Privacy and Security Enhancements in Biometrics

Terrance E. Boul1,2 and Robert Woodworth2

1 University of Colorado at Colorado Springs, Dept. of Computer Science, EAS#3, 1420 Austin Bluffs Parkway, Colorado Springs CO 80933-7150 thoulit@vast.uccs.edu

2 Securics, Inc, Colorado Springs, CO. rwoodworth@securics.com

Abstract. Many tout biometrics as the key to reducing identity theft and providing significantly improved security. However, unlike passwords, if the database or biometric is ever compromised, the biometric data cannot be changed or revoked. We introduce the concept of Biotopes™: revocable tokens that protect the privacy of the original user, provide for many simultaneous variations that cannot be linked, and that provide for revocation if compromised. Biotopes™ can be computed from almost any biometric signature that is a collection of multi-bit numeric fields. The approach transforms the original biometric signature into an alternative revocable form (the Biotope) that protects privacy while it supports a robust distance metric necessary for approximate matching. Biotopes provide cryptographic security of the identity; support approximate matching in encoded form; cannot be linked across different databases; and are revocable. The most private form of a Biotope can be used to verify identity, but cannot be used for search. We demonstrate Biotopes derived from different face-based recognition algorithms as well as a fingerprint based Biotope and show Biotopes improve performance, often significantly.

The robust “distance metric”, computed on the encoded form, is provably identical to application of the same robust metric on the original biometric signature for matching subjects and never smaller for non-matching subjects. The technique provides cryptographic security of the identity, supports matching in encoded form, cannot be linked across different databases, and is revocable.

1 Introduction: Background, Motivation and Related Work

In the current debate about biometric and privacy, the two opposing sides tend to take ideological positions which not only leave no room for compromise, but which leave no room for novel solutions that simultaneously improve both security and privacy. One side, claiming to be focused on security, profess that biometrics are not secrets, they can be acquired by following someone around in public, and therefore the biometric data should have no expectation of privacy and any use of biomet-
rics to improve state or corporate security should be pursued. The privacy advocates, on the other side, consider biometric data inherently private and oppose almost any usage of them. Privacy-based arguments can derail programs, e.g. when Quantas Airline wanted to start using a biometric based procedure, the Union threatened a strike if implemented citing privacy concerns (Thieme 2003). The results of biometric ideological posturing can be a reduction for everyone in both security and privacy. As we will show in this chapter, while biometrics are not secrets, they MUST be protected if, in the long run, we are to use them for security. We will also show that with recent technological advances, some uses of biometrics can be privacy enhancing.

There are many significant privacy concerns with biometric systems. (Thieme-2003) divides the concerns into two classes: personal and information privacy. Some of the informational privacy concerns can be addressed with information security. We do not discuss the privacy issues that can be addressed with proper traditional information security, except to note that within the US, sadly, there are no requirements for instituting such protection measures.

A 2002 poll Commissioned by SEARCH (ORC-2002), a group funded by the U.S. Bureau of Justice Statistics, looked at perception among biometric users. Of those who underwent some kind of biometric identification in 2002, the survey found that 88 percent were concerned about possible misuse of their personal information. However, 80 percent said they support governmental and private organizations’ use of biometrics “as a means of helping prevent crimes”. Both the ORC survey and (Thieme 2003) raise issues of function creep, something that happened with the social-security number. There is concern about such private data, especially fingerprints with their association with criminal investigations, being required and stored in many locations by many different agencies with varying policies and security. One such example of function creep is that a unique biometric stored in different databases can be used to link these databases and hence support non-approved correlation of data. Additionally, there is the concern that searchable biometric databases could be sold or shared and then combined with a covertly obtained biometric data (e.g. face image or latent fingerprint) to find additional information about a user. This function creep is not hypothetical as documented in (Krause 2001), Colorado DMV records, which include both face and fingerprint data, are available to “any government agency that cares...”, and there was an attempt to sell the database publicly. (Krause 2001) also discusses an instance where the police in Tampa, Florida used face recognition software on football fans at the Super Bowl without the knowledge of the people involved. The potential for abuse in such systems feeds the fears about privacy invasion of biometrics. Nevertheless, with all the fears this raises, it is not the biggest concern of the authors. The next subsection discusses what we see as one of the most significant challenges for biometrics.

1.1 The Biometric Dilemma and the biometric risk equations

The key properties of biometrics, those unique traits that do not change significantly over a lifetime, are also their Achilles heel. The biometric dilemma is that while biometrics can initially improve security, as traditional biometric databases become widespread, compromises will ultimately destroy biometrics’ value and
usefulness for security. One of the security issues here is spoofing, i.e. literally printing to generate fake “gummy fingers” from images of a fingerprint (Matsumoto et al 2002). Reducing the number of databases with biometric data can reduce the potential for spoofing, but there is an even broader issue. While many people like to think of biometrics as “unique”, they are not, at least not with the level of data we can measure. Even FBI examiners have made high-profile misidentifications. Automated systems have even greater challenges. The best fingerprint systems tested by the US government in the NIST Fingerprint Vendor Recognition test, only had 98% true acceptance rates (TAR), when set to reject 99.99% of false matches, and had an equal error rate (ERR) of 0.2% (ERR is when false match rate=false reject rate and is a common measurement of a system performance).

In a database of millions, it is likely, even at 99.99% True Reject Rate, to find an effective biometric doppelganger, i.e. someone your fingerprint just happens to naturally match well enough for the system. Of course not every system will be using the best system, the average ERR for the top 10 manufactures was 3.6% (NIST- FPvTe03)]. The majority of the systems having FAR above 1 in 100 if they would be configured to only reject correct 5% of the correct users (i.e. TAR=95%).

