Adapting to Movement Patterns for Face Recognition on Mobile Devices

Matthew Boakes[1 [0000−0002−9377−6240]], Richard Guest[1 [0000−0001−7535−7336]], and Farzin Deravi[1 [0000−0003−0885−437X]]

University of Kent, Canterbury, Kent, England, CT2 7NZ {mjb228, r.m.guest, f.deravi}@kent.ac.uk

Abstract. Facial recognition is becoming an increasingly popular way to authenticate users, helped by the increased use of biometric technology within mobile devices, such as smartphones and tablets. Biometric systems use thresholds to identify whether a user is genuine or an impostor. Traditional biometric systems are static (such as eGates at airports), which allow the operators and developers to create an environment most suited for the successful operation of the biometric technology by using a fixed threshold value to determine the authenticity of the user. However, with a mobile device and scenario, the operational conditions are beyond the control of the developers and operators.
In this paper, we propose a novel approach to mobile biometric authentication within a mobile scenario, by offering an adaptive threshold to authenticate users based on the environment, situations and conditions in which they are operating the device. Utilising smartphone sensors, we demonstrate the creation of a successful scenario classification. Using this, we propose our idea of an extendable framework to allow multiple scenario thresholds. Furthermore, we test the concept with data collected from a smartphone device. Results show that using an adaptive scenario threshold approach can improve the biometric performance, and hence could allow manufacturers to produce algorithms that perform consistently in multiple scenarios without compromising security, allowing an increase in public trust towards the use of the technology.

Keywords: Mobile · Face · Adaptive · Threshold · Motion · Scenario · Classification

1 Introduction

Biometric facial recognition is a useful security tool, allowing a method of authentication with little interaction from the users' perspective since images can be captured from a distance and while in motion merely requiring the use of a camera. The technology has gained mass traction in recent years with its incorporation into mobile devices.

Facial recognition has its share of criticism as campaigners claim the current technology is inaccurate, intrusive and infringes on an individual's right to privacy [1]. Recently, several locales have implemented or are considering implementing a ban of fixed system facial recognition technology, including San Francisco [16] and the European Union [1]. In order to support broad adoption of the technology, we must assure to certify that it is 'fit for purpose' and one method to achieve this would be to ensure consistently high recognition accuracy across a range of scenarios.

With a camera now installed on the majority of smartphone devices, it is becoming increasingly convenient to take a self-portrait image ('selfie') intended for facial recognition. Service providers are increasingly asking for users to submit an ID document photo alongside a selfie, captured on a mobile device, to authenticate their claimed identity as part of Electronic Identity Verification (eIDV) services [8]. Furthermore, smartphones now increasingly incorporate facial technology allowing users to verify themselves as well as access services and resources within the device and beyond.

Static biometric systems, fixed in position, such as airport eGates, have been in use for a while. In these scenarios, the operators have great control over the environment to help optimise recognition performance. The same is not valid with mainstream mobile biometrics, where the operator has no control over the operational environment. It, therefore, stands to reason that mobile biometrics would require a more adaptive approach for handling the authentication system.

In this paper, we describe a proof-of-concept adaptive model for mobile devices which has the potential to outperform a static threshold applied to all environments and usage conditions. Section 2 introduces related work and our inspiration for this. Section 3–Section 4 introduces our data collection and discuses how movement scenario impact recognition performance. Section 5 introduces our theory behind an adaptive framework to better deal with changing movement patterns. Section 6 discuses our approach to a scenario detection algorithm Sections 7–8 shows our experimental work and results in testing the adaptive threshold algorithm and Section 9 draws conclusions and suggests future work.

2 Related Work

The concept of adaptive biometrics systems is not new as Pisani *et al.* [23] has provided a comprehensive review of adaptive biometrics systems. However, the majority of approaches work by updating the biometric reference over time usually to account for template ageing. Pisani *et al.* note how "there is still a limited number of studies that evaluate adaptive biometric systems on mobile devices" and how researchers "should also acquire data from the sensors on these devices over time". Here we take a condition-sensitive (and quality index) adaptation criterion approach based on Pisani *et al.* taxonomy.

Our method intends not to alter either the sample or the probe but to utilise the mobile device's sensor information to determine the operation scenario and set thresholds accordingly. Techniques utilising the sensors embedded into smartphones and combining them with the biometric authentication process are present in the literature. Including the creation of behavioural biometric data to assess unique traits to identify individuals either independently or as part of a multimodal system with another physical or behavioural biometric trait to produce accurate biometric systems, commonly for continuous authentication purposes [21,25,15]. Another involvement of smartphone sensor data is in liveness detection [17] and defending against presentation attacks, Chen *et al.* [5] demonstrated a presentation attack detection approach to use the motion sensors to defend against 2D media attacks and virtual camera attacks.

The need for a more adaptive recognition framework is present in the literature as aspects like movement and portability of the device can vary between enrolment and recognition phases [24]. We previously [2] highlighted the potential factors that can affect a mobile biometric system and highlighted 'Scenarios' as one of these factors by categorising them under 'Stationary' and 'Motion'. Gutta *et al.* [12] have filed patents that suggest work and ideas relating to an adaptive biometric threshold, including the use of a light intensity sensor to assist in adjusting the threshold value in a facial recognition system. Similarly, Brumback *et al.* [3] (Fitbit Inc) has also filed patents for continuous authentication purposes on wearable technologies such as smartwatches and fitness trackers. However, they provide no practical examples of the proposals for mobile systems. Castilllo-Guerra *et al.* [4] proposed an adaptive threshold estimation for voice verification systems allowing the threshold to adapt to specific speakers. Similarly, Mhenni *et al.* [20] proposed to use an adaptive strategy specific to each category of users while investigating using Doddington's Zoo classification of user's keystroke dynamics.

Lunerti *et al.* [18] showed that for face verification in a mobile environment "it can be possible to ensure good sample quality and high biometric performance by applying an appropriate threshold that will regulate the amplitude on variations of the smartphone movements during facial image capture". This paper aims to contribute in showing how an adaptive approach can be the answer to having an "appropriate threshold" and begin to explore the gap in mobile biometric adaptive systems by exploring the potential impact of motion scenarios on recognition performance. We aim to answer the following question: can we improve mobile biometric recognition performance and security by using an adaptive approach to the decision component using knowledge of the operating scenario?

