An Algorithmic Approach for Optimising Biometric Systems Using Liveness and Coercion Detection

Peter Matthew, Susan Canning

Department of Computer Science, Edge Hill University, Ormskirk, Lancashire

Abstract

This paper looks at the possibility of creating an algorithm that will combine liveness and coercion modalities, along with organisational factors such as workforce composition. This algorithm will produce a security value, which can then be compared with other combinations of modalities, therefore, providing a way to test potential security setups without having to tangibly implement them. An experimental methodology has been used, focusing on the development of the algorithm, and the associated effects of it. The hypothesis is that the algorithm can produce a value that can further be used to compare different setups of biometric security. This comparative value can be used to discover the best combinations of modalities for fusion development or practical installation. After testing, the algorithm is proved to work as it creates an appropriate value, called the security value, which can then be compared with other setups to find the most suitable for a given situation. There are some issues with this primarily due to data provision, the requirements for more data to parse through the algorithm, and finally, the need for a suitable interface, otherwise it may be too complex for efficient usage in a traditional security environment. This algorithm provides a specific security value that can be applied to a variety of situations, for example, there are potential implications within a general security application; such as liveness and coercion multimodal fusion, autonomous system development and pervasive environments; such as allowing dynamic security systems to be developed. However the main focus of this algorithm is to highlight the fusion of liveness and coercion detection, and how they can be best applied to specific security scenarios.

Keywords: Biometric Security, Liveness, Coercion, Autonomic Computing, Self-optimisation, Fusion, Personalised Security.

1. Introduction

When considering biometric security, the most appropriate device and liveness detection techniques are of the utmost importance. These can often be heavily influenced by a variety of developer preferences such as ease of implementation; cost constraints; device suitability, subsequently leading to the potential integration of less than optimal techniques and components for a particular system, or for the system's workforce. This process is becoming increasingly complex as more security techniques are developed making successful fusion even harder to achieve effectively. When combining multiple technologies, sample acquisition and liveness, it is imperative that the method of combination is efficient and effective. This is also very important when working towards the future-proofing of new techniques of sample gathering, liveness detection but also new security approaches. This fusion of techniques and devices needs to be considered throughout the development process. At what point in the architectural journey will fusion be the most efficient and effective, either for each area individually, or the system as a whole? Research into device and liveness fusion is extensive as highlighted by [1] and [2], and has become a standard aspect of biometric security detection and monitoring. Whilst presentation attacks are the most utilised threat vector, with liveness being designed to combat this threat, there are other threat vectors being found all the time. In this paper, one such vector is proposed which starts at the presentation layer, and coercion detection; this checks to see if a user is begin coerced into authenticating. When trying to detect and deal with this, additional complexity will be added to the system, as well as more demands placed on the important metrics such as performance and time.

This paper presents a technique to incorporate coercion detection standards into current or future, biometric security system. To do this, a theoretical framework for measuring the most efficient liveness and coercion detection techniques in a given scenario will be discussed, placing a higher emphasis on efficiency and speed of completion. While the current research on coercion detection is quite limited, [3][4][5], however, this has been partially explored by [6]. This paper will focus on real-time liveness and coercion techniques that have been chosen due to their accuracy and universality. Furthermore, this framework will help identify the best methods of technique integration from a device and technique level. This is a novel approach as currently the inclusion of coercion techniques are not considered the norm for biometric security, and there are no current projects focusing on this area as far as the authors can find. However, there are numerous scenarios when the coercion is being perpetrated by law enforcement, and there is currently an extensive debate on the validity of this in numerous countries for example [7]. However, this is not the area this paper is focusing on. This paper focuses on an individual being coerced by a nefarious attacker, instead of a law enforcement representative.

Traditionally, data simulations of different techniques would be considered, or practical testing would take place. In this scenario, a small scale dataset is being used that corresponds to a higher education institution. This data is easily accessed via opensource repositories and allows for effective comparison of different devices and security factors such as biometric device and liveness/coercion standards. This would subsequently allow security developers to identify viable combinations for system implementations, as well as test combinations that might be otherwise difficult to model due to a particular workforce, or location noise deviation factors, for example, high levels of specific disabilities, or areas of extreme thermal divergence.

There are two main goals within this paper: firstly to measure the reliability of a specific system configuration, making use of biometric devices, liveness and coercion standards; secondly allowing the optimisation of the said system, either manually or autonomically to better suit the scenario.

To achieve this, the paper will discuss the creation of a security variable, which can be used to compare to other permutations within the model and find the most effective technique for a specific scenario, effectively addressing the first aim. This is done by developing an algorithm that will take into consideration all of the associated factors within a system and produce an output that can be used by the system developer. However, it will not consider the development of additional coercion techniques, instead, the ones highlighted in [6], such as tangible key techniques and facial micromovements,will be used. Data will be simulated through the model and will output an appropriate "score" which can then be compared to different technique combinations. This value can then be used to optimise said systems either manually or autonomically. Section 1 will introduce the paper; Section 2 will discuss coercion and liveness detection, and the impact on secure systems; Section 3 will detail the justification of development styles; Section 4 will discuss the testing factors used and consider the overall performance of the algorithm and finally; Section 5 will conclude the paper.

2. Coercion and Liveness Detection

Due to the ease in which some biometric techniques can be spoofed, the application of liveness standards within a device is no longer an afterthought, instead it is an integral requirement. This can be seen over many research projects spanning the last fifteen plus years such as [8] to [9]. The current application of liveness differs dramatically depending on the intra-fusion requirements and can be seen in current literature such as [10] [11] [12]. It is an excepted factor of biometric security that liveness needs to be included if security is the main requirement of the installation, otherwise, there runs the risk of compromising the system. Areas where this is not applied is the application of biometric techniques within smart devices, which rarely have robust liveness standards [13]. However, this concept is being challenged especially with events such as MoBio LivDet [14]. Whilst the focus has been firmly on liveness detection, this does not mean other threats should be ignored. In previous papers [6] coercion detection has been highlighted as a potential area of interest. This concept has been discussed briefly across a number of other areas such as: highlighting the impact it could have [15]; the impacts of coercion on responsibility shifting attacks [5]; using calming music to reduce neurophysiological responses during coercion [16]; despite this, there are biometric implications of coercion which have not been researched thoroughly. As with liveness detection, it will take some time to thoroughly integrate into the overall biometric process, however, the importance of this integration is plain to see across the biometric research spectrum [17].

The use of coercion within biometrics is not new, in fact, it has been discussed in a number of areas, however, this is normally only regarding the forced provision of personal samples - such as taking fingerprints by force and so on [18] [19] [20]. When considering coercion the following definition is used *The action or practise of persuading someone to do something by using force or threats.* [21]. However, there is very little information dealing with individuals forcing users to authenticate into a system, subsequently letting the attacker access restricted physical or digital areas. The few researchers looking into this area focus on countermeasures such as [16] and [5], highlighting that this is a potential threat area. This paper focuses on the integration of coercion techniques within a secure system, therefore providing the installation with techniques of detection and prevention, as well as how to check for suitability and effectiveness within a system by using the developed algorithm.

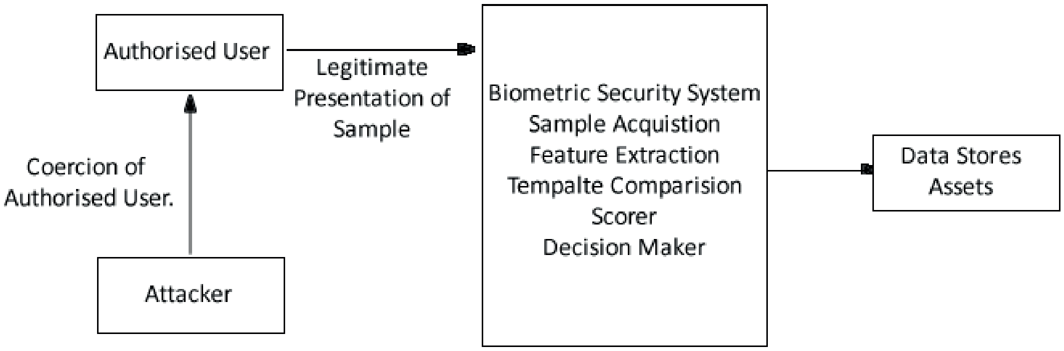

Figure 1: Coercion Attack Model.

2.1. Liveness Detection

Liveness detection modalities are included to address the innate flaws with biometric sample collection provision, specifically, the impact of latent sample collection, therefore reducing the impact of sample spoofing, which became a growing problem in the 2000s as highlighted by [22] and [23]. At first, these techniques were completely independent of the sample used and required additional progression steps. Now many different liveness detection techniques make use of current technologies, such as galvanic skin response GSR (now know as skin conductivity response SCR), respiration, and heart rate [24] and [25]. Therefore, the following section will endeavour to address the algorithmic development process that allows the framework to function. This will then be tested using data gathered from a standard organisation structure, as well as some government based statistics.

Numerous liveness techniques are used for different biometric modalities, and more are being developed all the time as can be seen in the annual LivDet challenge [26]. While there is a wide range of liveness modalities that can be accessed for different biometric samples, in this work, we will focus on fingerprint and facial recognition samples. This is due to their ubiquity with security environments [17], acceptance of use [27], and applicability through coercion modalities [28].