Understanding the implications of this approximate match property, and the issues of large databases, is critical. We now develop a formal model of the risks and rewards. While a motivated individual might follow someone around at a direct cost $c_f$, and following has some risk of being caught, say the cost of the risk from following is $r_f$. Once obtained, the expected value over the persons lifetime for using it for spoofing is $v_s$ and the risk cost of using the spoof is $r_s$. (This is an average value, a type of simplification as realistically it would depend on the application for which spoofing was being used, e.g. for unsupervised access, there is little risk for spoofing, but for supervised the risks are much greater.) With probability $p_d$, the acquired data could also provide for doppelganger access, so there is also potential value/risk if that biometric happens to be a doppelganger, which we represent as $v_d$ and $r_d$ respectively. As long as the cost and punishment outweighs the rewards the data is relatively safe. To formalize this we introduce the biometric risk equation

$$\max(v_s - r_s, p_d (v_d - r_d)) - (c_f + r_f) > 0$$

(1)

that determines when the individuals biometric data is probably at risk because potential reward outweighs costs those with nefarious intent. The maximum over the first two terms is another expectation simplification because it is, on average, either more profitable to spoof or to try a doppelganger attack.

If we build a large centralized database, with say $N$ records, the costs of obtaining it will be different, say $c_e$ and the risks of hacking the central DB will be $r_e$. This leads to the centralized biometric risk equation

$$N * \max(v_s - r_s, p_d (v_d - r_d)) - (c_e + r_e) > 0$$

(2)

or conversely the centralized biometric DB is at risk if the using the biometrics once obtained has a positive incentive (i.e. if either $v_s - r_s > 0$ or $(v_d - r_d) > 0$ and if

$$N > (c_e + r_e) / \max(v_s - r_s, p_d (v_d - r_d))$$

(3)
While it is true that since any individual can be followed to acquire their biometric data, equation 3 shows that with fixed risks and rewards if N is sufficiently large there will be a point when it is worth hacking the central DB to obtain them. The above model does not account for modified biometric Phishing or virus schemes, which will only work against a subset of a large DB, but for which the risks and costs are considerably smaller than attacking a well protected DB.

Accurately estimating any of the terms in equations 1-3 is difficult to near impossible, but there are some observations worth noting. Since the values \( v_s \) and \( v_d \) are measured over the total lifetime, probably averaging > 40 years into the future, the expected value grows as more applications use biometrics and as the person who owns the stolen data has more opportunities to sell it. As more and more places deploy biometrics, the cost and risk of hacking a DB continually decrease, and \( (c_s + r_i) \) decreases the necessary size N to flip the centralized biometric risk equation. In general, remote hacking is probably less costly (presuming travel is needed) and less risky than stalking so \( (c_s + r_i) < (c_i + r_f) \), which would make hacking even a small weakly/unprotected DB a better alternative than following an individual. Expecting that anti-spoofing and algorithm performance will improve, \( r_i \) and \( r_f \) can be expected to increase with time but are still per use, not accumulated over the lifetime.

This leads us to the fallacy of non-secrecy, i.e. the fallacious argument that just because biometrics are not secrets, does not mean they should be treated as private nor be aggressively protected. As centralized biometric databases grow, equation 3 shows that eventually there will be a flip in the risk/reward, and there will be a real incentive to violate them, and hence the privacy of the users in that database. Privacy is not about secrets; it is about controlling who has access to “sensitive” data. Thus, while biometrics are not secrets, we need to protect them as well as we protect other “sensitive” data like credit card numbers and social security numbers.

This brings us back to the biometric dilemma. As biometrics become widely used, both their value and venerability increase. Maybe the government biometric databases will be well protected; maybe the shopping databases will be; how about the time-and-attendance, the gym-access or the library? What about organizations that, when strapped for cash, sell their biometric data? Will a high-security facility have its biometric layer of protection neutralized because the records of a local gym were hacked years ago? Will bank accounts be emptied based on illegally purchased biometric records? At least 40 Million “financial records” were compromised or illegally sold in the US in 2005. These records have strong protection requirements, while biometric databases have no such requirements. Realistically it is a question of when, not if, a major biometric database will be compromised or sold. Moreover, when compromises happen, the data cannot be canceled or even effectively “monitored” like a credit report.

1.1 Revocable biometric tokens

We consider the following five properties to be essential to protecting privacy and briefly discuss and justify each. We use the term biotoken to refer to “any technology that converts biometric data into a revocable identity token.” To distinguish our own particular form of revocable identity token we use the Biotope™.
Non-linkable revocability: Transforms biometrics data using multiple keys into a biotoken, such that an individual’s biotokens made with different keys do not match and are not linkable. That is, they are spread through the space as in the normal usage of a biotope. The number of distinct non-matching forms must be extremely large, e.g. number of allowed integers.

Match while Encoded: Are matched in their secure encoded form, without decoding/decrypting.

PK Reissue: Support PKI (Public Key Infrastructure) conversion from the Biotope™ token to the data used to generate it, if and only if the appropriate private key is provided. This is to support reissue without requiring reenrollment.

Cryptographically secure: Provides computationally intractable and cryptographically strong protection from revealing the individual identity, other than matching the specific reference Biotope™ token. The cryptographic security cannot depend on the secrecy of any data used during the Biotope™ generation process.

Non-searchable option: Provide a verification only optional form, whereby an additional factor, possessed only by the individual and never stored separately, is included into the processing such that the resulting Biotope™ can be used for verification only and cannot be used for search or recognition.

The first feature, non-linkable revocability, is the requirement that we need revocable and non-linkable forms. Revocable tokens, with an intractable variety, is important for privacy since it allows the user to have different tokens in each application so no application can link its data (at least via the Biotope™ tokens) with another. If there is only one or a few “keys”, then the controller of the key determines when the data can be revoked, which reduces the privacy/security.

The second criterion, Match while Encoded, is a requirement for privacy for two reasons. First, if the data needs to be decoded/encrypted to match then the raw data will be exposed on each matching attempt. Secondly, without it the owners of the data, or a rogue insider, could still decode the data and sell it or use it for unauthorized purposes. Since one cannot match data subject to standard encryption, it does not provide privacy protection, especially against function creep.