3 Data Collection

To trial this approach, we conducted a data collection. This paper will focus on the results achieved using the Android-based Samsung Galaxy S9 smartphone device. We developed a custom application to collect and capture data from this device in an attempt to mimic a biometric authentication. Using the Samsung Galaxy S9, we were able to collect data including a 'selfie' image taken by the participant in the scenarios and background metadata obtained from the multitude of sensors (including accelerometer, gyro sensor and geomagnetic sensor) within the device. The device features an 8 megapixel (1.22 μm, f/1.7) front-facing camera. However, the default front picture size captures images at 5.2 megapixels meaning for the study we had images of resolution 2640x1980.

We had a total of 25 participants who completed this part of the study during one session visit. We tasked participants with operating the device in a variety of scenarios, the order of which was:

1. Sitting - Participant sat down in a chair.
2. Standing - Participant standing.
3. Treadmill - Participant walking at a steady speed on a treadmill (speed set by the participant).
4. Corridor - Participant walking at a steady speed down a corridor.

The aim was to mimic likely scenarios for smartphone use, the exception being treadmill, where the aim was to create a controlled walking scenario. We wanted to ensure the tasks were not too strenuous owing to the repetitive nature of repeat biometric transactions. The theory is to test the approach on indoor scenarios in typical biometric authentication environments (room lighting), allowing us to focus specifically on motion and movement. However, we would like to see the approach adapted to other scenarios and factors in the future.

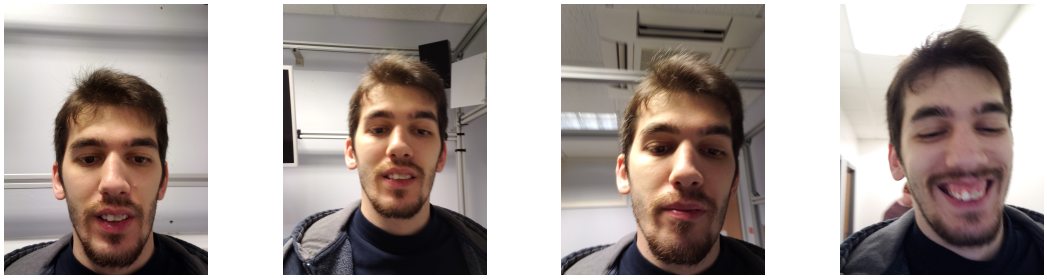
In each of the scenarios, the participant held the device with their own hands as they usually would when operating a smartphone device. The participants were pre-enrolled at the start of the session using the device's biometric system while in the seated position. For each scenario, we asked the participant to take a 'selfie' image. We deliberately did not make any recommendations on how to position the face within the image; the only requirement was that the face was within the image, as an additional part of the experiment was to see the impact on the device's facial recognition system.

Once the participant had captured the image, they remained in the same position, including the handling of the device. They were then presented with the device's in-built Android BiometricPrompt [10] to perform an authentication. While this was happening, we simultaneously collected the metadata (sensors, including Gyroscope, Linear Acceleration, Magnetic Field, Orientation) from the moment the device's face authentication started until the process had finished utilising the abilities of Android SensorManager [9]. As the face recognition authentication can be over within a second, we wanted to make sure we collected as much sensor data as possible. Therefore we set the sensor delay to 0.005s; however, we should note as stated in the documentation "this is only a hint to the system. Events may be received faster or slower than the specified rate".

Figure 1 shows examples of one captured 'selfie' image from each tested scenario, taken by a single participant in our study. Table 1 displays the number of images we collected from each scenario and within how many of those the facial recognition algorithm we used was able to detect a face. The work in this paper uses the images where the algorithm detected a face. Table 2 shows the breakdown of our participant ages. We can see that 76% of our participants who used the Samsung Galaxy S9 were under the age of 30, as we are capturing within a student population. Our participants had a gender split of 52% Female to 48% Male.

4 Scenario Performance

To test whether our adaptive framework has the potential to outperform a traditional system, we needed to create a prototype. Commercial off-the-shelf smartphone devices have the biometric components tightly locked down for security and privacy concerns. Therefore, we decided to use



(a) Sitting (b) Standing (c) Treadmill (d) Corridor

Fig. 1: One example image from each scenario obtained from one participant during the first session

Scenario	Images	Face Detected	No Face Detected
Sitting	139	139	0
Standing	124	123	1
Treadmill	121	116	5
Corridor	122	120	2

Table 1: Amount of images collected from each scenario

Age Ranges	# of Subjects
19–21	3
22–24	8
25–29	8
30–39	4
40–49	2
Total	25

Table 2: Participant Age Ranges

open-source software to help create a prototype of how a potential adaptive system could perform and function. We used the open-source 'face-recognition' python library (version 1.3.0) by Geitgey [6,7] as the face recognition algorithm for our prototype. This library utilises the machine learning library 'dlib'.

For each user, we took their first sitting attempt as the enrolment reference, to act as the base-case scenario, and used the remaining images from all the scenarios as verification probes. Meaning we had a total of 114 verification probes for the sitting scenario, 123 for the standing scenario, 116 for the treadmill scenario and finally 120 for the corridor scenario. The 'face-recognition' library calculated and returned the dissimilarity distance scores (between 0 and 1) of a given enrolled sample and a new verification probe. Here, a high score indicates that two images are unlikely to be of the same person (no match), and a low score indicates that the two images are likely to be of the same person (match). The library recommends a decision threshold of 0.6, meaning, we consider all comparisons that score 0.6 or below to be the same person, and anything above is different people.