When considering facial recognition, numerous techniques can be accessed, and although most Liveness techniques are based on texture factorsother factors can be considered such as life sign measures and deep feature analysis. When considering texture analysis, this is normally achieved with either traditional Local Binary Patterns (LBP), or by using advanced LBP. Using LBP involves gathering several parameters by splitting the sample image into several local regions. Here the LBP descriptors are extracted from each region independently, which are then combined to create one overall face description. When considering the application within liveness detection, the same process is followed however instead of a full face image, the texture within the different regions is considered, as 3d textures are required. This is something that a printed image should not be able to replicate unless using three-dimensional techniques such as 3d printers. Histograms of each region are then gathered and concatenated together, providing an overall sample view. These histograms allow a comparison to be conducted, and this comparison is achieved by using techniques such as euclidean distance, chi-square and absolute values. This process is highlighted within in Figure 2 [29]. Although this is a common technique, there are some very problematic factors. The main one is that the quality of the image is paramount, and there can be a massive drop in performance when the image is of low quality. The drop in performance is because the lower the quality of image, the harder it is to collect valid region data that translates into usable histogram data subsequently minimising the impact of the measure algorithm [30].

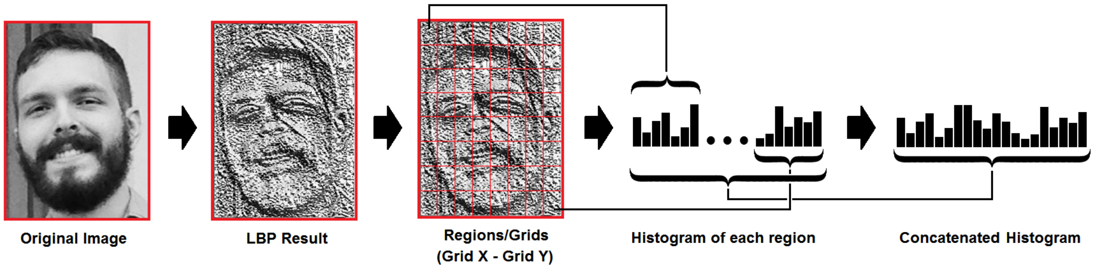

Figure 2: Coercion Attack Techniques. [29]

Alternative techniques include the use of life sign factors such as blood flow, voluntary/involuntary signals and facial micromovement. These are the techniques that have the most in common with coercion detection, as they can be affected by a multitude of emotional responses [31]. Microexpressions are very effective as they not intentionally created, and there do not produce false information that can be caused with specific manipulation techniques such as planning and timing. The process of gathering the liveness data is the same as a standard microexpression, and the most common processes make use of Active Shape Model (ASM) [32] [33], Discriminative Response Maps Fitting (DRMF) [34], Subspace Constrained Mean-Shifts (SCMS) [35], Face++ automatic facial point detector [36], and Constraint Local Model (CLM)[37].

Fingerprint modalities likewise have numerous techniques which are split between software and hardware. Hardware techniques make use of additional information such as blood flow [38], odor [39], and salinity [40]. The primary issues with these techniques are the requirement of additional hardware for them to function. This can be costly, and can add another layer of interaction in front of the user, a negative factor if attempting to accomplish a transparent process.

The alternative approach is to make use of software techniques that are focusing on data that is provided during the sample collection process and does not require any additional hardware to gather information. Techniques are usually highlighted as belonging to one of twoclasses, feature-based, and deep learning. The use of techniques such as LBP to gather texture details is a standard modality, however other techniques involve the inclusion of deep machine learning [41], and convolutional networks to detect liveness [42].

2.2. Coercion Detection

Traditional biometric architecture is well documented and has been discussed in many sources such as [17]. When considering how this is applied in coercion detection, Figure: 1 shows the workflow. Firstly the attacker finds an appropriate user then forces them, through some manner of physical or mental threatening behaviour, to authenticate into the system. When this happens, the user authenticates as they are legitimately allowed into the system. The attacker then takes control of the device/place and continues their nefarious activities. From a system standpoint, there is no security breach as the user has access rights to the system. Currently, there is no way to check for this coercion in standard biometric devices; however, in a previous paper,

[6] some techniques for coercion detection are proposed. Some of these techniques can be seen in Figure: 3 some of which are already used within other areas of biometric security, e.g. such as skin conductance response (SCR) tests, used within liveness detection modalities [43] while some are original techniques for coercion such as intentional false authentication [6]. Figure 3 highlights some coercion techniques, theses are not being discussed in detail in this work, for further information see [6].

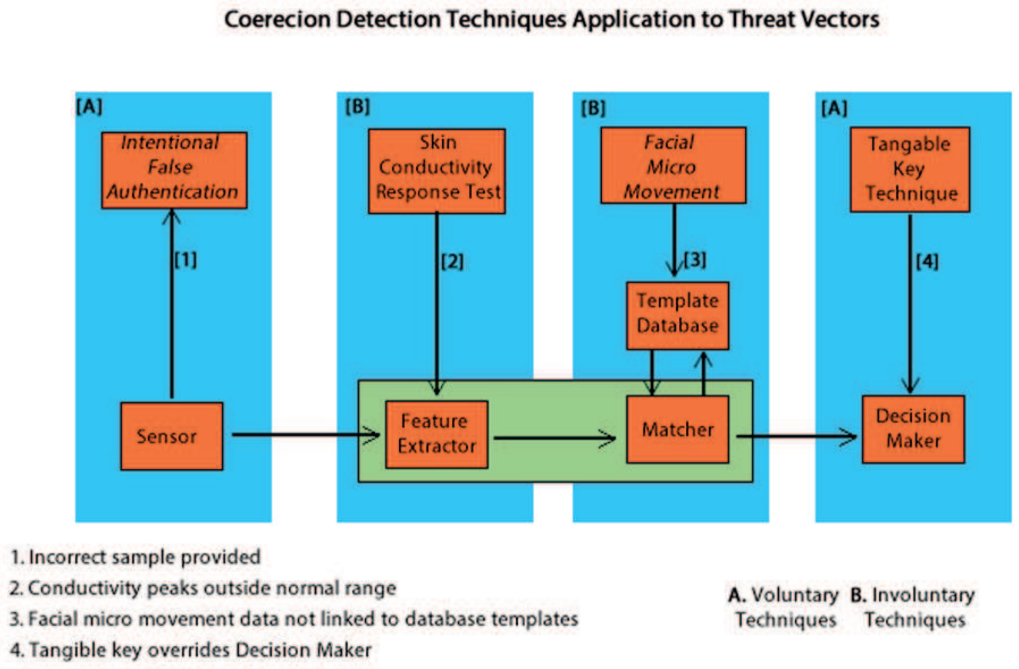
Coerecion Detection Techniques Application to Threat Vectors
[A]
Intentional False Authentication
[1]
Sensor
[B]
Skin Conductivity Response Test
[2]
Feature Extractor
[B]
Facial Micro Movement
[3]
Template Database
Matcher
[A]
Tangable Key Technique
[4]
Decision Maker
1. Incorrect sample provided
2. Conductivity peaks outside normal range
3. Facial micro movement data not linked to database templates
4. Tangible key overrides Decision Maker
A. Voluntary Techniques
B. Involuntary Techniques

Figure 3: Coercion Attack Example Techniques.

As Figure 3 shows there are three types of coercion modalities which are involuntary, voluntary and environmental. Involuntary modalities are primarily focused on physiological signals, which have the potential to be useful as they require the least conscious effort from the user and therefore are not susceptible to users getting things incorrect. There are many potential variances when gathering involuntary modalities such as facial micromovements [44], as well as other medical modalities such as heart rate [45], perspiration

[46], and so on. This is due to the transient nature of medical factors and the plethora of factors that can change the samples. [4][47], all of which can occur outside of a coercion scenario. The second concept would be a voluntary approach which requires the user to provide some additional information. This is usually done with non-medical data, for example, speaking a password or selecting a pattern. There are a variety of forms this could take. For example, the user may carry a key which may be used as a 'panic alarm'. This is standard practice in areas such as health and elderly care; therefore, the concept is already disseminated throughout society [48]. A second approach would be to utilise a selection of passcodes/keys that would denote coercion. Although to be thoroughly effective, the technique would have to be completely integrated within the full biometric system; otherwise, the use of the traditional key system would be seen as a simple and more straightforward approach to security. Without the thorough integration, it would also become evident to the attacker that some preventative measure is being taken, as the inclusion of a key in the final stage would become suspicious for a nefarious user, and therefore reduce the effectiveness of the technique. The final modality is via environmental data. Instead of utilising data provided by the user, either voluntarily or involuntary, data gathered from the surrounding area is used. Unfortunately, this technique can be problematic to implement as it correlates data with the user being used as a focal point and will have to search for multiple samples which can make some sample collection modalities become less effective [13]. For example, the use of cameras and proximity maps can depict if there are people close together. While on its own, this is both irrelevant, and can easily be explained away with a handful of reasons; user's are huddling from weather, close relationships and so on. However, when combined with additive security protocols, the development can become more impressive. For example, all users must make sure they are on their own when authenticating into a system, no more than one user with a biometric scanner at any one time etc. Therefore, if a proximity sensor can detect multiple people closer together, then it can indicate an attack of some kind [49]. There is a host of potential problems with this form of approach, least of all the ease of misunderstanding. Using the above example, if two users were carrying a heavy parcel, they would be identified as too close, and therefore the system would respond, in this case, erroneously. There are also other noise-based concerns, such as using thermal sensors/imaging and being able to differentiate between humans and other species such as cats and dogs.