The third criterion is the issue of reissue. At one level, a truly non-invertible form would seem to improve privacy. However, the problem with a biotoken that the company cannot cancel and reissue is that it makes it unlikely the system owners will actually cancel the biotokens and reissue because that will require all the customers to come in for an expensive reenrollment. Furthermore, the company would have to publicly admit the potential breach. In section 5, we introduce a process, based on Public-Key technology, which allows a company to revoke and reissue without bringing in the customers, including generation of a unique biotoken for each use. With unique tokens on each transaction, no matter who captured the data, there could be no impact or reuse. A single-use biotoken would address Phishing attacks as well. If or when biometrics start getting used over the web, Phishing can be expected to become a problem. A standard revocable biotoken will not solve Phishing, unless the person Phished recognizes the event and requests the canceling.

The fourth feature, Cryptographically secure, is obviously important for privacy. However, beyond the obvious, there are three more subtle points that needs to be considered. We argue that non-invertibility is neither necessary nor sufficient to protect privacy. Consider a transform that is not mathematically invertible, e.g. $y_i =$
While formally non-invertible, each point has only a 2-point ambiguity. Anyone that has ever done a cryptogram or puzzle knows that even a moderate amount of ambiguity is easily overcome with a little bit of knowledge or constraints. Even if the transform is formally non-invertible, knowledge of constraints and/or correlations in sets of data can often be exploited to remove ambiguity and hence effectively invert the overall transform. Thus, we can conclude that using a mathematically non-invertible transform is not a sufficient criterion to provide protection. Furthermore, since fingerprint systems are tolerant of moderate error levels even if the ambiguities can never be resolved, the protection may still not be sufficient. A related issue is ensuring the transform provides sufficient mixing to stop direct use. If a transform only makes 10% of the data non-invertible a matcher would probably still accept is since they tolerate missing or spurious features. To see that non-invertibility is not necessary to have protection, one only need consider encryption. A symmetric encryption algorithm, e.g. DES3, would require protection of the key. However, with public key algorithms, such as the well-known RSA algorithm, it is practical to have the algorithm and data necessary to protect data be publicly known yet still be able to recover the well-protected data at some future date. The last part of the cryptographically secure has to do with use of helper data. If knowledge of the helper data greatly reduces security, then like symmetric key encryption, we have simply traded protecting the biometric for protecting the helper data. Thus, we can conclude that a mathematically non-invertible transform is neither necessary nor sufficient to provide protection of data.

The final criterion for true privacy is to provide a non-searchable option. This is at the heart of many privacy issues. Biotokens that provide for recognition or search do not address the privacy issue of the system owners being able to track or identify the user. The issue of function creep was one of the concerns expressed in (ORC 2002). Furthermore if the database is compromised, with the data needed to do the transforms of each subject, the intruders or insiders can still use it to determine doppelgangers. At a minimum, this means the data for “revoking” the transforms, data needed to do the transform, should be maintained outside the system, so if it is compromised the data cannot be used. An even better solution, for those applications where it applies, is to use a verification paradigm rather than a recognition paradigm. However, a traditional verification system can be used for recognition by working through a DB trying to identify an unknown subject against each entry. It may have a lower recognition rate than a system designed for recognition, but it would still allow both tracking and, to a lesser degree, doppelganger detection. Thus, we need a revocable biotoken that does not allow search at all. This can be done by incorporating a passphrase into the biotoken generation process, where the passphrase is never stored. These “verification only” tokens could still be used for many applications, from shopping to physical or logical access applications. For applications that require “duplication detection” during enrollment, e.g. passports can be used for enrollment testing and verification. The enrollment testing DB, which is going to be infrequently used, can be more tightly controlled and keep the keys needed for generation of the revocable tokens in a different server. Then the verification only tokens can then be used for the day-to-day operation. We call this approach a primarily-verification system, which is still considerable better for both
privacy and security than a traditional verification system or even a revocable bioto-
ken-based verification.

1.2 Related work in revocable biometric tokens

Approaches for cancelable or revocable biotokens have been discussed in the
literature, with a review and classification of them presented in (Ratha et al 07).
They divide the field into four categories: Biometric salting, Fuzzy schemes, Biomet-
ric Key generation and non-invertible forms. (Note that our approach does not fit
within any of these categories.) Biometric salting, mixing randomness, e.g. (Cam-
bier et al. 02) (Teoh Jin Ngo Ling Goh 2004) or (Savvides, Kumar, and Khosla
2004) has been tried but has not produced effective systems. The “randomness” must
still be protected since recovery of the original data, given the pad, is generally easy.
The current papers in that area all also require pre-aligned data.

Fuzzy schemes, the best we consider to be (Tuyts, Akkermans, Kevenaar,
Schrijen, Bazen, and Veldhuis 2005), are making progress but still significantly
decrease the effectiveness of the underlying algorithms. The existing work also
presumes the data has been aligned, e.g. using core-delta alignment, which is prob-
lematic. Even given aligned data, the best reported results only increased the Equal
Error Rate of the underlying algorithm by factor of 2 while they embedded a 40bit
key.

A good review of biometric key generation is given in (Uladag et. al. 04). The
idea is a mixture of quantization and encryption of the biometric data to produce a
unique key. The encryption might be based on a user passcode allowing it to be
revocable. This approach has two problems. First if the encryption needs to be in-
vited to match on the original data, then the system will need the user passcode and
convert the data back to original form for matching, hence providing access to the
original biometric data. If the approach does not invert the data, then it must be
matching the encrypted form of the biometric. However, the process of encryption
will transform input such that adjacent items, i.e. nearly identical biometrics, will be
encoded to very different numbers. Given that any biometric has a range of expected
variations for the same individual, either the encrypted biometric will often not
match the individual or the data must be degraded so that all variations for an indi-
vidual map to the same data. However, this would significantly degrade the miss
detection rate. Furthermore, the approach would have to fix the FMR/FNMR rate
that would limit use in different applications. The results in (Uladag et. al. 04) show
a loss of two orders of magnitude in both FMR and FNMR.