We previously [2] showed how scenarios could impact the false reject rate of the Samsung Galaxy S9 and showed the performance results from the device, although also noted how additional factors could have caused this impact. As an exploratory investigation, we used the dissimilarity score information provided by this library, and investigated if a need existed for having a different threshold for each scenario by examining the performance observed within each. We can see this by exploring how the genuine dissimilarity scores vary in each scenario. Table 3 shows this information along with the standard deviation and informs us that our average dissimilarity score for the stationary scenarios was $0.21(\pm 0.08)$, whereas the average score for our in-motion scenarios was $0.30(\pm 0.06)$. It is indicating a 43% score increase from a user being in a stationary scenario to them being in a motion scenario. We can also see that the baseline recognition performance varies across scenarios.

Here we used a total of four impostors for each genuine user as discussed in Section 7.1 and the largest FAR occurs in the same scenario as used for the enrolment. However, this is also the scenario which has a mean dissimilarity score significantly lower than the baseline threshold of 0.6, highlighting the problem and affect that using impostor probes taken in the same scenario has on the false accept rate. We believe that an adaptive threshold could provide greater security by restricting these passive impostor attacks. These findings highlight reasons for the introduction of unique thresholds into biometric algorithms.

	Mean Dissimilarity Score	**Baseline Recognition Performance**
Sitting	0.16 ($\pm$0.07)	FRR: 0.00 FAR: 11.30
Standing	0.25 ($\pm$0.06)	FRR: 0.00 FAR: 9.04
Treadmill	0.31 ($\pm$0.07)	FRR: 0.00 FAR: 8.70
Corridor	0.29 ($\pm$0.05)	FRR: 0.00 FAR: 9.41

Table 3: Performance variations for each tested scenario

5 The Adaptive Scenario Threshold

A traditional biometric system can be seen in Figure 2 from the International Organization for Standardization (ISO) based on prior work from Mansfield *et al.* [19]. The component we are

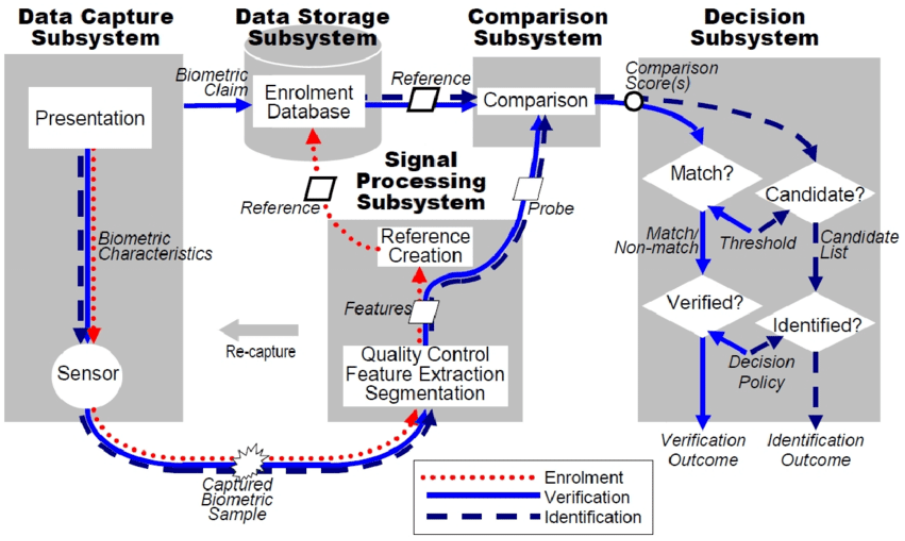


Fig. 2: Components of a general biometric system [14]

interested in here is the 'Decision' ('Matcher') component of the system. In a traditional static system, this component is relatively straightforward. We compare our stored enrolment reference to an additionally provided probe and receive a match score from the system that can determine how similar or dissimilar the two are. Having received this match score, we can use a threshold pre-defined to allow our genuine users to access the system while keeping as many impostors from accessing the system as possible. The aim is to set a threshold to keep the False Reject Rate (FRR), the percentage of genuine people rejected by the system, and False Accept Rate (FAR), the percentage of impostors accepted by the system, as low as possible. The equal error rate (EER) is the value where the FRR and FAR are identical with a low equal error rate indicating a high accuracy for the biometric system. Figure 3 shows an example of the 'Decision' component of a static system.

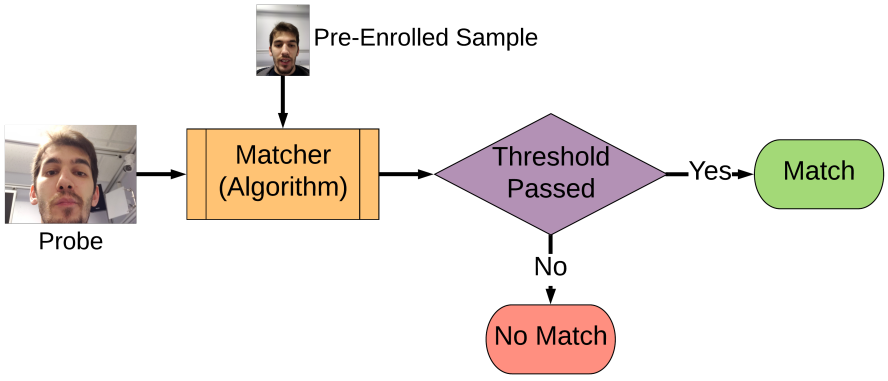


Fig. 3: A traditional matcher/decision of a biometric system

Our method addresses whether we can achieve an improvement in overall biometric performance, by adjusting the threshold adaptively, based on what we can find out from the authentication environment. When using a traditional (static) biometric system, we can create an appropriate environment and provide directions to users to help ensure optimal usage, giving the best chance of

successful authentication. However, with the unpredictability of the environments, scenarios and conditions in which mobile devices are operated within, and hence where the biometric authentication can occur, can we alter the decision threshold instead to allow for optimal performance? We present a sample of how this framework could function in Figure 4. Here we illustrate that instead of having a single threshold to cover the entire spectrum of environments and scenarios as depicted in Figure 3; we can have a separate limit set for specified situations, such as in this example using 'Stationary' and 'Motion'. To the best of our knowledge, this is the first work to utilise smartphone sensor data to classify scenarios in an attempt to create an adaptive biometric system for mobile devices by adjusting the threshold accordingly.

