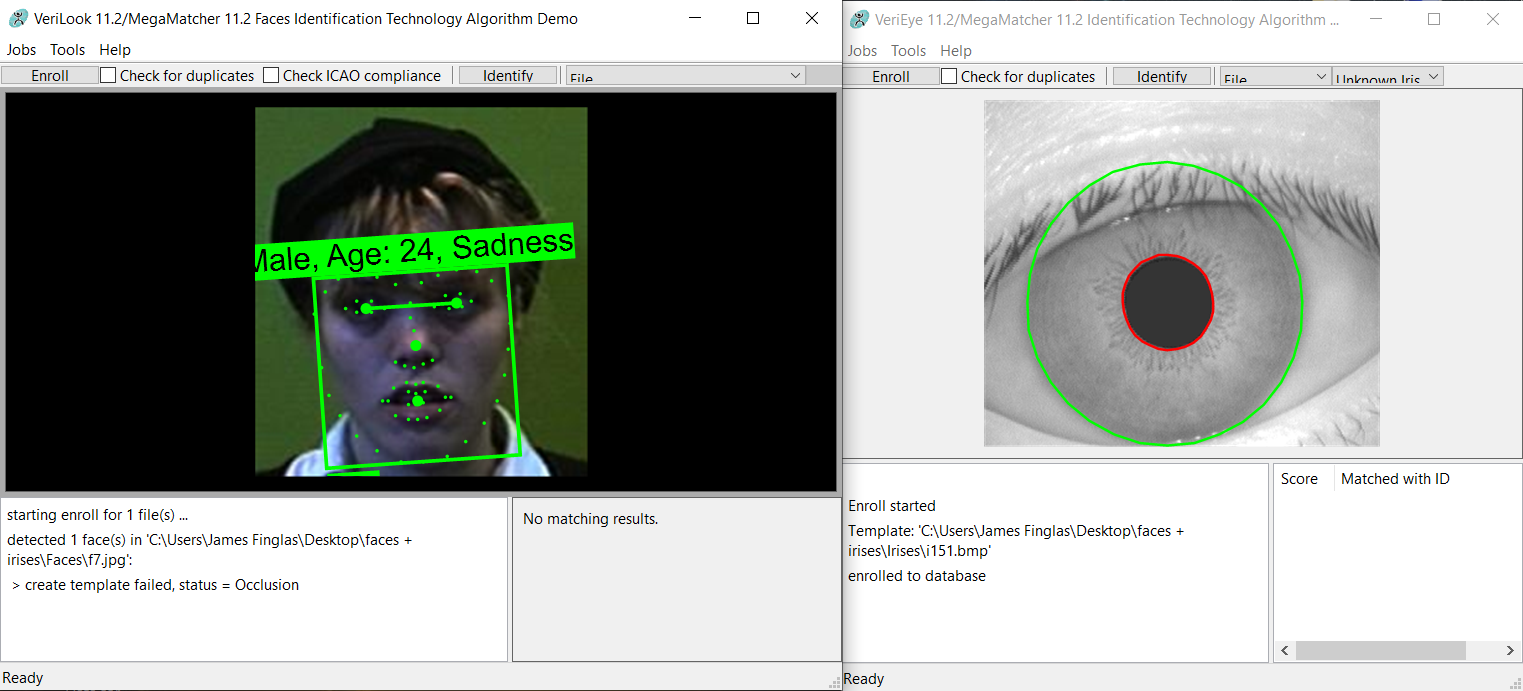


Figure A failed enrollment attempt.

The total enrollments were 100, with 9 failed attempts giving me a training set of 109. Each of my attempts was documented in full in Excel spreadsheet.

## 3.2 FRR

Next, I moved on to determining my FRR rates. This requirement me to attempt to enroll some users from my enrollment set. I decided to use 5% of my enrollment set for a total of 5 users. Once I began my FRR attempts I noticed an interesting pattern.

My process involved me testing multiple pictures of each user multiple times. I began with a set of 20 pictures of each user. I discarded the first picture as this was the picture I used to enroll. This left me with 19 images. I then added the iris image used to enroll. This gives me 19 sets of face + iris combinations for testing. Each of these sets I attempted to identify with face biometrics and verify with iris biometrics 5 times each. This gives me 95 attempts to identify / verify per user for a total of 475 FRR attempts.

During the process of the scans I noticed that for each picture, the score came out consistent. This means, that because I was lucky enough to choose picture with proper lighting and the users hold the correct possess, the pictures were always expected to pass. What I determined from this is the following. Without physical access to hardware and a proper implementation of the biometric system one cannot truly test a biometric system, however; one can test the testing methodology of the biometric system. Since the main purpose of this assignment is to demonstrate our ability to test a biometric system and out testing process, I consider the testing process valid and a success. If the goal was to actually test a biometric system, I would consider this test invalid.

I had no failures to enroll. This make my equation as follows: 0 \* 100 / 475 giving me an FRR rate of 0%. Again, had I been using hardware, I would have expected the number to be different.