In (Ratha et al 07) a collection of sophisticated transforms which are formally
non-invertible for producing “cancelable” fingerprint. However, those have very
limited ambiguity and are not cryptographically strong. In their preferred, “surface
folding” transform, only about 8% of the data changes its local topology, hence only
a small fraction of the data is logically non-invertible. Given the transform, one
could invert data in the non-folded regions and take each point in the folded region,
and send it to both potential locations. Since a fingerprint matcher would likely
match a print with 8% spurious data, we would consider that effectively invertible
and hence not cryptographically secure.
Having introduced the limitations of large-scale biometric databases, the concept of a Biotope, and its difference from other revocable biotokens, we now discuss the computation of Biotopes in a general setting. In section 3, we will look at face-biotope performance and in section 4, finger biotopes.

2 Computation of Biotope

The computation of Biotopes uses feature space transforms based on the representation of the biometric signature, i.e. after all transforms are computed. Most importantly, the transform induces a robust distance/similarity metric for use in verification. In a sense, it is an “add-on” after all the image processing has extracted features. The approach supports both transforms that are public-key cryptographically invertible, given the proper private key, or using cryptographic as one-way functions such as SHA1, which trade less risk of compromise for more effort in reenrollment or transformation, if data is compromised. In either case, even if all transformation parameters and transformed data are compromised, the original data is cryptographically secure, thus removing the risk of reconstruction if centralized Databases are compromised. Obviously, if an invertible version is used, access to the private key plus the transform parameters and data would allow inversion, but that key is not used in the verification process and need not be online at all. As we will see, our preferred embodiment uses PK transforms, to address reissue. If one really does not care about reissue, the private key does not have to be saved, so we will discuss it in a PK context.

Biotope can support an integrated multi-factor verification, wherein the stored data cannot be used for identification (or search), even using the “assume each person and verify” approach. Existing multi-factor approaches store the biometric and other factors separately, verify each and only provide access if all are successful. Our approach stores fused data and neither the biometric nor the added factors are directly stored in the DB. The advantage is the first “verification only” approach; the down side of this is that we cannot change the password without reissue.

In short, the fundamental advances of the approach are provided by a biometric transformation that provides a robust distance-based computation for supporting confidence in verification while supporting revocability, verification without identification, and can have thousands of simultaneous instances in use without the ability for anyone to combine that stored data to reconstruct the original biometric. It can be applied to almost any biometric with minimal changes. We now discuss details of implementing these ideas for face-based biometrics.

2.1 Distance/Similarity Measures and Robust distance computation

For the sake of simplicity in understanding, we initially explain the approach presuming all fields are floating-point numbers. It applies directly to reduced bit representations but proper protection of them requires additional discussion beyond the scope of this chapter. Before we discuss multi-dimensional finite field samples, let us illustrate the idea with a simple biometric signature with one field and we assume, for simplicity of explanation, that the “distance” measure is simply the distance from
Figure 1: Penalty functions used in distance measures. For weighted least squares errors, the penalty is a constant times distance, and grows quadratically. Thus a single outlier significantly impacts the fitting. For a robust similarity metric, the penalty is limited to maximum value so that outliers have a constant, and limited, impact on the overall measure. Given measurements p,q, we can define a robust measure \( m_\theta(p,q) = c \) if \( \text{abs}(\text{r}(p) - \text{r}(q)) > b \), and \( m_\theta(p,q) = (\text{r}(p)-\text{r}(q))^2 \) otherwise.

the probe to the gallery data (i.e. items in the DB) and that the “verification” is then based on threshold of the absolute distance.

A key insight into the approach is that a robust distance measure is, by definition, not strongly impacted by outliers (Huber 1981). In many of the traditional distance measures, e.g. L2, weighted L2 or Mahalanobis measures, the multidimensions penalty for a mismatch grows as a function of distance, thus if the data in one sub-dimension is far off then the penalty is high. In a robust measure, the penalty for an outlier is generally constant. Most fingerprint systems use a robust distance measure and the open-source face systems have them as options as well. (Most commercial face systems do not detail their similarity measures.)

The transforms we defined separate the data into 2 parts, one, \( q \), that must match exactly, basically defining the “window” for the robust computation, and the second, \( r \), which supports the local distance computation. Since \( g \) must match exactly, if it matches, it matches before or after encryption. The mapping hides the actual value and encroits the larger (and hence very stable) part of the position information, \( g \) producing a field \( w \), thus effectively hiding the original positional data and protecting privacy. This key idea was inspired from computation in RSA-type algorithms.

We describe the process for a simple biometric signature with 1 field, and will generalize it later. It is assumed, for simplicity of explanation, that the “distance” measure is simply the distance from the probe to the enrolled data, and that the “verification” is then based on a mixture of the absolute distance. It is assumed that the biometric produces a value \( v \) which is then transformed, e.g. via scaling and translation, \( v' = (v-t)*\delta \). The resulting data is aliased/wrapped back, and without loss of generality, this can be represented with residual \( r = \text{the remainder (e.g. fraction) part of} \ v', \) and general wrapping number \( g \) saying how many times the result wrapped (e.g. the integer part of \( v' \)). The shaded region on the axis of Figure 1 shows an example “residual region” after an appropriate transform and wrapping. A mapping hides the actual value, but as it separates the result, it leaves an unencrypted value within the “window” in which local distance can be computed, and then encrypts the larger (and hence very stable) part of the position information, thus effectively hiding the original positional data. In terms of privacy protection, this is like saying some
Figure 2: Different user transforms being applied to 4 raw data samples

one is \( w.7 \) transformed inches tall where \( w \) is the encrypted set of leading digits. Even if one can make the \( .7 \) back to raw inches, it does not reveal significant information about the individual.

In this overview, the values \( r, w, s, t \) and other representations are used throughout to illustrate certain principles. However, their use is not meant to limit the applicability of these principles, though all examples herein use simple translation and scaling transforms.