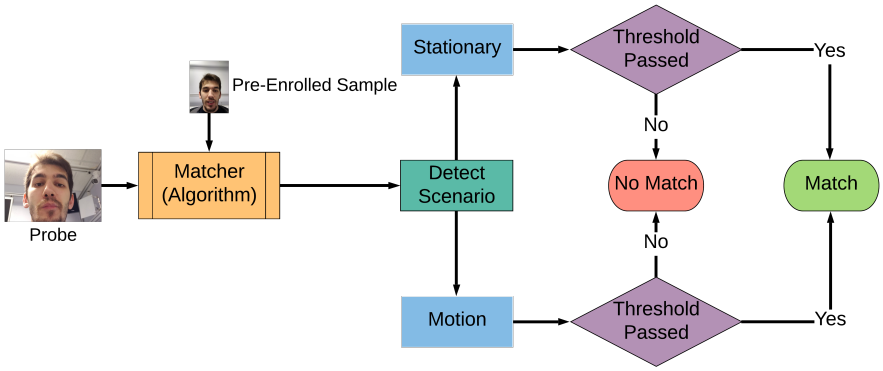


Fig. 4: An example framework for a simplified adaptive threshold decision for a biometric system

In using an adaptive threshold, we expect that we will be able to tailor the authentication experience to better deal with changing movement patterns and allow for enhanced security and user satisfaction. Our primary driver is to allow genuine users unobstructed access while keeping out passive impostors. We, therefore, considered it vital to use appropriate impostors while designing and testing the framework. This is discussed further in Section 7.1.

6 Automatic Scenario Detection

To achieve this adaptive threshold, we need a methodology to allow us to know in what scenario the user of the device was performing the authentication within. The first step is to distinguish between our 'Stationary' and 'Motion' scenarios.

We used a total of five features for our classifiers, including four of the in-build mobile sensors, two motion-based sensors, two position-based sensors and a facial image quality assessment. The motion sensors were Gyroscope and Linear Acceleration. The position sensors were Magnetometer (Magnetic Field) and phone Orientation. All of these sensors operate on an x, y, and z axis system, and the data from each channel was collected. We began collecting the sensor data from the moment the participant started the authentication until the transaction was complete (successful authentication, timeout, attempt limit exceeded). Because we collected the sensor data during the authentication process alone, we used the entire sample for analyses purposes. The participant was already and remained within the scenario when the authentication process began, meaning, we do not expect outliers in the data from the participants preparing themselves.

Our fifth and final feature was the quality assessment of the 'selfie' image. This information came from an open-source library known as 'FaceQnet' and which uses a Convolutional Neural Network to "predict the suitability of a specific input image for face recognition purposes" [13]. FaceQnet provides a score for an input image between 0 and 1 where 0 means the worst quality, 1 means the best quality. FaceQnet recommends cropping images to the facial region first before

assessing them. By using the open-source Multi-task Cascaded Convolutional Neural Networks (MTCNN) library based on the work provided by Zhang *et al.* [26] we were able to achieve this. In the rare occasion that the MTCNN algorithm was unable to produce a cropped version of the image (usually because the facial region was already over the frames of the images), we used the original un-cropped image instead.

We processed the data in the feature set to allow us to achieve reasonable accuracy. The magnitude ($\sqrt{x^2 + y^2 + z^2}$) of the gyroscope, linear acceleration and magnetometer were calculated for each data point obtained for each authentication attempt. Figure 5 shows a sample of plotted Gyroscope data from one random sitting scenario. For the orientation, we used the median value from our captured data as our feature from each transaction.

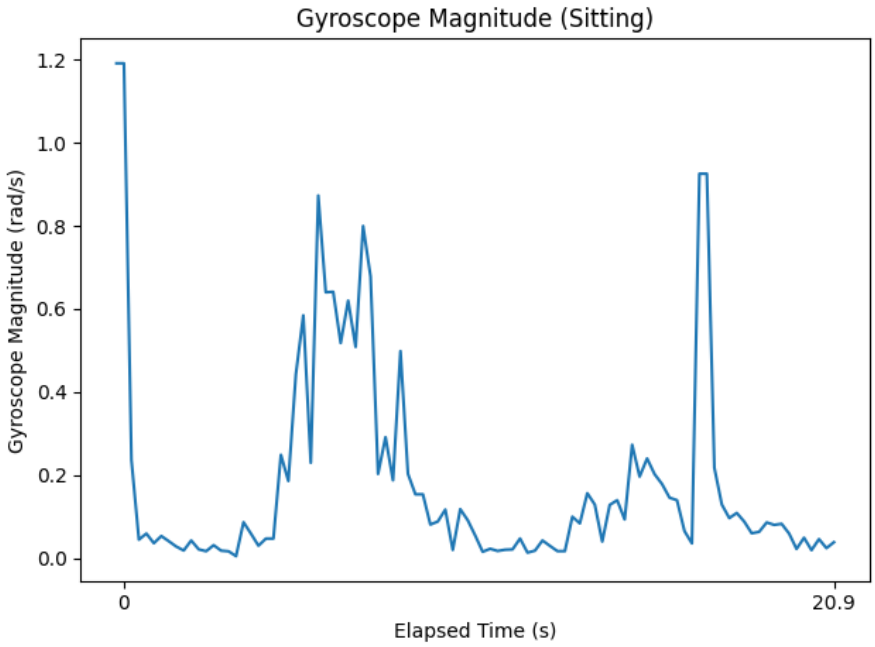


Fig. 5: A sample of gyroscope plot recorded from one transaction during the sitting scenario

We tested standard classifier algorithms (SVM, kNN, Naive Bayes, Decision Tree) to see the impact on the performance. We started from a 'Stationary' and 'Motion' classifier as we believed this would provide the most generic form of scenario categories. We then wanted to create a classifier that could detect the four scenarios that we are interested in ('Sitting', 'Standing', 'Treadmill', 'Corridor'). Finally, we tested a combination of three classifiers; one to categorise 'Stationary' and 'Motion' and another two to classify into the sub-scenarios of each.