3. Algorithm Development

The algorithm has been developed using a hybrid style: brute force alongside a divide and conquer technique. Initially, a brute force base is being used as it is more easily combined with other techniques. This will improve efficiency as the implementation of this algorithm will use a variety of different calculations and disciplines, therefore making a brute force technique efficient and effective [50]. Secondly, as the overall algorithmic complexity is not high, especially from a mathematical standpoint, the comparative simplicity of the brute force technique will be ideally suited to this development style [51]. The second algorithmic technique will be divide and conquer due to the range of components and the number of sub-algorithms being used that will calculate specific factors and these will make up the overall main algorithm [52]. This combination, theoretically, negates some of the main disadvantages associated with the other algorithmic development types, for example, brute force does not use shortcuts and can be quite inefficient, while divide and conquer's primary advantage is the efficiency it brings to the development process, subsequently minimising brute force's detrimental issues [53]. While other techniques, such as the decrease and conquer and transform and conquer, could potentially be viable, the focus on simplifying the problem to get a simpler result is not the objective in this research [54]. There are the two main goals within this paper: firstly measure the reliability as a specific system configuration, making use of biometric devices, liveness and coercion standards; secondly allowing the optimisation of the said system, either manually or autonomically to better suit the scenario. To do this an algorithm has been created that can produce a single output that denotes total system viability for different security techniques. This does not mean improvement is something to avoid, however it is not covered in the scope of this article. It will contain factors that will be specific to the installation environment and will not change unless different environments are considered i.e. different location, company, etc. The associated coercion and liveness standards will change the overall algorithm, therefore allowing different combinations of techniques to be tested and this will help identify suitable multi-layer fusion which will, in turn, denote the statistical security level within the installation. However, in the future adding other algorithmic techniques to improve performance and efficiency is something that would be considered.

By adding new security systems, component, etc. into a system, there is the requirement that it solves a particular problem, and its successes depend on the development of either a reliability value or a comparative optimisation metric [54]. Therefore, within this scenario, there is an inbuilt testing metric which, if satisfied, indicates a success. Therefore the success of this technique involves the development of a usable security level that is scalable and flexible allowing developers to make informed choices for a better optimisation security application. This algorithm will take liveness and coercion techniques, alongside a variety of other metrics, and calculate the overall security rating an installation has. At first this metric is abstract, however, it can be used to compare different installation setups, when different scenarios and combinations of techniques have been run through the framework. This is useful as biometric security systems can have dramatically different requirements going from scenario to scenario. The metrics can change due to a variety of factors such as workforce composition to device usage. Therefore having a straight forward tool that can be easily changed and expanded on, with a variety of metrics and flexibility, can make the development of system securities much more simple.

When considering the different metrics and techniques of application, the initial algorithm was overly complex as seen within Equation 1, therefore the first factor to consider is to eliminate errors and to make sure the algorithm was as efficient and effective as possible.

Table 1 highlights the main components of the final algorithm as shown in Equation: 13. These factors have been gathered due to their importance within a security system combined with the unique factors central to this research, such as the coercion techniques, and the device redundancies.

$$\sum_{i=0}^{n} y_i - \overline{y}^2 = A_s = \frac{T}{P/A_b} * D_r = \frac{A_s}{L_t + L_c} \tag{1}$$

As shown in Table: 1, A_s denotes the final score; T denotes execution time; P denotes participants; A_b denotes anomalous users; D_r denotes device redundancy; L_t and L_c denote liveness and coercion techniques respectively.

Each of these metrics requires its own calculation and development, however, due to the complexity of some areas further calculations were needed. For example initially, the intent was to measure when a user logged into a system. However, this did not take into account the potential for multiple authentication attempts that a single user could conduct, therefore some form of differentiation regarding this scale was needed. To better facilitate this, as

Component	Notation	Considerations
Autonomic Score	As	Overall value - Higher = More secure
Time	T	Time taken to authenticate.
Participants	P	Amount of users
Anomalous User-base	Ab	Permanence and collectability of samples
Device Redundancy	Dr	Combined level of Liveness/Coercion Techniques

Table 1: Final Algorithm Components.

well as to minimise the complexity of the other metrics, the algorithm was split up, following the divide and conquer concept creating a number of sub-calculations that provided the final algorithm with only the required data, and reducing complexity by removing superfluous information. The next section will highlight the different components within the sub-calculations and how they make up the main algorithm.

3.1. Algorithm Components

This algorithm will have a variety of applications including scenario simulations identifying best-fit liveness and coercion standards; as well as identifying most suitable fusion locations and so on. When used in this way the fusion identification tool would improve biometric security development as it would allow a more thoroughly tested system without the same degree of expense. Another potential route would be by using the algorithm within an automated system as the values generated by the algorithm could be used to adapt autonomically, creating context-aware security techniques that could dynamically change the security provision ad-hoc. However, to begin time will be considered and will have the notation T.

3.1.1. Time

Time has been used because it is a good way to help highlight the accuracy of a system, and is necessary to provide much-needed context to the other components. For example, if the algorithm output is based on one day only, then there may be events specific to that day that cause intra-variations, or when the measurement time period is a longer or shorter value. Subsequently, the time value is mutable and can differ depending on the level of security and the expected outcomes. Small scale time periods may be fine for initial tests between 5-10 hours etc. However if the time value is desired to be longer, greater than 24 hours for example, then other factors must also be considered such a noise, user-base size, etc. As most security systems are in place for longer than one day that it is a reasonable assumption that there are a plethora of other components to consider.

The purpose of time within the algorithm is to create a snapshot over a specific period, the longer the time T, the more potential security threats and problems may occur. Therefore within this algorithm, time will indicate the time of execution.

Whilst this version of time could be viable; a variation that became apparent was based around usage as well as time, what if users logged on multiple times in a day. This may be more suitable within scenarios where users log on and off multiple times a day, such as teaching in different classrooms and so on. These constant authentication attempts can pose security threats, and make the system more vulnerable to spoof attack for a number of reasons. Firstly, the more attempts to authenticate means more chances for a nefarious attacker to get into the system. Secondly, the more authentication attempts the more chance of an administrative attack to occur. To highlight this user traffic concept, instead of accessing time only, the algorithm would make use of an average authentication request over a set period of time. For example user 'x' has logged in once in this 24 hour period, however, user 'y' has logged in 10 times over the same 24 hour period, therefore, increasing interactions by 900%. One additional consideration with this measurement is that the more times a user is logging into the system the more normal it becomes, and then if the system, autonomically or manually, is looking for suspicious login behaviour, it will be harder to highlight.

Within the algorithm time is denoted by the value T, which equals time taken during authentication, therefore, the more authentication attempts that occur, the higher the time value will become. To generate this value the authentication attempts are combined with the time taken. Therefore, the greater amount of authentication will present a higher value and, therefore, more options for security breaches to occur. These values were divided by the Tn component, with n being the number of departments/samples within the calculation (this would depend completely on the scenario), therefore always keeping the value to a manageable number. An example of this is shown in Equation 3; where a_i equals number of people within department with a sequence - 16, 25, 10, 40; l_i equals logins per department with a sequence - 5, 30, 15, 80; k equals constant value less than or equal to one; t_n equals summation samples of $n + 1$.

$$T = \frac{\sum_{i=0}^{n} \frac{a_i}{l_i+k}}{t_n}$$

$$T = \frac{\sum_{i=0}^{n} + \binom{16}{5+1}\binom{25}{30+1}\binom{10}{15+1}\binom{40}{50+1}}{4}$$

$$T = \frac{9.2}{4} = 2.3 \tag{2}$$

This value allows the inclusion of both time taken to authenticate and a number of times authentication occurs. Obviously, the higher either of these values, the more chance a security threat can occur. This is due to the correlation between the number of authentication attempts and increased potential of attacks. The same can be said that the longer a system is in use the more security threats will occur.

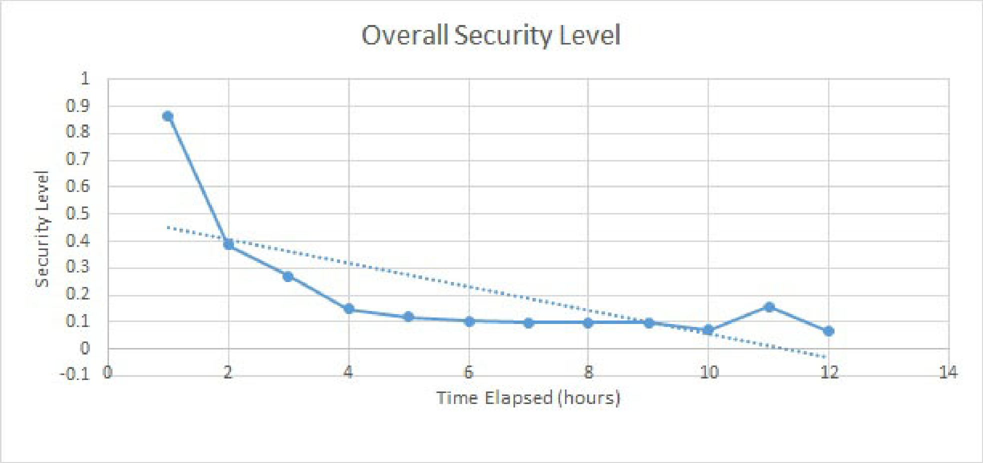

Figure 4: Time elapsed (hours) vs Security

Figure 4 shows the relationship between time elapsed and A_s, the autonomic security level of the technique combination and gathered from processing the algorithm, and it can be seen that there is a sequential degradation in security performance as the time progresses, however, there is a large jump towards the end of the 12-hour segment. To identify this anomaly, the data has been analysed and a problem regarding the user login and attempts data, specifically that there are fewer logins on certain days when compared to the number of people who can log in. This is interesting as there are some potential ramifications here: firstly, and most benignly, are the full complement of users not utilising the system. This could be due to some factors such as staff illness, holiday, and lack of need, etc. and while this does impact the system overall it does not represent a threat innately. This adds to the threat vector research highlighted by [55] [56] [28]. It presents a potential threat vector: if the normal login quantity is not being reached then the addition of a nefarious user will be harder to detect. The second factor could be due to a security breach within the system that is preventing legitimate users access, concepts such as poisoned cache attack, comprised password databases, etc. would prevent system use becoming a denial of service based problem. The third option is that there is a legitimate fault with the system, or there is an administration based attack [15].