Four different transforms, and their effects on 4 data points are shown in Figure 2. The raw positions are shown on the axis/table row on bottom. The first transform, the top line, has a larger “window” size”, which equates to a smaller scaling \( (s) \) and translates 1 unit left. The second example has a larger scaling or smaller “window size” and translates 7 units right. The remaining two examples have the same scaling \( (s) \) but different translations \( (t) \). The table on the right shows the resulting numerical representation of the 4 symbols. Note how, for the last two transforms, the ? symbol wraps directly on top of the + symbol (i.e., their \( r \) values are equal) with only the generalized wrapping number, \( g \), being different. In the first transform the ? symbol aliases on top of the * symbol.

Using this general idea, we describe the actual biotope process including the optional pin, shown overall in Figure 3. The transform and wrapping is computed and the passcode is then fused with the generalized wrapping index, \( g \), before encoding. Example fusion modules could include concatenation, xor or even addition. The inclusion of the passcode provides a strong form of revocation, and protection from its use in search or identification rather than recognition. To ensure that the biometric data is protected even if the “transformation” parameters are compromised, we need to ensure that the mapping from \( g \) to \( w \) is non-invertible or at least cryptographically secure. The “security” of the revocable approach is determined by this transform and the associated Public Key Encryption.

The preferred approach is to use a PK encryption of \( g \) to produce \( w \), e.g. RSA or an Elliptic Curve Cryptography. This allows the system to support user-requested retransformation as part of revocation where the user could receive the database entries, compute the original vector using their private key (which only they have), and then recompute the new transformed data. This can be done without the need for a reacquisition of biometric data and hence without access to the sensors. While we show only one level of encryption, we note that the encrypted \( w \) could be encrypted multiple times with different keys which will be discussed in section 5. An alternative would be to use a one-way hash (e.g. MD5 or SHA). For simplicity we refer to
the transformation \( v \) to \((r,w)\) as encoding and \(r,w\) as encoded data. If a passcode or pin is mixed with \(g\) before encoding, the result is a biotope that is suitable for verification only, which prevents recognition or search, as the user passcode is not stored anywhere.

Now that we have shown how the data is transformed and protected, it is critical to show we can compute distances between the encoded fields directly.

### 2.2 Robust distance computation on the encoded data

Assume for signatures \( p,q \), encoding using \( s,t \) yielding \( r(p), r(q), w(p), w(q) \), we define the robust dissimilarity metric \( d(p,q) \) as follows:

- if \( w(p) \neq w(q) \) then \( d(p,q) = c \)
- if \( w(p) = w(q) \) and \( \text{abs}(r(p) - r(q)) \geq b \) then \( d(p,q) = c \)
- otherwise \( d(p,q) = \frac{(r(p)/s(p) - r(q)/s(q))^2}{b} \)

This distance computation is just one example of a robust distance measure, one that uses a constant penalty outside a fixed window and linear square penalty within the window. The unique property of the mapping ensures that the window around the correct data is mapped to a window in which any robust distance measure can be computed.

Clearly given \( r,s,t \) and \( g \), the original data can be reconstructed. It should also be obvious that many distinct data points will all have the same value for \( r \) and that without knowledge of \( g \), the original cannot be recovered. The biometric store would maintain \( r,s,t \) and \( w \) (the encrypted version of \( g \)). We can consider each of these as user specific functions that can be applied to an input signature, e.g., \( r_k(v) \) is the residual associated with biometric signature \( v \) when using the \( k \)th user’s transform, and \( w_k(v) \) is key \( w \) that results from \( v \) after applying the transform and the encryption associated with user \( k \).
A key issue is the choice of the scale and translation. If we let \( e_{kj} \) be the \( j \)th biometric signature for user \( k \), then we assume \( s_k \) and \( t_k \) are chosen to satisfy the robust window equation:

\[
b s_k < r_k(e_{kj}) < (1-b) s_k \quad \forall \ j
\]

for each field in the signature. Since we are free to choose \( s \) and \( t \) separately for each user and each field and can do so after we have obtained the enrollment data, it is straightforward to satisfy the robust window equation for all enrollment data. For this to be truly effective, the range of values used to determine the scale in Equation 4 should be larger than the actual variations of that parameter for that user, not just over the enrollment data. In practice, we have increased the enrollment range by a factor of 3 to ensure that the actual user’s data is very unlikely to fall outside the scaled window. Even with the described constraints, there are still “infinitely” many choices for \( t \) for “real” numbers and a huge range for floating points. Changing \( t \) impacts both \( r \) and \( g \) and combined with the encryption for \( w \), provides protection of the underlying identity. For finite bit representations, the constraints are more limiting, as is discussed later, but for some values of \( b \) it can be satisfied for any field with more than a single bit.

**Theorem 1:**

If a transform satisfies the robust window equation, and the distance measure has a constant penalty outside the window that is at least as large as any penalty within the window, then computing distances in the encoded space cannot decrease, but may increase, the accuracy of the system.

**Proof:** Given the assumption that all matching user data satisfies equation 4, it is easy to see that \( d(p, e_k) = m_{\text{sh}}(p, e_k) \), i.e. for the matching users, the robust dissimilarity measure applied to the transformed data is the same as the original robust metric applied to the raw data with a robust window of size \((s_k \ast b)\).

For an imposter, \( e_i \), encoded with user \( k \)’s transform two possibilities exist. If \( b s_k < r_k(e_{ij}) < (1-b) s_k \) \( \forall \ j \), for every field within the signature then \( w(p) = w(q) \) and the distances for the imposter \( i \) will be the same before and after transform. Otherwise, scaling/shiftinng has resulted in at least one field distance being equal to \( c \), even though the field was initially close enough that the pre-encoded distance was \(< c \). Since \( c \) is chosen such that it is greater or equal to the maximum distance within the robust window, then for non-matching \( i \neq k \), the transform may increase, but cannot decrease, the distance. \( Q.E.D. \)

For a real biometric with \( N \) dimensions, we treat each dimension separately, so given a raw biometric vector \( V \) with \( n \) elements, we separate the result of the transformation, this time into the residual vectors \( R \), and general wrapping \( G \). Again, \( G \) is transformed to the encrypted \( W \), and the system stores the transform parameters, \( R \) and \( W \). If the system designer usually uses a Mahalanobis transform before distance computation, the covariance transform should be applied to \( V \) before it is transformed. The distance \( D(P, W) \) is then the component sum of \( d(p,w) \) for each dimension. If the biometric has control or validity fields, e.g. which say when particular
fields are to be ignored, they should not be transformed but should be incorporated into the computation of D.