At this point, we grouped our features into the individual transactions, and a transaction contains multiple rows of features as the sensors continue to release information. We removed half (50%) of the transactions for training and testing the classifiers. The reason for doing this was to simulated having unseen data for testing the adaptive framework in its entirety later. We repeated this five times, selecting a different 50% each time to see the impact of classification accuracy.

To produce a classifier, we used Python's Scikit Learn library [22]. We split our features into a training (66%) and testing set (33%). We estimated the accuracy using k-fold cross-validation with a fold value of five and reported the F1-score. We found that with our features the k-nearest neighbour algorithm, with a k-value of three, performed the best. Table 4 shows the accuracy results for tested classifiers when classifying our four scenarios.

Classifier	Cross-Val (F1-score)	Training	Testing
Support Vector Machine	0.57 (± 0.03)	0.57	0.57
Decision Tree	0.81 (± 0.04)	0.83	0.83
Random Forest	0.80 (± 0.02)	0.79	0.81
Naive Bayes	0.57 (± 0.09)	0.59	0.60
Quadratic Discriminant	0.58 (± 0.09)	0.60	0.62

Table 4: Classification accuracy for standard classifiers

The kNN classifier with a k-value of three was capable of classifying all four of our scenarios with a testing accuracy of **97%**. Table 5 shows the accuracy results for each of our scenario detection classifiers using kNN for each attempt. The random split of data from attempt five provided the most accurate classifier according to the F1 scores, and this is the one we use for the reaming work in this paper. Table 6 gives the corresponding confusion matrix for the 'Four Scenarios' classifier when testing with the kNN classifier in attempt five. We can bin the vast majority of errors under 'Stationary' and 'Motion' where scenarios within each category are getting misclassified with each other.

Scenario Classifications	Accuracy	1	2	3	4	5
Stationary vs Motion	Cross-Val (F1-score)	0.99 (±0.01)	0.98 (±0.01)	0.98 (±0.01)	0.98 (±0.00)	0.99 (±0.01)
	Training	0.99	1.00	1.00	0.99	1.00
	Testing	0.99	0.99	0.99	0.99	0.99
Four Scenarios	Cross-Val (F1-score)	0.95 (±0.01)	0.96 (±0.01)	0.96 (±0.01)	0.96 (±0.01)	0.97 (±0.01)
	Training	0.98	0.99	0.99	0.98	0.99
	Testing	0.95	0.97	0.97	0.97	0.97
Stationary	Cross-Validation (F1-score)	0.95 (±0.02)	0.96 (±0.03)	0.98 (±0.01)	0.97 (±0.01)	0.98 (±0.01)
	Training	0.98	0.98	0.99	0.99	0.99
	Testing	0.96	0.96	0.98	0.97	0.97
Motion	Cross-Val (F1-score)	0.96 (±0.02)	0.98 (±0.01)	0.97 (±0.01)	0.98 (±0.02)	0.97 (±0.01)
	Training	0.99	1.00	0.99	0.99	0.99
	Testing	0.97	0.99	0.98	0.98	0.99

Table 5: Scenario Classification Results (kNN)

		Predicted			
		Sitting	**Standing**	**Treadmill**	**Corridor**
True	**Sitting**	775	14	5	4
	Standing	24	570	3	4
	Treadmill	1	2	494	11
	Corridor	6	5	3	563

Table 6: 'Four Scenarios' Confusion Matrix

Using the kNN classifier to classify all four of our scenarios provided a testing accuracy of **97%** and **99%** when classifying between stationary and motion scenarios. In all our classifiers we have been able to achieve a testing accuracy of above 90%.

7 Testing the Framework

We tested our framework by using the metadata (features) with our classifier(s) and the 'selfie' image with the 'face-recognition' python library [6,7]. However, in theory, we would like to see the approach incorporated into commercial devices and working in real-time by integrating it into the biometric authentication process. The approach for this would be similar to our off-line approach, the main difference being the real-time collection of the data. The device would collect the sensor information relating to motion and position as the biometric process was happening and continuously turn this information into a feature set similar to ours. This feature set would be continuously passed to a classifier to assign a scenario, whereby a majority vote method, where the operational scenario with the highest number of occurrences, would be used to assign the overall scenario classification. The overall scenario classification will be the one used to assign an adaptive decision threshold. For our prototype, we took the same approach off-line by using our pre-collected data.

7.1 Choosing the Impostors

To assess the effectiveness of the proposed adaptive framework, we need to test the potential to keep out passive impostors and evaluate the false accept rate of the system. For each enrolled participant, we wanted to find the most suitable (tailored) participants to act as impostors. The set theory below represents our algorithm for achieving this.

- Amount of Impostors Required: x
- Current User: c
- Set of Users: U where $c \in U$
- Set of Impostors: $I \subset U = c \notin U$
- An Impostor: i where $i \in I$
- Gender Subset: $G \subseteq I \ \forall \ c.\text{Gender} == i.\text{Gender}$
- Age Group Subset: $A \subseteq I \ \forall \ c.\text{AgeGroup} == i.\text{AgeGroup}$
- Ethnicity Subset: $E \subseteq I \ \forall \ c.\text{Ethnicity} == i.\text{Ethnicity}$
- Nationality Subset: $N \subseteq I \ \forall \ c.\text{Nationality} == i.\text{Nationality}$
- Subset of Tailored Impostors: $T = G \cap A \cap E \cap N \subseteq I$
- if $|T| >= x$ {Randomly select x elements from set}
- while $|T| < x$
  - Randomly select from $G \cap A \cap E \cap N$ until $|T| == x$ is reached
  - if $G \cap A \cap E \cap N$ becomes $\emptyset$ {Randomly select from $G \cap A \cap E$ until $|T| == x$ is reached}
  - if $G \cap A \cap E$ becomes $\emptyset$ {Randomly select from $G \cap A$ until $|T| == x$ is reached}
  - if $G \cap A$ becomes $\emptyset$ {Randomly select from G until $|T| == x$ is reached}

For our purposes, we define our 'AgeGroup' as the ranges specified in Table 2 and our 'Ethnicity' under the five broad ethnic groupings specified by the UK Government [11]. This algorithm should result in a set of x tailored impostors for each participant who most resembles that of the participant. We expected our impostor set to provide the most likely cases to cause a false accept to occur. We experimented adjusting the number of tailored impostors to provide meaningful results because when using our algorithm, the more impostors we add, the less tailored they will be resulting in dilution of the results. For our data, we found using a total of four impostors per genuine user (2015 impostor comparisons) seemed to provide a fair balance before our impostors became less tailored. We discuss this more in Section 8 and Figure 6.