As figure 5 shows, the Dr is very resistant to T changes, however, changes in the liveness or coercion provision produce bigger changes as can be seen. Techniques that have a low permeance level, and therefore do not have the flexibility when gathering samples, will negatively impact the overall score, as ideally, a sample would be very resistant to mutability. This is not always practical, as this example highlights: a user that normally has no issues using a palm print scanner, however, due to an accident they have their arm in a cast and subsequently cannot provide the required sample [17]. If the technique instead was fingerprint based then due to the high permanence and collectability the system would still be able to access a sample unless the entire hand is no longer available. This outlier will always be present, but this algorithm will endeavour to minimise the impact of them. Therefore when fusion occurs with two or more techniques the better they work together and the overall higher level of Dr meaning it is more resilient to deviations, make the overall system more secure and reliable.

To gather the data for this testing process a University in the Northwest of the UK is being used as a case study. All of the information is being gathered from their website and other public data repositories, for ease of replication. For this work, universities are being considered as the main focus of data. This is simply due to the ease of data collection, however, more information would be advantageous in the future to denote specific user requirements.

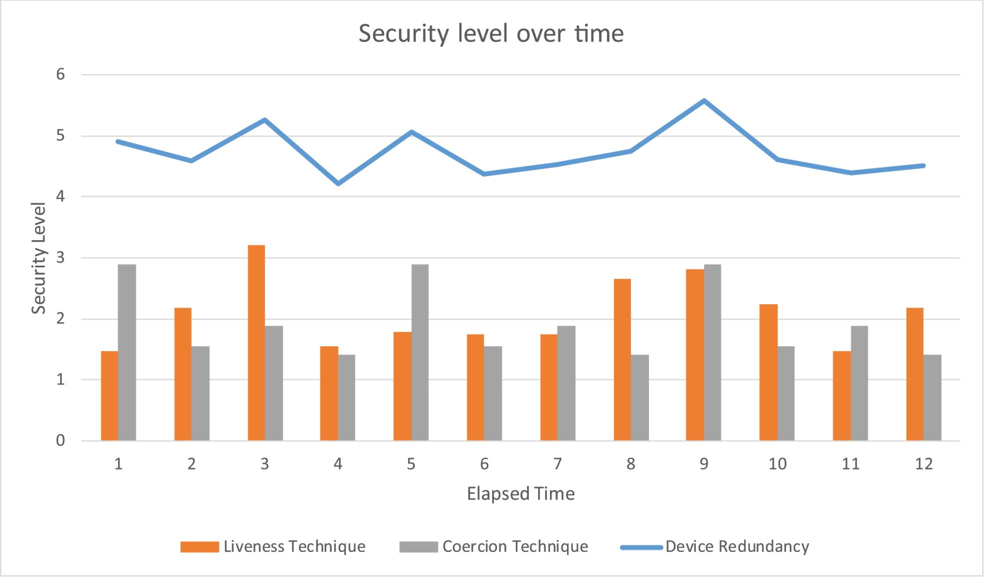

Figure 5: Security level over time

3.1.2. User-base

The user base is the area most prone to deviations, as the range of medical/behavioural/noise characteristics can dramatically change depending on location, and organisation. This component identifies the number of users being parsed throughout the system during a specific period. This metric can change dramatically depending on date and time, for example: when considering frequency and number of logins per day as well as the user base, an initial assumption would be that the more users within a system then, the more logins will occur. While this is a reasonable assumption, other situations can impact this dramatically. For example, it would be reasonable to expect ten users to have fewer authentication attempts than 20 users. However, if the ten users are power users, and log on 30 times each within a particular session equalling 300 authentication attempts and the 20 users log on ten times each equalling 200 authentication attempts, it shows that the values alone do not show all of the potentials. This highlights the importance of user location and composition when considering what to expect from the user-base. Therefore, the purpose of this user-base metric is to identify the styles of access attempts, when will these access attempts occur, and what specific variations are within the user base. A broad generalisation will be identified for the user base assuming that everyone can access and use the security techniques.

Subsequently, both the number of users and the usage per user must be considered as both will have an impact on the system. Firstly the user amount will be discussed.

3.1.3. User Amount

The first thing that is identified is the number of potential users within an installation. For the purpose of this example, a university is used: and within this example, there are three categories of measurement each representing a different organisational group within the university; a department, a school(which comprises of one to three departments) and a faculty comprising of 'n' schools and departments. As there can be a variety of values within each area, the mean of users will be found which will enable a more regulated calculation to be developed. The mean is being used as it is almost impossible to predict, without additional tools, how many authentications attempts there will be on a system at any one period. This minimises the impact of negative user involvement such as time periods that normal user authentication is minimised e.g. illness, holidays, meetings, etc.

To better visualise this data; Table 2 identifies a faculty level calculation with each row denoting a school or department. Whilst Table 3 shows the mean data assuming that 'n' is the number of users within a department, and A_n equates to the sequence of users:

Users (A_n)	Attempts to Authenticate (A_j)
16	20
25	100
10	10
40	50
55	200
10	40
18	10

Table 2: Mean Data

This would then be used to create the sum of users within the simulation as shown by Equation 3:

Sequence
A_n =16,25,10,40,55,10,18

Table 3: Data Sequence

$$\begin{gathered}\sum_{i=1}^{n} A_n = a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7 \\ \sum_{i=1}^{n} A_n = 16 + 25 + 10 + 40 + 55 + 10 + 18 = 174 \end{gathered} \tag{3}$$

It may seem unnecessary to include the sigma summation for this simple equation. However, this example has limited data complexity. If implemented in a large enough organisation the amount of data may become extremely large and, therefore, would become more difficult to manipulate, hence the future proofing technique. The final stage would be to calculate the final mean as shown by Equation 4:

$$\sum_{i=1}^{n} A_n = 16 + 25 + 10 + 40 + 55 + 10 + 18 = \frac{174}{7} = 25 \tag{4}$$

Within this data, the mean value for users is 25 this can then be used to show how many users are authenticating within the system and is the first stage of creating valuable data that can be used within the overall algorithm; as shown in Table 4.

One technique that this metric lends itself to is the application of autonomic data collection and context awareness [57] [58]. The inclusion of this logins per organisation structure would allow the autonomic environment access to important data regarding user log on trends. This would further allow self-optimisation techniques that improve resource allocation to certain environments or changing security techniques completely.

To further evaluate the data additional factors would be needed to be considered such as the mean and absolute deviation. If Table 4 is used as

Actual value	Mean Average Deviation (25) Users (A_n)
16	9
25	0
10	15
40	15
55	25
10	15
18	7

Table 4: Mean Variation

the base level then Equation 5 shows the absolute deviations and Equation 6 the mean deviation.

$$\sum_{i=1}^{n} x - \theta = 9 + 0 + 15 + 15 + 25 + 15 + 7 = 86 \tag{5}$$

This can then be used to find out the mean divergence, therefore allowing the algorithm to check if a department is varying too much from the mean, which could indicate a system error, lack of resources, or a potential security issue.

$$\frac{\sum_{i=1}^{n} x - \theta}{N} = \frac{9 + 0 + 15 + 15 + 25 + 15 + 7}{7} = 12.28 = 12 \tag{6}$$

While this data is useful from the algorithmic standpoint, simply understanding the number of users is not sufficient, as it is unlikely that users will only log in once a day, indeed depending what environment is being worked within, the logins may contain huge divergence intra-departmentally. This is due to the amount of workforce that have a permeance altering factor associated with their account. Therefore, the next stage of this metric's development would be to identify the number of logins per period; this could then be combined with some users to produce a valuable metric which will identify the number of average attempts to authenticate over each period.