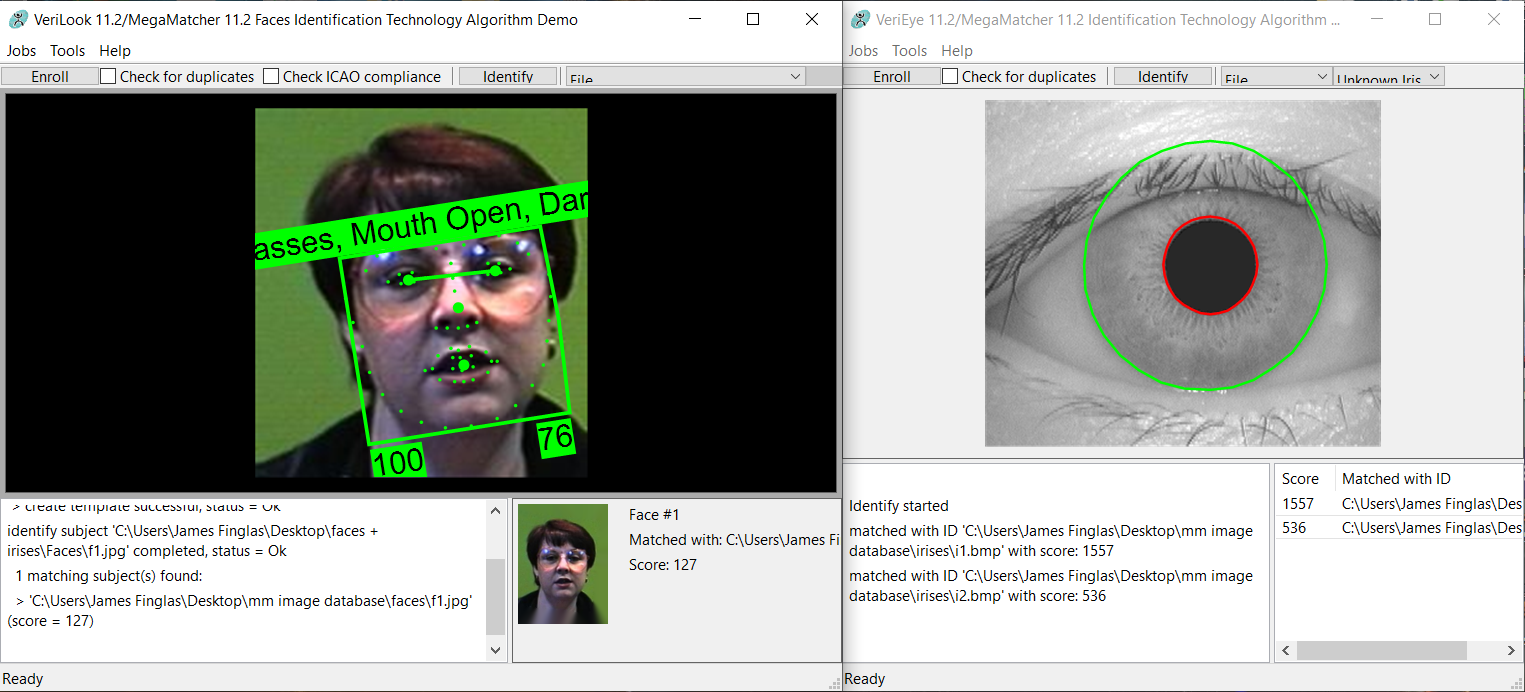


Figure An example of my FRR testing process.

Above, in figure 12, can see a sample of my FRR testing with the above user identified with a score of 127 in facial biometrics and verification score of 536 in iris biometrics.

## 3.3 FAR

The FAR process followed the FRR process with 1 small change. As I had not enrolled any users from these images, I had 20 images for each user. This makes my total attempts for FAR 500. Once again, as I was attempting to identify and verify from file. I expected to see predictable patterns, and this was the case. I had 0 circumventions / FAR’s. I would have expected this for iris as this is very difficult to circumvent. But I would have expected some circumventions were I using hardware and attempting to fully manually circumvent.

So, with a FAR score of 0, my calculation was as follows: 0 \* 100 /500 giving a FAR score 0%.