7.2 Examining the Threshold

The recommended threshold from the python 'face-recognition' library [6,7] is 0.6. When using our data, this gives us a false reject rate of 0.00%, a false accept rate of 10.22% and an equal error rate of approximately 0.64%. It seems the library is recommending a practical threshold value for

the majority of cases. To test our adaptive theory, we would like to devise a scenario whereby security is of great concern. Therefore, we require a low (<1%) false accept rate by setting tighter, more restrictive thresholds.

We identified in Section 4 that the match score varies across scenarios and that we should be setting other thresholds for each. We took several approaches to set appropriate threshold values, and in our case, we wanted to consider trialling multiple thresholds for our scenarios. The trials allowed us to see how varying thresholds could affect overall system performance. For example, we could use the maximum distance score obtained from our data. We experimented with using the 95^{th} percentile, maximum distance, and the EER threshold value from our scenario data as the threshold values. Our theory is that this will allow for the majority of genuine cases without causing extremes and outliers in our data to be accepted.

Similarly to how we handled the creation of the scenario classifier, we used a random 75% sample from our dissimilarity score data (75% from genuine and 75% from impostors) to create the thresholds. The impostor scores used for this were the ones created using our tailored impostors. We repeated this five times, picking a new random set each time to see the impact as shown in Table 7.

8 Results

Bringing the framework together, we can use the classifiers produced as discussed in Section 6, along with the thresholds found in Section 7.2 and chosen tailored impostors as based on Section 7.1. Using the open-source 'face-recognition' library [6,7] and our pre-collected data, we can examine how our adaptive framework could perform.

Figure 6 shows how the false accept rate changes as we used our algorithm to alter the number of impostors used (the algorithm was rerun for each iteration). We randomly selected a total of 450 comparisons, to provide a reasonable sample, from the impostor comparisons pool with x number of impostors per genuine user, we repeated this three times and took an average to produce the graph. We can see that the baseline's FAR declines as the impostors become less tailored; however, our adaptive approach outperforms the baseline with the most tailored impostors and continues to do so even when we include less tailored impostors.

We tested our more generic classifier that can classify the authentication metadata into a 'Stationary' and 'Motion' category. We followed this with a test of the classier that could distinguish between the four scenarios that we were experimenting with: 'Sitting', 'Standing', 'Treadmill', 'Corridor'. Finally, we trialled a combination of the two classifiers, where the data would first classify into 'Stationary' and 'Motion' and then into separate classifiers for the scenario that belonged to either category. Trialling both using '95^{th}', 'Max' and 'EER' thresholds, we can achieve recognition results as shown in Table 7.

We know previously from our classifier accuracy that we are not classifying all the scenarios correctly every time. Meaning there is a risk of an incorrect classification to a scenario that has an alternative acceptance threshold. This misclassification poses a risk for impostors to be accepted by the system. Further work to improve the classifier accuracy will result in improved recognition performance.

Our results show that using an adaptive approach can be capable of producing reliable recognition accuracy, particularly with maintaining and improving a low false accept rate above a traditional fixed value. Table 8 highlights this comparison when using our four scenario classifier and EER (number 3 in Table 7) thresholds, to the baseline and a perfect classifier. A perfect classifier would be able to accurately categories our scenarios 100% of the time. Our most significant success in using our adaptive approach has been in reducing the false accept rate by approximately 95% from baseline performance.

8.1 Verification

Having had success testing our scenario adaptive threshold method on the Samsung Galaxy S9, we wanted to test the same concept on another device to see if the approach was interoperable.