To do this: the average number of attempts will be gathered, this information can be used within either the time or the user component of the algorithm. Equation 7 follows the same equation process as Table 3 and Equation 3. Firstly users need to be identified, secondly average attempts to login/authenticate within a period e.g. hours, days, months, etc. need to be identified. This is seen within: Table 2 (A_n and A_j).

$$\sum_{i=0}^{n} \frac{A_n}{n} = 16 + 25 + 10 + 40 + 55 + 10 + 18 = \frac{174}{7} = 24.85$$
$$\sum_{i=0}^{n} \frac{A_n}{n} = 25 \tag{7}$$

$$\sum_{i=0}^{n} \frac{A_j}{n} = 20 + 100 + 10 + 50 + 200 + 40 + 10 = \frac{430}{7} = 61.42$$
$$\sum_{i=0}^{n} \frac{A_j}{n} = 61 \tag{8}$$

Now that both of the means have been calculated the next stage is to generate the product of the mean.

$$\sum_{i}^{n} \frac{A_n}{n} \div \sum_{i}^{n} \frac{A_j}{n} = 61/25 = 2.4 \tag{9}$$

After these means have been gathered and the product of both is calculated and the average authentication attempts for the particular period will be highlighted, in this case, one day for a university faculty as shown in Equation 9. The main problem here is that the time data is static and the user data is only based on a single area. To make this more useful for the full algorithm, more output is needed, potentially including multiple times values/multiple faculties and so on. Currently, all that can be gathered from the data is how many average attempts a period there are. This then needs to be combined with other aspects within the overall algorithm for context e.g. liveness detection false acceptance rate (FAR) of 0.01 means that if this number is too great, it will cause authentication problem as users that should not be able to access the system will be able to do so [17].

Currently, the time and user-base have been identified, whilst the users/ participants have been considered it is imperative that they are focused on. The subsequent sections will look into this

3.1.4. Participants

After T has been calculated P must be found, and the following equation finds out the value of P, which is participants, which in turn denotes the user data for testing using the ranges identified within Table 5.

Participants Calculation
a_i = number of people within department.
l_i = logins per department.
k = constant value less than or equal to one.

Table 5: Participants Calculation

To begin, P will always require to be a positive number, otherwise it will cause the security value to drop unnecessarily, and whilst this might be useful in the future it is not viable now. Then a_i and l_i and used to gather the summation value, which indicates the total amount of logins in a period of time, finally the constant is included to keep the positive value as mentioned above, this is seen in Equation 10.

$$\begin{aligned} P = 0 \leq \left(\sum_{i=0}^{n} \tfrac{a_i . l_i}{n} \right) + k \leq 1 \\ P = 0 \leq \frac{\big(16 * 5\big) + \big(25 * 30\big) + \big(10 * 15\big) + \big(40 * 80\big)}{4} + 1 \leq 1 \\ P = 1045.25 \\ a_i = 16, 25, 10, 40 \\ l_i = 5, 30, 15, 80 \end{aligned} \tag{10}$$

The reason for this distinction is that while a user may log in only once a day within department X, department Y may face a huge amount of logins due to different classrooms, installations, subjects, etc. The summation calculates this value and makes sure that it is greater than zero. This allows the final value to be included in the overall algorithm. Now that both time and user base have been identified the next area to consider is the inclusion of both liveness and coercion detection techniques.

3.1.5. Liveness Detection and Coercion Detection

Current research highlights methods of biometric metrics, and, for the purpose of these papers, the metrics discussed in [17] and [9] will be used with the addition of an interval measurement system proposed in [6]. This will allow a method of measuring the effect implementing both liveness and coercion techniques into a system, with more meaningful data that can be compared and evaluated more accurately than the ordinal techniques used in the past. Subsequently, the results can be used to highlight if a technique is better or worse for a given scenario, with higher output denoting more secure techniques. This will also allow for the inclusion of additional features such as the specific scenario factors e.g. an installation might need high security but has very little space for additional hardware [59]. Therefore, a software/intrinsic solution, while not as robust from a security view, may be a better option. By making this algorithm as adaptable as possible allows a more personalised security application process, and the use of these metrics makes the application within autonomic environments easier to develop.

Biometric devices can be defined by a variety of these salient characteristics and a plethora of considerations must be taken into account when deciding on suitable procedures for a system. Normally the choice of such techniques includes factors such as device location, liveness technique, etc. Therefore, the liveness and coercion techniques will be measured against the following factors: universality, permanence, collectability, performance, acceptability and circumvention [17]. These have been populated using a selection of liveness standards as shown in Table: 6 and then Table 7, the data has been gathered from four techniques, two liveness [60] [61] and two coercion standards[6].

These metrics were gathered by assigning values to different liveness and coercion techniques as shown in [6]. This took into account factors such as hardware, software requirements, ease of sample collection and so on. Currently, this is somewhat dependent on the interpretation of the developer, but once again as more usage occurs the easier it will be to compare values.

The purpose of this component is to identify the different biometric devices and techniques that work best together; which can be based on multi-

Table 6: Liveness and Coercion Techniques

Technique	Universality	Permanence	Collectability	Performance	Acceptability	Circumvention	Total
Liveness - Facial Modality	1.67	4.00	4.00	5.47	1.00	3.12	3.2
Liveness - ECG based biometric identification	2.33	1.33	5.00	2.13	1.00	1.62	2.2
Coercion - Skin Conductivity	2.00	1.00	4.00	1.80	1.00	1.45	1.9
Coercion - Facial Micro-Movements	1.67	1.00	1.67	1.53	1.00	1.63	1.4

Liveness Technique	Coercion Technique
1.47222	2.888889
2.18055	1.548611

Table 7: Algorithm Scenario Values

layer fusion and in this situation specifically detailing liveness and coercion. This component will have some sub-calculations that will provide the final device redundancy value for use in the main algorithm.

3.1.6. Device Redundancy

The device redundancy (D_r) is the metric that utilises this liveness and coercion data. This provides a level of security with lower values denoting a secure system while higher values denoting less secure. These base values are shown in Table 9 and calculated using the equation shown in Equation 12.

It is possible for this value to be 0 which would mean that there are no liveness or coercion techniques included. If this occurs then, the values being zero will have an overall negative effect throughout the algorithm. Therefore to prevent this problem occurring, a constant has been used to add to the sub calculation which is the standard exponent equalling 2.718.

$$\sum_{i=0}^{n} \left(\tfrac{l_d+c_d}{2}\right) + e^1$$

$$D_r = \sum_{i=0}^{n} \left(\tfrac{1.33+1.33}{2}\right) + e^1 = 6.298$$

(11)

Within Equation 11 the D_r is found, this is done by taking the values highlighted in Table 7 which equate to l_d for the liveness detection and c_d for the coercion detection. This is because the choice of techniques will have the greatest impact on whether a user can access the technology, i.e. a user with no hands is unable to use fingerprint-based techniques and so on. There is also a constant used as if the value would equal zero then the algorithm would fail to run. This prevents the lack of liveness and coercion standards causing failures within the system and would represent a single liveness and coercion technique being used. Whilst this would not provide the most secure system, it is a common theme to have a unimodal approach, especially for older installations. However, there is the potential that fusion techniques could be encountered therefore the algorithm must be robust enough to support these factors. As well as the device redundancy the composition of the workforce/user base must be considered. This is a much greater concern for biometric-based security since the massive impact feature deviation can have on the security of a system.

3.1.7. Anomalous User-base

The permanence and collectability of biometric samples are a key factor within the biometric classification and this has a deep impact on the individual techniques, it is also a factor that needs to be highlighted within the overall algorithm. If a user is unable to use a biometric due to impairment, then this must be taken into account when designing the system's security, otherwise ostracising a portion of the users, again highlight the potential advantage for dynamic security and autonomous provision.

This data can be gathered by applying permanence and collectability metrics, therefore, highlighting what could affect the techniques and routes. Ideally, up-to-date accurate data would be used here, gathered from HR departments within a company. However, for this simulation, the following assumptions identified within Table 8 have been used to highlight the ratio of impaired to non-impaired staff. This can be used as an example baseline if the actual value is unknown.

Table 8 shows that, within the UK, there is a large percentage of the working population who have a disability, as defined by the UK Government, and this calculation indicates that for every one officially categorised

Metric	Percent	Actual
Current Population	100%	63,000,000
<14 Years	17.60%	11,088,000
>64 Years	16.40%	10,332,000
Non-workers	34.00%	21,420,000
Potential working pop	66.00%	41,580,000
Pop with disability	16.40%	10,332,000
% disabled pop in workforce	46.00%	4,752,720
% non-disabled pop in workforce	76.00%	36,827,280
Disabled to non-disabled workforce	7.748674	8:1

Table 8: Ratio Impaired/Non-Impaired

disabled person within a particular workforce, there are eight non-disabled members of the same workforce [62]. Therefore, this component highlights an example value which uses the above ratio of people with impairments. If a sample organisation of 20 is considered, then it would be expected that two members of that workforce have an impairment. This means that there are at least two users that might not be able to provide some of the common biometric samples dependent on the impairments in question, especially as mobility issues are some of the most common impairments and subsequently techniques such as gait, facial or iris scans, etc. may encounter problems [62]. The notation for this value is A_b and takes into account participants within a system that will have an impairment that will cause a technique to work either sub-optimally or not at all. This follows the 8:1 ratio shown in Table 8 and the calculation can be seen in Equation 12. This shows that in using the data highlighted, there will be 10.1 members of the workforce that potentially can have some permanence impacting disability.

$$\begin{gathered}\sum_{i=0}^{n} \left(\tfrac{A_n}{n} = 16 + 25 + 10 + 40 = 91 \right) \\ \frac{91}{9} = 10.1\end{gathered} \tag{12}$$

By taking all of these metrics into consideration; these components will identify different stakeholder groups that can impact the security within a system, and are catered to biometric security specifically, culminating in Equation 13, the final algorithm.

$$A_s = \frac{T}{\frac{P-A_b}{P} * D_r} \tag{13}$$

The purpose of these metrics is to highlight different variables that can affect security when focusing on liveness and coercion detection. These are time, users, device redundancy, liveness and coercion detection. The following section will detail these factors. These factors will produce an output 'A_s'which will denote the overall level of security for the current system set-up, allowing comparison to other installations by changing the liveness and coercion standards used.