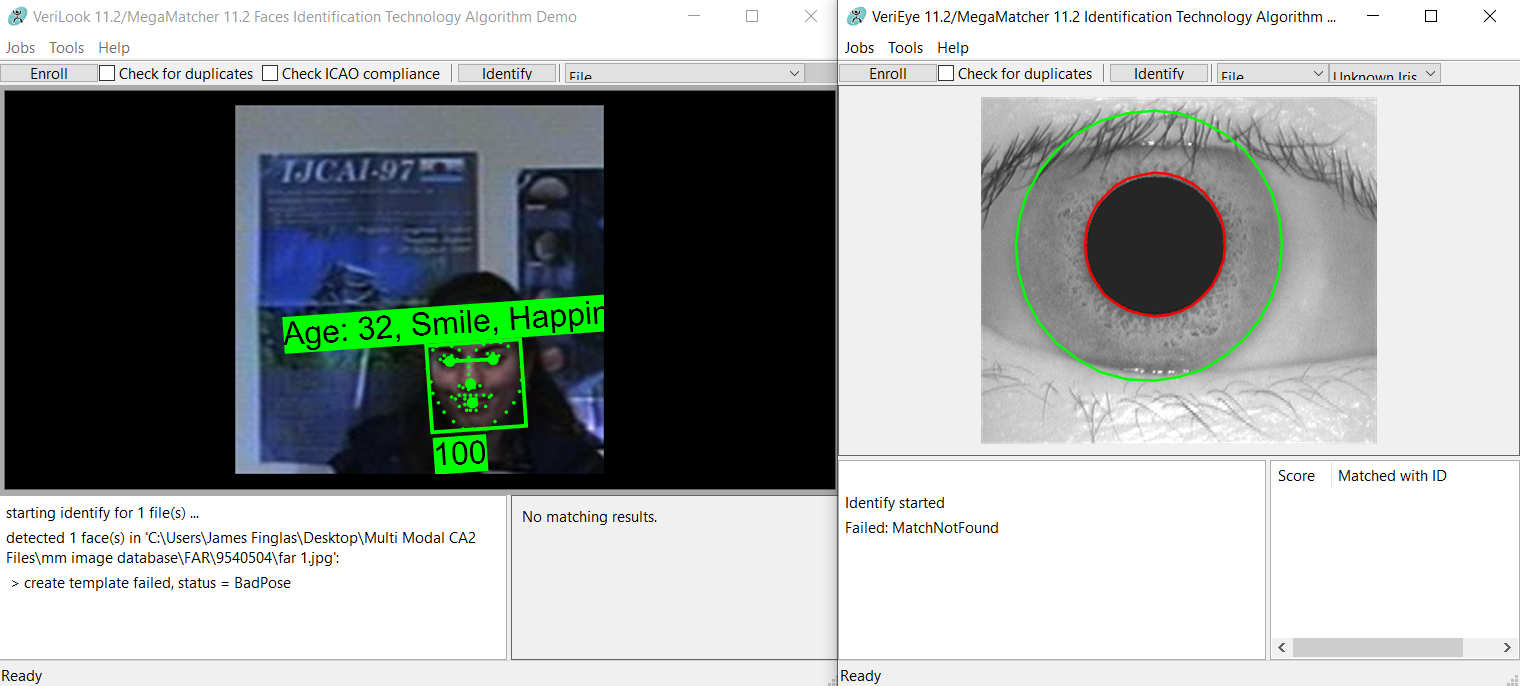


Figure Sample FAR attempt with a rejection of both face and iris.

# Conclusions

What I can take from these results is that the testing phase should not be conducted prior to full actualized implementation of a biometric system with hardware integration. However, one can determine if the testing system they intend to use is valid. And I feel that based on my results, my testing system is valid.

My FTE rate was high, but I know that this is since I am enrolling from file and thus, lighting conditions and user pose, and facial expression are out of my control. Therefore, I cannot ensure that the users do all they can to avoid a failure to enroll, such as removing glasses, parting hair ad adding extra light. However, it was important that I have failures to validate the testing process. For this reason, while the rate was high, I consider it acceptable for the purposes of this testing process.

Both the FRR and FAR rates were entirely predictable once I saw the patterns emerging during testing. However, I could also see the validity of the testing process. I would fully expect these to alter if I was unlucky enough to have picked one of the picture sets with contain some images that fail to be enrolled. I did in fact test this and verify that some sets did exist that contained some pictures that would enroll, but also some that would not. So, the fact that FRR came out at 0% was entirely by luck. Thus the FRR process is valid.  
  
FAR is a slightly different case. While I could have manually circumvented the faces, I feel that this would have skewed my results since all other data came from file only. Since the process for FRR Far follows the same methodology I am confident that my process in FAR is valid.

Lastly, since we know that with my ‘AND’ system, if a user fails the facial identification, he or she may not proceed to the iris verification process; whereas in an ‘OR’ a user would be able to proceed and attempt to verify even if identification fails, the ‘AND’ system stands out as the more secure system. While the ‘OR’ system allows for redundancy. However, since the purpose of biometrics is generally twofold, security and auditing it is very important to keep out who should not enter, and to document who does enter. For this reason, it is the opinion of this student, that the ‘AND’ system is superior to the ‘OR’ system.

I stand by all my results and present for report and results for demonstration and evaluation at my lecturers’ discretion and convenience.

# 5.0 References

1. Neurotechnology™ (Nov 5th, 2019), ‘MEGAMATCHER, VERIFINGER, VERILOOK, VERISPEAK AND VERIEYE SDK TRIAL’, [ONLINE] Available at: <http://www.neurotechnology.com/download.html#megamatcher_verifinger_verilook_verieye_sdk_trial> [Accessed: Dec 12th, 2019]