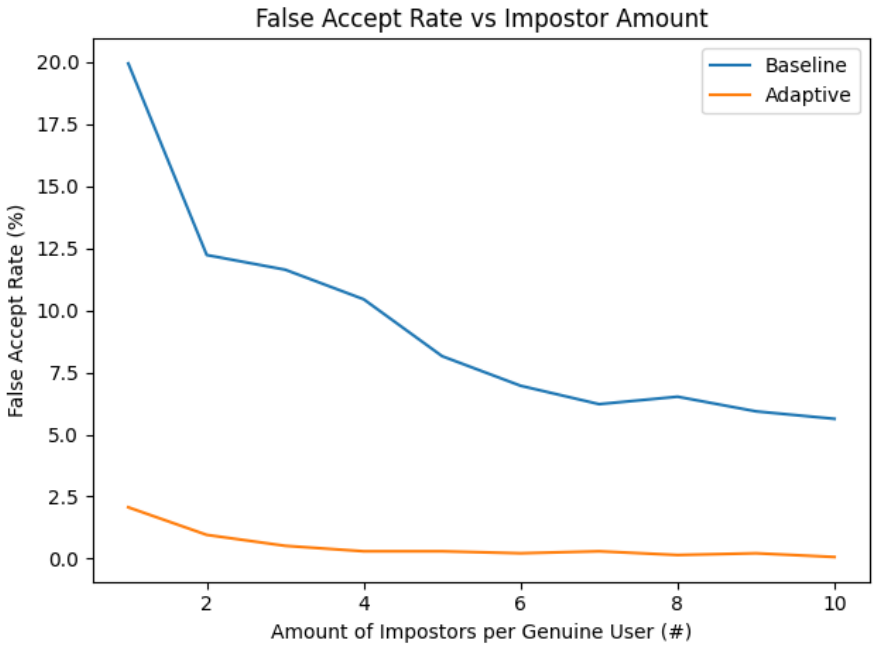


Fig. 6: Changes to false accept rate with varying impostor amounts

Classifier	Threshold	1	2	3	4	5
Stationary vs Motion	95^{th}	FRR: 5.29 FAR: 0.00	FRR: 4.65 FAR: 0.00	FRR: 6.98 FAR: 0.00	FRR: 5.07 FAR: 0.00	FRR: 6.13 FAR: 0.00
	Max	FRR: 0.00 FAR: 2.53	FRR: 0.00 FAR: 2.53	FRR: 0.00 FAR: 2.53	FRR: 0.00 FAR: 2.53	FRR: 0.63 FAR: 0.35
	EER	FRR: 0.42 FAR: 0.35	FRR: 0.42 FAR: 0.69	FRR: 0.42 FAR: 0.50	FRR: 0.42 FAR: 0.50	FRR: 0.85 FAR: 0.25
Four Scenarios	95^{th}	FRR: 6.13 FAR: 0.05	FRR: 7.19 FAR: 0.10	FRR: 7.40 FAR: 0.00	FRR: 5.92 FAR: 0.10	FRR: 6.98 FAR: 0.00
	Max	FRR: 0.85 FAR: 1.39	FRR: 1.06 FAR: 1.39	FRR: 0.00 FAR: 2.53	FRR: 0.00 FAR: 2.53	FRR: 1.48 FAR: 0.25
	EER	FRR: 0.85 FAR: 1.39	FRR: 1.27 FAR: 0.89	FRR: 1.06 FAR: 0.50	FRR: 1.06 FAR: 0.55	FRR: 1.48 FAR: 0.25
Stationary/Motion + Scenarios	95^{th}	FRR: 6.13 FAR: 0.05	FRR: 7.19 FAR: 0.10	FRR: 7.40 FAR: 0.00	FRR: 5.92 FAR: 0.10	FRR: 6.98 FAR: 0.00
	Max	FRR: 0.85 FAR: 1.39	FRR: 1.06 FAR: 1.39	FRR: 0.85 FAR: 1.39	FRR: 0.00 FAR: 2.53	FRR: 1.48 FAR: 0.25
	EER	FRR: 1.06 FAR: 0.55	FRR: 1.27 FAR: 0.89	FRR: 1.06 FAR: 0.50	FRR: 1.06 FAR: 0.55	FRR: 1.48 FAR: 0.25

Table 7: Recognition performance results when trialing the adaptive threshold

	Recognition Performance
Baseline	FRR: 0.00 FAR: 10.22
Adaptive Threshold	FRR: 1.06 FAR: 0.50
Perfect Scenario Classifier	FRR: 0.42 FAR: 0.60

Table 8: Comparing recommended baseline performance to our adaptive approach

We experimented with another Android-based device the Google Pixel 2. However, Google Pixel 2 does not allow developer access to its 'Trusted Face' feature meaning that we cannot collect background sensor data during the authentication process. To counter for this, we had the device collect the sensor features while the participant was operating the in-built device's camera and taking a 'selfie' in an attempt to simulate the authentication process. The side effect of this means there was a lot more sensor data collected as operating and using the camera on average takes more time than the usual biometric authentication prompt to complete.

We collected the data under the same scenario conditions. We had an additional 100 genuine 'sitting' transactions, 116 'standing' transactions, 124 'treadmill' and 141 'corridor' from around 30 different individuals of a similar student demographic which operated the Samsung Galaxy S9. When running the 'selfie' data collected from the Google Pixel 2 through the 'face-recognition' Python library with baseline threshold (0.6), we receive the performance results of FRR: 0.00% and FAR: 9.50%.

We took the same approach as before by removing half of the transactions before attempting to classify. As devices are unique with different sensors, we cannot rely on using the same classifier as before, and unique ones will need producing for each device/model. The classifier evaluation results were promising with the 'four scenarios' classifier reporting results of cross-validation accuracy being 1.00 ($\pm$0.00) along with both the training and testing accuracy being 1.00%. We also use 75% of the dissimilarity score data to set appropriate thresholds. As a trial, we generated three sets of EER thresholds by altering the 75% of the data used. For the three trials, our results were again showing improvements over the baseline case and beginning to prove that the adaptive threshold is interoperable:

- FRR: 1.25% and FAR: 0.00%
- FRR: 2.91% and FAR: 0.00%
- FRR: 1.66% and FAR: 0.00%

9 Conclusion

In this paper, we presented a novel adaptive approach to biometric authentication for a mobile device, an area of research currently lacking in the literature as noted by Pisani *et al.* [23]. We proposed the creation of an extendable 'Adaptive Framework', whereby we set a unique threshold value for specified scenarios. The theoretical advantage to this approach is to allow for stricter control over access, by not having to specify a one-off static threshold value to account for the vast amount of conditions where a biometric authentication may occur.

Our approach utilised the sensors readily available on the vast majority of modern smartphones (and wearables) with developer access. It showed the transformation into potential features for building a classifier that could recognise simple scenario categories. Our classifier for detecting our four simple scenarios had a testing accuracy of **97%**. The framework relies on having the ability to identify the scenario reliably, and our results suggest that this ability is a significant factor in the overall function of the adaptive framework to perform optimally. The paper focuses on using an adaptive approach for face recognition, but we see no significant obstacles for using the same

technique for other physical and behavioural biometric modalities. We intend to continue this work into mobile scenario detection to investigate the impact features have on classification accuracy.

We demonstrated using collected data from a commercial device, and an open-source face recognition algorithm that this method has potential merit. By imagining a scenario where security and privacy are of grave concern, and hence a low false accept rate may be considered more important than the false reject rate, we tested our method against a static, fixed threshold. With this in mind, we produced an algorithm to help us identify the best impostors to use for each participant to help stress-test the approach. We demonstrated the impact of tailoring in Figure 6, which proved that our algorithm was working.

We then performed off-line testing using Python's 'face recognition' library [6,7], which recommends a threshold of around 0.6. We saw when using our collected data; this threshold gave a false reject rate of 0.00% and a false accept rate of 10.22%. By taking our adaptive approach, we found that one of our best methods was using the classifier to detect between stationary and motion scenarios along with the EER threshold value. In doing so, we were able to achieve a result that gave a false reject rate of **0.42%** and a false accept rate of **0.35%**, a reduction of 95% over the algorithm's baseline threshold. We also showed the interoperability of the approach by replicating it using another device with similarly successful results. Our relatively simple adaptive method was able to produce an improvement on recognition performance, which could outperform an algorithm using a single static threshold value.