4. Testing and Evaluation

This paper claims that the algorithm presented will enable systems architects to understand potential fusion between the two sub-layers of proposed biometric security after sample acquisition: liveness detection and coercion detection. This algorithm will show how different combinations of techniques can be used within a specified organisation, therefore creating an aimed metric that can inform the most suitable choices of techniques.

The evidence to support this is seen by gathering information relating to a variety of features and then subsequently highlight their appropriateness within a specific installation. In this section, the algorithm will be tested using a range of dummy data to show how the overall environment works. In this testing scenario, the dummy company data in Table 9 will be used. This data is based on the statistical data gathered from [63]. With the login attempts this represents user that might log on only once, whilst others might log on and off a number of times per day, for example due to teaching duties in different rooms. The values at this time are random, but within an active scenario these values would be gathered from the organisation's IT department and would be focused depending on the installation requirements. For example if the installation is occurring within the School of History then only those logins would be required and so on.

As well as the user data there also needs to be liveness and coercion technique data, as the changing of these factors will allow different techniques

School	Departments	Users	Login Attempts
School of Technology		91	130
	Computing Science	16	5
	Engineering	25	30
	Mathematics	10	15
	Applied Technology	40	80
School of History		91	130
	Classical Studies	32	32
	China Studies	25	18
	Medieval Studies	18	22
	Archaeology	52	110

Table 9: Initial Company Data

to be tested before expensive implementation needs to occur. By using a taxonomy to categorise different liveness and coercion techniques, a value denoting the viability of said techniques can be identified and used. The taxonomy that is used to classify the information is highlighted within [6]. Two examples for each are as shown within Table 9.

Therefore, the two schools will be tested with four techniques, the combinations of these liveness and coercion detection practises are as follows:

1. Facial Modality and Skin Conductivity Tests
2. Facial Modality and Facial Micro-Movements
3. ECG Techniques and Skin Conductivity Tests
4. ECG Techniques and Facial Micro-Movements

Therefore if the School of Technology is considered first then Equation 14 would be as follows, with values better the nearer to 1:

$$a.A_s = \frac{4181}{\frac{1045.25-10.11}{1045.25} * 5.259948} = 0.767891837$$

$$b.A_s = \frac{4181}{\frac{1045.25-10.11}{1045.25} * 4.544671} = 0.960678883$$

$$c.A_s = \frac{4181}{\frac{1045.25-10.11}{1045.25} * 5.259948} = 0.846084856$$

$$d.A_s = \frac{4181}{\frac{1045.25-10.11}{1045.25} * 5.259948} = 0.888748991 \tag{14}$$

This shows that within the school of technology the best techniques would be option B - containing facial modality [60] along with facial micro-movements. Whereas the worst combination would be facial modality [60] with skin conductivity tests - option A. However, when using the same coercion and liveness data on the second test set (school of history), the same pattern can be seen in Equation 15.

$$\begin{aligned}
a.A_s &= \frac{7591}{\frac{1897.75-14.11}{1897.75} * 5.259948} = 0.766160666 \\
b.A_s &= \frac{7591}{\frac{1897.75-14.11}{1897.75} * 4.544671} = 0.958513083 \\
c.A_s &= \frac{7591}{\frac{1897.75-14.11}{1897.75} * 5.259948} = 0.844177402 \\
d.A_s &= \frac{7591}{\frac{1897.75-14.11}{1897.75} * 5.259948} = 0.886745353
\end{aligned} \tag{15}$$

Once again it shows that while the overall values have changed, this is encountered for by the increased size of the raw data, and within this area, the same two techniques are the best (b) and the worst (a).

These data values can change dramatically, and more or less variation can be seen by changing the number of users within the system, the number of logins per day per person, along with the composition of the workforce. Currently, this algorithm utilises a 1:8 ratio of an impaired workforce, however, this value can be made much more specific with actual company information. This alongside the data specificity is the two main problems that have been identified within this work

During this testing and analysis of the algorithm and data, some factors became apparent that are of significance. To further improve the algorithm process and the applicability within the security design environment, it is imperative that these are addressed. They are:

1. Single Data Reliance
2. Data Specificity
3. Interface development

4.1. Single Data Reliance

The initial observation when using this algorithm is that whilst it can correctly identify the security level of the overall system, the focus is on one specific set of data. For example one authentication style over a period of time. To take full advantage of this algorithm a way to apply the data gathered in an autonomic way, therefore, promoting ad-hoc security. Therefore if this comparison was adopted then the following would be seen, and they can be compared together, as shown by Graph 6.

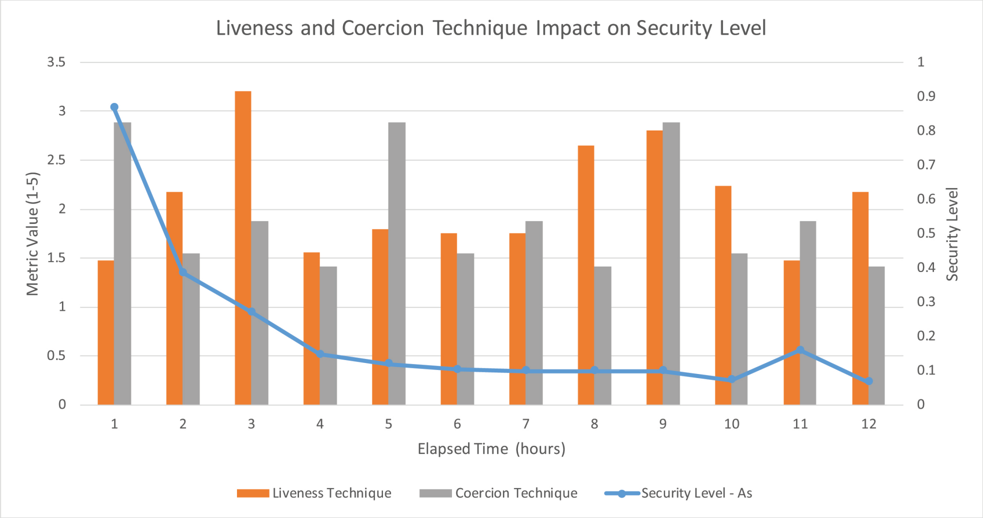

Figure 6: Real coercion and liveness data

Graph: 6 highlights the differences between the combinations of liveness and coercion techniques. These are grouped as appropriate to the sample provision, however, if completely separate techniques were being used together, to promote a more secure and robust environment for example, then the algorithm would be able to take this into account by simply inputting the correct values. For example, currently if focusing on fingerprint techniques then utilising the dermalog liveness technique [64] along with intentional false authentication coercion techniques [6] then there is an obvious connection between the two techniques. However, if the developers wanted a coercion technique that was specifically different from the original sample, then they could use facial micro-movements [6]. This could add a level of complexity to the system, would require more hardware and infrastructure. However overall it could be more secure. The key to this scenario is the ability to change the weightings of the algorithm depending on the scenario requirements. In this situation, if the above example was going forward the architects could reduce the impact of additional hardware within the algorithm, therefore, allowing techniques with different hardware requirements to be viewed more favourably, therefore, providing a dynamic and reactive technique that could adapt to the scenario it was being applied to.

When this comparison has been achieved it becomes easier to identify techniques that are superior to the others as they are numerically different from each other. In this case, it is obvious that the "best" technique is the most secure, and the technique that had the highest coercion and lowest liveness was the second most secure and most practical. When viewing this data other factors need to be considered, as this is not intended to be the only degree of information, instead, this is the top level component of the analysis.

4.2. Data Specificity

The second factor that became apparent when dealing with the algorithm is that there is a need for specific data and the more data it can parse the more robust and accurate it will become. This is because the algorithm uses real data to provide information about a technique or system over a period of time, therefore, demanding regular injections of data, which in the current environment is very difficult to provide. Therefore to improve the overall system more data needs to be injected into the algorithm allowing for greater evaluation. Coupled with this innate scalability, the practicality of including specific data, e.g. such as the data regarding the number of users and login attempts throughout the system, is something that is very important to include as it provides specific information about the requirements. Within this algorithm a rough ratio of 1:8 has been used to detect if a user has any impairment based problems utilising biometric environment, however, the actual data would be much more relevant to use within the overall algorithm.

4.3. Interface development

One factor that is not directly related to the technical areas of the algorithm, but has an enormous influence on the usefulness, is the lack of a user interface. The current algorithm is quite complicated to calculate however with the integration of a valid interface the algorithmic output could become much easier to work with and relevant to different users. This interface would enable researchers and developers to enter their data into easy to use forms which would convert the data, using the algorithm into meaningful outputs, including graphs etc. This is something that would be advantageous to the overall system, whilst not being urgent, therefore it would be suitable for future work to identify.