For local verification, the client process requests the particular transforms, R and W and the encryption key PK (which should be digitally signed to avoid man-in-the-middle attacks). If appropriate, it requests the user passcode. It then locally computes the transformation and computes the robust distance D, thresholding to verify the user. For a central verification, the client process requests (or has stored in a smart card) the transform parameters and PK then computes the transformed data R and W, which are sent to the central site (digitally signed) for the distance computation and verification. As in many existing systems, the reported biometric data may also include liveness measures or added fields that are modified versions of a digital signature sent from the central authority to insure that the data is not a replay of old biometric data.

3    Face-based Biotopes™ Performance

As noted earlier, previous work on transformed biometrics did not provide quantitative evaluation of the performance. Our initial work on face-based Biotopes was published in (Boul 2006), with some of the results just briefly summarized here. No previous work on face-based biotokens has included quantitative evaluation of their performance so we cannot compare to other approaches.

A revocable Biotope transform is first determined for each individual and entered in the DB. The transformations were determined from the individual variations within the gallery. We also defined a “group” transform that has shared scaling parameters across the group, which allows “enrollment” using only a single image. For verification, the transform of the claimed identity is applied to the probe data and then compared, using the robust distance measure, with the stored data. We treat identification/recognition as a sequence of verification attempts, apply person k’s transform and then compute the distance to person k’s biotope.

The testing was done by extending the CSU toolkit, (Bolme, Beveridge, Teixeira and Draper 2003). The experiment applied the robust revocable biometric to a gallery of all the FERET data to generate all pair-wise comparisons, and then subsets of that data were analyzed for different "experiments". The standard FERET experiments were done including FAFB, FAFC, DUP1, and DUP2 (Phillips, Moon, Rizvi, and Rauss 2000). The Secured Robust Revocable Biometric consistently outperformed the CSU baseline algorithms as well as all algorithms in the FERET study and all commercial algorithms tested on FERET.

Table 1 summarizes the rank-1 recognition, for CMC graphs see (Boul 06). While in the past PCA has been used as a “straw man” that was not taken seriously, the use of robust revocable transforms make it a viable algorithm. For example, on DUP2, the group revocable scored 86.5% and the individually scaled Biotopes were >90% on DUP2, the best score on the original FERET tests was 52.1%, and highest previously reported score, of which we are aware including commercial algorithm, is still <90%. The dramatic improvement in recognition performance shows the significance of using a robust revocable metric in a revocable biotope. The excellent performance of the group robust algorithm based on PCA combined with the simpli-
fied enrollment processes and fast computation, suggest this algorithm has considerable potential. The EBGM Biotope algorithms were the more effective overall, but also the most expensive algorithms.

An obvious issue for the Group techniques is the definition of the group used for training. There is a subtle issue here of training/testing overlap. The scaling transforms require some data to determine the range of variation. This can be done with multiple images per person. Not all tests had non-overlapping testing/training data. For the listed Group Biotopes, which allow single image enrollment after training, we have tested with different training groups, all 3541 images, DUP1 (243 subjects, 722 total images), FC images (2 each of 194), and the 2 images each of 71 individuals (X2) use to train the FERET PCA space (feret_training_x2.srt from CSU’s toolkit). Note that FAFC has no subjects/images in common with any of DUP1, DUP2 or X2, so the 100% rank 1 recognition is not a training/testing overlap. It is interesting to note that when the training was in fact overlapped, i.e. PCA Group Biotope (FC), the performance was lowered from 100 to 99.48, but this may be caused by the random “offsets” used to define the revocable transforms.

While we demonstrated the performance on recognition, we believe the best use of a face-based biotope is for verification only, because the potential for search with faces is so high given the ease with which a raw image can be obtained from a non-cooperative subject. Here we encounter a different but subtle issue. In verification