We acknowledge that this has been one relatively simple example to demonstrate the practicalities and proof-of-concept of using this adaptive approach. Further testing will be required to prove the competency of the method thoroughly, including a greater variety of scenarios and environmental lighting and weather conditions. Testing should include an approach to adapt presentation attack detection (PAD) methods for individual scenarios and mitigate malicious actors in exploiting weaknesses in the adaptive approach. We hope that others will take the work we have started to produce and further investigate the effectiveness of the method. As well as allow developers and manufacturers to incorporate a scenario-based threshold adaptive approach into future algorithms in mobile biometric systems, to allow for higher security without jeopardising performance.

References

1. BBC News: Facial recognition: Eu considers ban of up to five years. [Online] https://www.bbc.co.uk/news/technology-51148501 (Jan 2020), https://www.bbc.co.uk/news/technology-51148501, [Accessed: 29/03/20]
2. Boakes, M., Guest, R., Deravi, F., Corsetti, B.: Exploring mobile biometric performance through identification of core factors and relationships. IEEE Transactions on Biometrics, Behavior, and Identity Science **1**(4), 278–291 (2019)
3. Brumback, C.B., Knight, D.W., Messenger, J.D.M., Hong, J.O.: Biometric sensing device having adaptive data threshold, a performance goal, and a goal celebration display (May 27 2014), uS Patent 8,734,296
4. Castilllo-Guerra, E., Diaz-Amador, R., Julian, C.B.L.: Adaptive threshold estimation for speaker verification systems. Journal of the Acoustical Society of America **123**(5), 3877 (2008)
5. Chen, S., Pande, A., Mohapatra, P.: Sensor-assisted facial recognition: an enhanced biometric authentication system for smartphones. In: Proceedings of the 12th annual international conference on Mobile systems, applications, and services. pp. 109–122 (2014)
6. Geitgey, A.: face-recognition. [Online] https://pypi.org/project/face-recognition/ (Mar 2013), https://pypi.org/project/face-recognition/, [Accessed 29/03/20]]
7. Geitgey, A.: Machine learning is fun! part 4: Modern face recognition with deep learning. [Online] https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78 (Jul 2016), https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78, [Accessed 29/03/20]]
8. Goode, A.: Digital identity: solving the problem of trust. Biometric Technology Today **2019**(10), 5–8 (2019)

9. Google: SensorManager — Android Developers (2016), https://developer.android.com/reference/android/hardware/SensorManager, [Accessed: 16/10/20]
10. Google: Show a biometric authentication dialog — Android Developers (2020), https://developer.android.com/training/sign-in/biometric-auth, [Accessed: 16/10/20]
11. GOV.UK: List of ethnic groups. [Online] https://www.ethnicity-facts-figures.service.gov.uk/ethnic-groups, https://www.ethnicity-facts-figures.service.gov.uk/ethnic-groups, [Accessed: 04/06/20]
12. Gutta, S., Trajkovic, M., Philomin, V.: System and method for adaptively setting biometric measurement thresholds (Jan 4 2007), uS Patent App. 10/574,138
13. Hernandez-Ortega, J., Galbally, J., Fierrez, J., Haraksim, R., Beslay, L.: Faceqnet: Quality assessment for face recognition based on deep learning. arXiv preprint arXiv:1904.01740 (2019)
14. ISO: Text of standing document 11 (sd 11), part 1 overview standards harmonization document. Standard, International Organization for Standardization (Aug 2010)
15. Kumar, R., Phoha, V.V., Serwadda, A.: Continuous authentication of smartphone users by fusing typing, swiping, and phone movement patterns. In: 2016 IEEE 8th international conference on biometrics theory, applications and systems (BTAS). pp. 1–8. IEEE (2016)
16. Lee, D.: San francisco is first us city to ban facial recognition. [Online] https://www.bbc.co.uk/news/technology-48276660 (May 2019), https://www.bbc.co.uk/news/technology-48276660, [Accessed: 29/03/20]
17. Li, Y., Li, Y., Yan, Q., Kong, H., Deng, R.H.: Seeing your face is not enough: An inertial sensor-based liveness detection for face authentication. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. pp. 1558–1569 (2015)
18. Lunerti, C., Guest, R., Baker, J., Fernandez-Lopez, P., Sanchez-Reillo, R.: Sensing movement on smartphone devices to assess user interaction for face verification. In: 2018 International Carnahan Conference on Security Technology (ICCST). pp. 1–5. IEEE (2018)
19. Mansfield, A.J., Wayman, J.L.: Best practices in testing and reporting performance of biometric devices. Centre for Mathematics and Scientific Computing, National Physical Laboratory (2002)
20. Mhenni, A., Cherrier, E., Rosenberger, C., Amara, N.E.B.: Adaptive biometric strategy using doddington zoo classification of user's keystroke dynamics. In: 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC). pp. 488–493. IEEE (2018)
21. Patel, V.M., Chellappa, R., Chandra, D., Barbello, B.: Continuous user authentication on mobile devices: Recent progress and remaining challenges. IEEE Signal Processing Magazine **33**(4), 49–61 (2016)
22. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. Journal of machine learning research **12**(Oct), 2825–2830 (2011)
23. Pisani, P.H., Mhenni, A., Giot, R., Cherrier, E., Poh, N., Ferreira de Carvalho, A.é.C.P.d.L., Rosenberger, C., Amara, N.E.B.: Adaptive biometric systems: Review and perspectives. ACM Computing Surveys (CSUR) **52**(5), 1–38 (2019)
24. Poh, N., Wong, R., Kittler, J., Roli, F.: Challenges and research directions for adaptive biometric recognition systems. In: International Conference on Biometrics. pp. 753–764. Springer (2009)
25. Vasiete, E., Chen, Y., Char, I., Yeh, T., Patel, V., Davis, L., Chellappa, R.: Toward a non-intrusive, physio-behavioral biometric for smartphones. In: Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services. pp. 501–506 (2014)
26. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)