The factors have shown that while the overall modus of the algorithm works, several areas need improvement. Specifically, the use of accurate data regarding the logins per day; however, this is something that is only practically available when there is a specific organisation that is willing to provide this information. Which is why it was simulated in this paper. The next and most vital aspect to consider is the application of more data within the algorithm, allowing a much higher degree of confidence. This is something that is being considered in future work.

5. Conclusion

This paper has highlighted the construction and development of an algorithm that will be used to develop a comparable standardised value that can be applied to biometric system design and development — enabling system architects to optimise or compare system components and fusion technique. By allowing an easier to implement, and accurate, comparison of system capabilities the process of security development can be streamlined and will become scalable to future techniques and threat errors. To validate this hypothesis data from a UK university was used in the testing process, regardless the technique is applicable in any workforce and organisation. By using this model, architects can develop robust systems more easily, especially when including different security components such as liveness and coercion detection techniques. Currently, the main focus of these results has been from a developer standpoint, allowing optimisation and general reliability testing. However, a secondary area that this could make use of this technique would be autonomous system development. The value provided by this algorithm could be used to denote an individual's personalised security process. Therefore by changing the device/liveness/coercion techniques to better suit the individual depending on noise factors, or performance impact. For example: minimise the use of iris scanners if there are a large proportion of users with that specific disability such as Anophthalmia. This application of self-optimisation and context awareness would allow personalised security to be developed.

Acknowledgements

The author would like to thank Professor Mark Anderson and Professor Yonghuai Liu for all of their support, advice and suggestions.

Biography

Peter William Matthew received his PhD in 2016/17 from Edge Hill University and since then has been involved in researching biometric security techniques, and pattern recognition in education. His current research is focusing on the development and application of coercion detection methods within biometric systems. His research interests are coercion detection, autonomous biometric security and native animal recognition for welfare and agriculture.
Susan Canning works as a senior lecturer at Edge Hill University, and focuses on computer security and forensic computing. Her research interests include 3d visualisation techniques for forensic computing and innovative techniques in forensic teaching.

References

[1] M. Krieg, N. Rogmann, Liveness detection in biometrics, Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI) P-245 (2015). doi:10.1109/BIOSIG.2015.7314611.

[2] A. Hadid, N. Evans, S. Marcel, J. Fierrez, Biometrics Systems Under Spoofing Attack: An evaluation methodology and lessons learned, IEEE Signal Processing Magazine 32 (5) (2015) 20–30 (2015). doi:10.1109/MSP.2015.2437652.
URL http://ieeexplore.ieee.org/document/7194916/

[3] T. Ring, Spoofing: Are the hackers beating biometrics?, Biometric Technology Today 2015 (7) (2015) 5–9 (2015). doi:10.1016/S0969-4765(15)30119-3.
URL http://linkinghub.elsevier.com/retrieve/pii/S0969476515301193

[4] P. Gupta, D. Gao, Fighting Coercion Attacks in Key Generation using Skin Conductance, 19th USENIX Security Symposium, Washington, DC, USA, August 11-13, 2010, Proceedings (2010) 469–484 (2010).
URL http://www.usenix.org/events/sec10/tech/full$_{p}apers/Gupta.pdf$

[5] P. Gupta, X. Ding, D. Gao, Coercion resistance in authentication responsibility shifting, Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security - ASIACCS '12 (2012) 97 (2012). doi:10.1145/2414456.2414512.
URL http://dl.acm.org/citation.cfm?doid=2414456.2414512

[6] P. Matthew, M. Anderson, Developing coercion detection solutions for biometrie security, in: Proceedings of 2016 SAI Computing Conference, SAI 2016, 2016, pp. 1123–1130 (2016). doi:10.1109/SAI.2016.7556118.

[7] S. Soong, Judge Says Facial Recognition Unlocks Not Allowed Under 4th and 5th Protections.
URL https://www.documentcloud.org/documents/5684083-Judge-Says-Facial-Recogni

[8] A. Ross, A. Jain, Information fusion in biometrics, Pattern Recognition Letters 24 (13) (2003) 2115–2125 (2003). arXiv:arXiv:1011.1669v3, doi:10.1016/S0167-8655(03)00079-5.

[9] H. Van De Haar, D. Van Greunen, D. Pottas, The characteristics of a biometric, 2013 Information Security for South Africa - Proceedings of the ISSA 2013 Conference (2013). doi:10.1109/ISSA.2013.6641037.

[10] M. Leghari, S. Memon, A. A. Chandio, Feature-level fusion of fingerprint and online signature for multimodal biometrics, in: 2018 International Conference on Computing, Mathematics and Engineering Technologies: Invent, Innovate and Integrate for Socioeconomic Development, iCoMET 2018 - Proceedings, Vol. 2018-Janua, IEEE, 2018, pp. 1–4 (mar 2018). doi:10.1109/ICOMET.2018.8346358.
URL https://ieeexplore.ieee.org/document/8346358/

[11] R. Dwivedi, S. Dey, Score-level fusion for cancelable multibiometric verification, Pattern Recognition Letters (apr 2018). doi:10.1016/j.patrec.2018.04.022.
URL https://www.sciencedirect.com/science/article/pii/S0167865518301429

[12] G. S. Walia, T. Singh, K. Singh, N. Verma, Robust multimodal biometric system based on optimal score level fusion model, Expert Systems with Applications 116 (2019) 364–376 (feb 2019). doi:10.1016/j.eswa.2018.08.036.
URL https://www.sciencedirect.com/science/article/pii/S0957417418305517

[13] Z. Akhtar, C. Micheloni, G. L. Foresti, Biometric Liveness Detection: Challenges and Research Opportunities, IEEE Security and Privacy 13 (5) (2015) 63–72 (sep 2015). doi:10.1109/MSP.2015.116.
URL http://ieeexplore.ieee.org/document/7310809/

[14] Z. Akhtar, C. Micheloni, C. Piciarelli, G. L. Foresti, MoBio-LivDet: Mobile biometric liveness detection, in: 11th IEEE International Conference on Advanced Video and Signal-Based Surveillance, AVSS 2014, IEEE, 2014, pp. 187–192 (aug 2014). doi:10.1109/AVSS.2014.6918666.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6918666

[15] K. Nandakumar, A. K. Jain, Multibiometric template security using fuzzy vault, BTAS 2008 - IEEE 2nd International Conference on Biometrics: Theory, Applications and Systems (2008). doi:10.1109/BTAS.2008.4699352.

[16] M. Wolotsky, M. Husain, E. Choe, Chill-Pass: Using Neuro-Physiological Responses to Chill Music to Defeat Coercion Attacks (2016) 14 (may 2016). arXiv:1605.01072.
URL http://arxiv.org/abs/1605.01072

[17] A. K. Jain, K. Nandakumar, A. Ross, 50 years of biometric research: Accomplishments, challenges, and opportunities, Pattern Recognition Letters 79 (2016) 80–105 (jan 2016). doi:10.1016/j.patrec.2015.12.013.
URL http://www.sciencedirect.com/science/article/pii/S0167865515004365

[18] Dabanga, Migrants beaten for fingerprints in Italy, says Amnesty — Radio Dabanga (2016).
URL https://www.dabangasudan.org/en/all-news/article/migrants-beaten-for-finge

[19] J. Mortimer, Care charity forcing staff to sign in using fingerprints may be breaching law — Left Foot Forward (2018).
URL https://leftfootforward.org/2018/05/care-charity-forcing-staff-to-sign-in

[20] CaritasEuropa, Force fingerprinting of children - www.caritas.eu (2018).
URL https://www.caritas.eu/forcing-children-to-take-their-fingerprints-is-unw

[21] O. E. Dictionairies, Definition of coerce in English: (2014).
URL http://www.oxforddictionaries.com/definition/english/coerce

[22] T. Matsumoto, H. Matsumoto, K. Yamada, S. Hoshino, ¡title¿Impact of artificial "gummy" fingers on fingerprint systems¡/title¿, in: Proceedings of SPIE, Vol. 4677, 2002, pp. 275–289 (2002). doi:10.1117/12.462719.
URL http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=878135

[23] A. A. B, S. Schneider, Smart Card Research and Advanced Applications, in: Smart card research and advanced applications: IFIP TC8/WG8. 8 Fourth Working Conference on Smart Card Research and Advanced Applications, September 20-22, 2000, Bristol, United Kingdom, Vol. 7771, Kluwer Academic Publisher, 2013, pp. 152–167 (2013). arXiv:9780201398298, doi:10.1007/978-3-642-37288-9.
URL http://link.springer.com/10.1007/978-3-642-37288-9

[24] J. Guerra-Casanova, C. Sánchez-Ávila, G. Bailador-del Pozo, A. Santos-Sierra, Advanced Biometric Technologies, in: Advanced Biometric Technologies, 2011, p. 394 (2011). doi:10.5772/969.
URL http://www.intechopen.com/books/advanced-biometric-technologies

[25] W. Pohs, Building a taxonomy for auto-classification, Bulletin of the American Society for Information Science and Technology 39 (2) (2013) 34–38 (2013). doi:10.1002/bult.2013.1720390210.
URL http://doi.wiley.com/10.1002/bult.2013.1720390210

[26] G. Orrù, R. Casula, P. Tuveri, C. Bazzoni, G. Dessalvi, M. Micheletto, L. Ghiani, G. L. Marcialis, LivDet in Action - Fingerprint Liveness Detection Competition 2019, Tech. rep. (2019). arXiv:1905.00639.
URL http://arxiv.org/abs/1905.00639

[27] M. Ghorbani, M. Alizadeh, A. E. Omran, M. M. Asem, An Investigative Review of Human Authentication Based on Fingerprint, in: 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2018, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1359–1366 (jan 2019). doi:10.1109/IEMCON.2018.8614910.