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>DUP1</th>
<th>DUP2</th>
<th>FAFB</th>
<th>FAFC</th>
</tr>
</thead>
<tbody>
<tr>
<td># subjects</td>
<td>243</td>
<td>75</td>
<td>1195</td>
<td>194</td>
</tr>
<tr>
<td># Matched scores</td>
<td>479</td>
<td>159</td>
<td>1195</td>
<td>194</td>
</tr>
<tr>
<td># Non-matched</td>
<td>228 K</td>
<td>25 K</td>
<td>1427K</td>
<td>37 K</td>
</tr>
<tr>
<td>PCA L2</td>
<td>33.79</td>
<td>14.10</td>
<td>74.31</td>
<td>04.64</td>
</tr>
<tr>
<td>PCA MahCos</td>
<td>44.32</td>
<td>21.80</td>
<td>85.27</td>
<td>65.46</td>
</tr>
<tr>
<td>LDA IdaSoft</td>
<td>44.18</td>
<td>18.80</td>
<td>70.96</td>
<td>41.75</td>
</tr>
<tr>
<td>EBGM Predictive</td>
<td>43.63</td>
<td>24.78</td>
<td>86.94</td>
<td>35.57</td>
</tr>
<tr>
<td>FERET &quot;BEST&quot;</td>
<td>59.1</td>
<td>52.1</td>
<td>86.2</td>
<td>82.1</td>
</tr>
<tr>
<td>Simple Robust PCA</td>
<td>85.73</td>
<td>85.47</td>
<td>98.32</td>
<td>100.0</td>
</tr>
<tr>
<td>PCA Biotope</td>
<td>90.72</td>
<td>87.18</td>
<td>99.50</td>
<td>100.0</td>
</tr>
<tr>
<td>PCA Group Biotope (all)</td>
<td>86.57</td>
<td>85.47</td>
<td>98.32</td>
<td>100.0</td>
</tr>
<tr>
<td>PCA Group Biotope (DUP1)</td>
<td>85.46</td>
<td>85.47</td>
<td>98.24</td>
<td>100.0</td>
</tr>
<tr>
<td>PCA Group Biotope (X2)</td>
<td>83.80</td>
<td>83.76</td>
<td>97.99</td>
<td>99.48</td>
</tr>
<tr>
<td>PCA Group Biotope (FC)</td>
<td>81.85</td>
<td>82.05</td>
<td>97.15</td>
<td>99.48</td>
</tr>
<tr>
<td>LDA Biotope</td>
<td>90.72</td>
<td>87.18</td>
<td>99.50</td>
<td>100.0</td>
</tr>
<tr>
<td>LDA Group Biotope (all)</td>
<td>88.78</td>
<td>85.47</td>
<td>98.91</td>
<td>100.0</td>
</tr>
<tr>
<td>LDA Group Biotope (x2)</td>
<td>87.95</td>
<td>84.62</td>
<td>98.83</td>
<td>100.0</td>
</tr>
<tr>
<td>EBGM Predictive Biotope</td>
<td>91.27</td>
<td>88.03</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>EBGM Search Biotope</td>
<td>91.27</td>
<td>88.03</td>
<td>100.0</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Table 1: Rank 1 Recognition Rates for Face Biotopes and CSU test algorithms
only, a user passcode is included in the process. However, this cannot be easily tested with traditional verification techniques, because it is unclear how to include the hardening factor introduced by the pin. If we used brute force “guessing”, then the FA rate would simply be scaled by the pin size, e.g. by 1e-7 for a seven digit pin.

Realistically, a pin the user can remember is probably not uniformly distributed. To make comparisons easier with past work, we present the verification only results presuming the intruder has already compromised the pin. Realistically, verification only with a reasonable passcode would reduce the FA rate by 0.00001 or even 0.0000001. Figure 4 shows two verification ROC curves in log-log format because the new algorithms perform so well that the more traditional linear ROC plot is useless. The values of 0 do not fit on a log plot and have been truncated to 1e-5. Note, for the biotope algorithms, the vertical axis of the ROC curves is generally truncated by the sample set size causing a radical jump to the right axis when the data is exhausted.

While the ROC plots have considerable complexity, the overall advantage of the robust revocable algorithms is quite apparent. In general, the EBGM-based Biotopes had the best performance, though on DUP2, the PCA Biotope (PCA) algorithm was slightly better. The very strong performance of the PCA group biotope, which has

![Figure 4: Verification LogLog ROC plots for FAFB](image)

no training/testing overlap and supports single-image enrollments, shows face-based verification only is a viable technique.

We also did experiments to show the revocable transforms do not allow matching/linking across databases. The first self-matching test took 191 subjects in FAFC, and made 25 copies of each image, with the resulting rank one recognition rate being 0.0055 or about 1 in 200 expected from random matching. We also processed the feret_all set (3541 images) and, after enrollment, gave each image of a person a different transform. The resulting rank one recognition was zero on 3 out of 5 rounds and 0.0002 on the remaining two runs. For Verification, the ERR for both test was consistently .9997. Thus, as expected, different transforms for an individual match at random and hence protect privacy.

This section analyzed performance of face-based biotopes showing that with this approach we not only gain the privacy protections of a revocable biotoken, we improved the accuracy of the underlying systems thereby increasing security.

4 Fingerprint Biotope™ performance

We implemented the fingerprint-Biotope by extended the NIST/FBI Bozorth matcher (also called NIST VTB), which is part of the NIST publicly available code base. It is worth noting that the underlying algorithm does NOT require alignment or special features such as core/delta. To date this is the only revocable fingerprint-based biotoken that does not need such data and that can be applied to partial prints. There are at least 2 major aspects of performance, speed and accuracy, which are discussed separately.

During enrollment we require the generation of an RSA key and full PK encryption, which is the most expensive step. For a 380x380 image, enrollment takes approximately 750ms to 3000ms (i.e. 3 second) on a 1.6Ghz Pentium 4 processor depending on the size of the chosen RSA key (512-2048 bits). The break down is 250ms to 2500ms for the key generation and encoding the AES key, 350ms for the minutiae extraction and image processing, and 50ms for the AES encoding and biotope generation. Matching does not require the PK-key generation steps, greatly reducing the time to a total average of 423ms, of which 394ms is for the image processing and 29ms is the biotope generation and matching. The matching is only 8ms more than the time for the standard NIST implementation of the Bozorth matcher on which our Biotope™ is based, a speed decrease of about 2%.

More important than speed, however, is how the Biotope™ process impacts matching accuracy. The natural form of the Bozorth matcher takes as input a minimal minutiae file with x,y,θ,q, where x,y is the location, θ the angle and q the quality. The system converts this to a “pair table”, which stores the distance between the pair, and the angles of each minutia with respect to the line connecting them and the overall orientation of the line connecting them. We apply the transform to the pair table rather than the original minutia, which does produce a larger biotope template. This is done to better address issues of small finite fields. Accuracy is a strong function of the number of minutiae or table size maintained. For the Bozorth algorithm we use the pre-supplied defaults, which allows for 150 minutiae and 10,000 pairs.
For the Biotopes, we limited the table size to keep the Biotope™ storage size below 24K, with an average size of 12K. Limiting was done first using the defaults for pruning on NIST-computed quality of the minutiae but also trying to ensure that each minutiae was included in a few pairs rather than letting the best minutiae take part in all of their pairs. This was done to ensure better spatial coverage. While a few may consider a 12K token large, we believe a little storage is a small price to pay for the security and privacy enhancements of Biotopes. Moreover, a 20k template for every finger of every person in the US is still only around 40GB and fits on a laptop disk.

Using the table-based representations, we made only minimal changes to the matcher code, extending it to handle added columns caused by splitting fields into encoded and un-encoded parts and to test the encoded fields. Because the encoding of the tables and quality pairing can change the number of entries, we added normalization to the scoring.

We compare our finger Biotopes with the underlying NFIS2 Bozorth3 matcher using the same inputs (based on mindctd) on standard datasets. The datasets are from the international Fingerprint Verification Challenge (FVC) 2000 and 2002. Each verification test has 8 images from 100 individual fingers, producing 2800 true matches and 4950 false match attempts. Accuracy is show comparing 2 ROC curves shows in Figure 5. The ROC’s show a significant improvement. To make that improvement quantitative we use ERR, which was one of FVC primary performance metrics. Table 2 shows the percentage improvement in ERR for 2 datasets in each of 2002 and 2002. The finger Biotope scores for FVC2000, these scores would have resulted in it being the 3rd place algorithm overall, and in the top ten in FVC2002.

We note that (Rahta et al 07) provides some performance results for their algo-

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Verification EER Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>FVC '00 db1</td>
<td>30% to .029</td>
</tr>
<tr>
<td>FVC '00 db2</td>
<td>37% to .025</td>
</tr>
<tr>
<td>FVC '02 db1</td>
<td>34% to .012</td>
</tr>
<tr>
<td>FVC '02 db2</td>
<td>30% to .031</td>
</tr>
</tbody>
</table>

Table 2: Finger Biotope Performance

Figure 5: ROC curves comparing our Biotope™ s and the NIST matcher on FVC02 data

rithm; it was tested on a small dataset (181 match scores) internal to IBM, and only in graphical form. Nevertheless, it is clear their approach reduced the verification rate of the underlying algorithm, while our approach significantly improved the underlying algorithm.

5 Biotoken Reissue

We believe it is important that the revocable identity-token can be revoked and a new one reissued without requiring the user to physically interact with the system or provide a new biometric sample. This is critical to both the cost effectiveness of the business and its security. From a cost point of view, it is critical because the cost of enrolling or reenrolling a user in the biometric system is a considerable expense of time both for the user and for the provider. From a security point of view, the cost will therefore need to be balanced with the risk after a potential compromise. Efficient reissue support -- security policies that revoke and reissue on even the slightest chance of a security breach, and policies that regularly revoke the data and reissue - limits the potential impact of an undetected security breach. Such regular cancellation policies not only improve the protection of the data, they also reduce inherent value of attempting to compromise the data, thus decreasing the incentive for anyone to attempt to steal it and reduce the risk of compromise, see equation 3.

In the biotope approach, since we have separated data into parts to be encoded and parts left unencoded we have a particularly simple solution, as described in Figure 6. The encryption step becomes a cascading of encryptions, possibly mixing in company specific “passkeys” beyond the public-key operations. The first transform is with the user’s public-key followed by one or more transforms with the appropriate provider public key(s).

To cancel and reissue within the company, the final provider simply uses their operational private key to recover the master encoded data, and then chooses a new operational PK pair and reencodes using that new private key. The company can

![Diagram](image)

Figure 6: Biotope generation followed by multi step PK encoding to provide easily reissued biotopes. Data in dotted boxes is not stored.
then have a security policy that reissues biotopes on a regular basis, just to ensure
attacks, even if undiscovered, do not compromise the data for any serious amounts of
time.

If the user wants to reissue, either a new company or to use a new transform for
the original company, they can request the company use their private keys to provide
the “first PK encoded form” of Figure 6 (wrapped appropriately in a signed en-
rypted email). The user then decodes this, using their private key, to recover the
original data. From the original data, they can apply a new transform, reencode with
their public key, and then apply the appropriate company’s master public key. The
advantage of this embodiment is that the data is never in long-term storage in a form
that is not cryptographically protected, and either the user or the company can easily
revoke and reissue without the need for rescanning the biometric data.

Finally, since data can be mixed into the encoded streams before reencoding,
just as we did with the user’s passcode, other data can be added to increase the secu-
rity and privacy of the system. In the most extreme form, which is what we believe is
the important case, we can take the operational biotope and add another stage were
we take a transaction ID, mix it with the encoded data then provide an added encod-
ing for the transaction level. The central site can produce such a transaction-specific
biotope and send all the data to the biotope generation device for comparison, or the
biotope generation device can get all the parameters and send the transaction-specific
token back to the central server for verification. Either way the transaction specific
biotope for one transaction will never be reused.

6 Conclusions and future work

While biometrics are not secrets, this paper introduced the biometric risk equation as
a formalism for analysis of the risks associated with not properly protecting biomet-
ric data. Large biometric databases are an accident waiting to happen and the bio-
metric dilemma is that wide spread use today may compromise the potential value
for the future. The paper discussed the ideal properties for revocable biotokens and
introduced a robust revocable biotoken called Biotopes™.

This paper showed the effectiveness of Biotopes™ on face-based biometrics and
fingerprints. The transforms are applied to biometric template data to produce two
components one of which is encrypted while the other is stored unsecured. The
transforms combined with encryption maintain the privacy while the unencrypted
part supports a robust distance measure, something that is critical to make biometrics
effective. While the paper presents only face and fingerprint, the approach applies to
an even wider range of biometrics.

While Biotopes™ are a good start, there is a lot of work to do in terms of infra-
structure to support these novel approaches, cryptographic analysis to ensure the
integrity and further adapting the approach to a wider class of algorithms. But with
the major biometric vendors having a vested interested in protecting their market
share, and the government focuses on interoperability and the ability to share data
across all biometric databases, it will take more than just science and engineering to
get any revocable biotoken solution adopted.
References


ORC International 2002. “Public attitudes toward the uses of public attitudes toward the uses of biometric identification biometric identification technologies by government technolo
gies by government and the private sector and the private sector”.