[28] M. Joshi, B. Mazumdar, S. Dey, Security Vulnerabilities Against Fingerprint Biometric System (2018). arXiv:1805.07116.
URL http://arxiv.org/abs/1805.07116

[29] Kelvin Salton, Face Recognition: Understanding LBPH Algorithm – Towards Data Science (2017).
URL https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d

[30] G. Kim, S. Eum, J. K. Suhr, D. I. Kim, K. R. Park, J. Kim, Face liveness detection based on texture and frequency analyses, in: Proceedings - 2012 5th IAPR International Conference on Biometrics, ICB 2012, 2012, pp. 67–72 (2012). doi:10.1109/ICB.2012.6199760.

[31] S. Liu, Y. Song, M. Zhang, J. Zhao, S. Yang, K. Hou, An identity authentication method combining liveness detection and face recognition, Sensors (Switzerland) 19 (21) (2019). doi:10.3390/s19214733.

[32] S. Milborrow, F. Nicolls, Active shape models with SIFT descriptors and MARS, in: VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications, Vol. 2, SCITEPRESS - Science and and Technology Publications, 2014, pp. 380–387 (2014). doi:10.5220/0004680003800387.
URL http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/00046800038

[33] L. Wang, X. Ding, C. Fang, Face Live Detection Method Based on Physiological Motion Analysis, Tsinghua Science and Technology 14 (6) (2009) 685–690 (2009). doi:10.1016/S1007-0214(09)70135-X.
URL `https://www.sciencedirect.com/science/article/pii/S100702140970135X`

[34] A. Asthana, S. Zafeiriou, S. Cheng, M. Pantic, Robust discriminative response map fitting with constrained local models, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2013) 3444–3451 (2013). doi:10.1109/CVPR.2013.442.
URL `http://ibug.doc.ic.ac.uk/resources.`

[35] J. M. Saragih, S. Lucey, J. F. Cohn, Face alignment through subspace constrained mean-shifts, Proceedings of the IEEE International Conference on Computer Vision (2009) 1034–1041 (2009). doi:10.1109/ICCV.2009.5459377.
URL `https://www.researchgate.net/publication/221111184`

[36] FacePlusPlus, Face Landmarks - Face++- Face++ Cognitive Services (2018).
URL `https://www.faceplusplus.com/landmarks/`

[37] D. Cristinacce, T. Cootes, Feature detection and tracking with constrained local models (2006). doi:10.5244/c.20.95.

[38] J. Lee, S. Moon, J. Lim, M. J. Gwak, J. G. Kim, E. Chung, J. H. Lee, Imaging of the finger vein and blood flow for anti-spoofing authentication using a laser and a MEMS scanner, Sensors (Switzerland) 17 (4) (apr 2017). doi:10.3390/s17040925.

[39] D. Baldisserra, A. Franco, D. Maio, D. Maltoni, Fake fingerprint detection by odor analysis, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 3832 LNCS, 2006, pp. 265–272 (2006). doi:10.1007/11608288$_3$6.

[40] P. Johnson, S. Schuckers, Fingerprint Spoofing and Liveness Detection, in: Forensic Science: A Multidisciplinary Approach, Wiley-VCH Verlag, 2016, pp. 373–382 (mar 2016). doi:10.1002/9783527693535.ch16.

[41] D. M. Uliyan, S. Sadeghi, H. A. Jalab, Anti-spoofing method for fingerprint recognition using patch based deep learning machine, Engineering Science and Technology, an International Journal (2019). doi:10.1016/j.jestch.2019.06.005.

[42] R. Frassetto Nogueira, R. De Alencar Lotufo, R. Campos Machado, Evaluating software-based fingerprint liveness detection using Convolutional Networks and Local Binary Patterns, in: BIOMS 2014 - 2014 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications, Proceedings, Institute of Electrical and Electronics Engineers Inc., 2014, pp. 22–29 (nov 2014). doi:10.1109/BIOMS.2014.6951531.

[43] J. D. Preez, Liveness assurance in biometric systems, Ph.D. thesis (2006).

[44] L. A. Jeni, J. M. Girard, J. F. Cohn, F. De La Torre, Continuous AU intensity estimation using localized, sparse facial feature space, 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, FG 2013 (2013). doi:10.1109/FG.2013.6553808.
URL https://www.researchgate.net/publication/259714804

[45] F. Agrafioti, J. Gao, D. Hatzinakos, Heart Biometrics: Theory, Methods and Applications, in: Biometrics, InTech, 2011 (jun 2011). doi:10.5772/18113.

[46] S. T. Parthasaradhi, R. Derakhshani, L. A. Hornak, S. A. Schuckers, Timeseries detection of perspiration as a liveness test in fingerprint devices, IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews 35 (3) (2005) 335–343 (aug 2005). doi:10.1109/TSMCC.2005.848192.

[47] C. S. Avila, J. G. Casanova, F. Ballesteros, L. J. M. Garcia, M. F. A. Gomez, D. D. S. Sierra, G. B. D. Pozo, State of the art of mobile biometrics, liveness and non-coercion detection, Tech. rep., EC (2014).

[48] C. Perkins, D. Beecher, D. C. Aberg, P. Edwards, N. Tilley, Personal security alarms for the prevention of assaults against healthcare staff (dec 2017). doi:10.1186/s40163-017-0073-1.

[49] M. Bishop, Computer security: Art and science, 2nd Edition, Vol. 1, 2018 (2018). doi:10.1109/msecp.2003.1203217.

[50] L. Blum, F. Cucker, M. Shub, S. Smale, Complexity and real computation, Springer New York, 2012 (2012). doi:10.1109/ICPADS.2015.113.
URL `http://link.springer.com/chapter/10.1007/978-1-4612-0701-6`$_1$

[51] A. Levitin,
, Vol. 4, Villanove University, 2012 (2012). arXiv:arXiv:1011.1669v3, doi:10.1017/CBO9781107415324.004.
URL

[52] E. Horowitz, Design and Classification of Algorithms, in: Encyclopedia Of Computer Science, 2003, pp. 33–37 (2003).

[53] S. Nigam, D. Garg, Choosing Best Algorithm Design Strategy For A Particular Problem 2 . Comparison of algorithm Design Strategy 3 . Best Algorithm Design Selection for a particular problem, in: IEEE International Advance Computing Conference (IACC 2009), no. March, Patiali, 2009, pp. 6–7 (2009).

[54] A. V. Aho, J. E. Hopcroft, J. D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley, 1974 (1974).
URL `http://www.chapters.indigo.ca/item.asp?Catalog=booksItem=978020100029`

[55] C. Roberts, Biometric attack vectors and defences, Computers and Security 26 (1) (2007) 14–25 (2007). doi:10.1016/j.cose.2006.12.008.

[56] A. Bhargav-Spantzel, A. Squicciarini, E. Bertino, X. Kong, W. Zhang, Biometrics-based identifiers for digital identity management, Proceedings of the 9th Symposium on Identity and Trust on the Internet - IDTRUST '10 (2010) 84 (2010). doi:10.1145/1750389.1750401.
URL `http://portal.acm.org/citation.cfm?doid=1750389.1750401`

[57] W. D. Christensen, C. a. Hooker, Anticipation in Autonomous Systems: Foundations for a Theory of Embodied Agents, International Journal of Computing Anticipatory Systems 5 (2000) 135–54 (2000).

[58] P. Lalanda, J. A. McCann, A. Diaconescu, Autonomic Computing, Innovations in Systems and Software Engineering v1 30 (April) (2013) 79–88 (2013). doi:10.1007/978-1-4471-5007-7.
URL `http://link.springer.com/10.1007/978-1-4471-5007-7`

[59] D. Hartung, Vascular Pattern Recognition and its Application in Privacy-Preserving Biometric Online- Banking Systems, PhD Thesis (Gjovik University College), 2012 (October) (2012) 27–45 (2012).

[60] R. Gettu, S. N. Shareef, K. J. Ernest, Evaluation of the robustness of SCC, Indian Concrete Journal 83 (6) (2009) 13–19 (2009).
URL http://www.ijetae.com/files/Volume3Issue12/IJETAE$_1$213$_1$11.*pdf*

[61] P. Luesakul, Voces exc??ntricas de la Argentina del siglo XIX en Finisterre de Mar??a Rosa Lojo, Rilce 32 (1) (2016) 182–200 (sep 2016). doi:10.1016/j.jnca.2014.04.008.
URL http://www.sciencedirect.com/science/article/pii/S1084804514000915

[62] Department for Work and Pensions, Disability prevalence estimates 2002/03 to 2011/12 (Apr to Mar) - Publications - GOV.UK (2014).
URL https://www.gov.uk/government/statistics/disability-prevalence-estimates-2002

[63] Hesa.co.uk, Overview - HESA - Higher Education Statistics Agency (2016). doi:10.1103/PhysRevB.79.054510.
URL https://hesa.ac.uk/overview

[64] Dermalog, Dermalog Afis (2013).
URL http://www.dermalog.com/pdf/AFIS.pdf