# Generation of Replaceable Cryptographic Keys from Dynamic Handwritten Signatures

W.K. Yip[1,2], A. Goh[2], David Chek Ling Ngo[1,2], and Andrew Beng Jin Teoh[1,2]

[1] Faculty of Information Science and Technology (FIST), Multimedia University, Jalan Ayer Keroh Lama, Bukit Beruang 75450, Melaka, Malaysia
{yip.wai.kuan04, david.ngo, bjteoh}@mmu.edu.my
[2] Corentix Technologies Sdn Bhd, B-S-06, Kelana Jaya, Petaling Jaya, 47301 Selangor, Malaysia
alwyn@corentix.com

**Abstract.** In this paper, we present a method for generating cryptographic keys that can be replaced if the keys are compromised and without requiring a template signature to be stored. The replaceability of keys is accomplished using iterative inner product of Goh-Ngo [1] Biohash method, which has the effect of re-projecting the biometric into another subspace defined by user token. We also utilized a modified Chang et al [2] Multi-state Discretization (MSD) method to translate the inner products into binary bit-strings. Our experiments indicate encouraging result especially for skilled and random forgery whereby the equal error rates are <6.7% and ~0% respectively, indicating that the keys generated are sufficiently distinguishable from impostor keys.

## 1 Introduction

In authentication systems, it is well known that password and public-key systems do not physically associate the user hence, identity frauds can be easily carried out. Therefore, there is a need to incorporate biometric factor (what you are) for authentication to provide better security. In this paper, we are interested in using dynamic hand-signatures as the biometric features because they are socially and generally well-accepted and are more cost effective in terms of capturing equipment (eg. PDAs, smartphones and mouse-pen). In particular, we are interested in deriving bit-strings from dynamic hand-signature data to be used as cryptographic keys in authentication protocols. The following issues are addressed in this paper: (1) biometrics is not exactly reproducible, (2) non-revocability of biometrics in that they are permanently associated with the users, and (3) non-secrecy nature of the biometric. Our solution to (1) is to use a modified MSD with Gray encoding to allow keys to be encoded as closely as possible within a permissible threshold bounded by the statistical deviation. Issue (2) is resolved using iterative inner product that causes the biometric feature to be projected into another random subspace dictated by the stored user random token which is an independent factor from the biometric. Lastly, the fact that our key statistics are linked to the mixed biometric with token randomness, and the inherent one-way transformation of the iterative inner product, guarantee the non-revelation the actual biometric even if the final keys are stolen.

D. Zhang and A.K. Jain (Eds.): ICB 2006, LNCS 3832, pp. 509 – 515, 2005.

## 2 Previous Works

The first biometrics hash on dynamic hand-signature was proposed by Vielhauer et al [3] which used a 24-feature-parameter set from dynamic hand-signature and an interval matrix, which stores the upper and lower threshold permissible for correct identification. Although the authors reported that the system has FAR 0% and FRR of 7.05% achieved for only 11 test subjects, it is not clear if the performance will be the same for a larger sample set. Similarly, Feng-Chan [4] also uses 43 features (but not all are published) and reported 8% EER but the uniqueness of the output vector is only 1 in $2^{40}$, which is insufficiently long for cryptographic usages. Another scheme for face data, Chang et al [2] also uses user-specific boundaries information. The keys are generated from the biometric (permanent association) and hence, if compromised, the user needs to create a new biometric which is not feasible. This shortcoming is also observed in Davida et al [5] method of using error-correction codes directly on iris features.

Cancelable keys can be achieved by incorporating random tokens as in Soutar et al [6], Monrose et al [7-8], Juel-Wattenberg [9], Juels-Sudan [10] and Clancy et al [11]. Schemes [6-8] which utilized lookup tables, and [10-11] which require storing quite substantial number of additional chaff points, are not storage-efficient while Juel-Wattenberg is subjected to multiple key attack [12]. On the other hand, Goh-Ngo [1] is storage efficient, as only a randomized token is required, and is a secure one-way transform as the inner product cannot be reversed to recover the actual biometric.

For feature extraction, although most hand-signature verification methods [13-17] have reported the successful use of dynamic time warping (DTW) whereby the test signal is non-linearly aligned to a template signal, it is not suitable for our application due to the open storage of template. Another approach more suitable for our application is to process the signal using Fourier transformation as in Martinez et al [18] and Chan-Kamins [19]. We choose this approach as feature extraction as no template is needed and there exist a Fast Fourier Transform (FFT) that executes in n.log(n) time compared to common approaches DTW and linear discrimination methods that require at least $n^2$ computation.

## 3 Proposed Method

We adopt similar strategy as with Chan-Kamins [19] for feature extraction using FFT but using different combination of the dynamic signals derived from the input positional signals from the user devices. Our method then combines the Goh-Ngo iterative inner product step of mixing random token with the biometric data, with the discretizing scheme of Chang et al [2] to product multiple bits for each feature element, as outlined in Fig 1.

We assume a stylus-enabled PDA for capturing the signature in (x,y) coordinates and a timestamp (t) for each point. The signals are then pre-processed by cubic spline interpolation to derive the velocity (x1,y1) and acceleration (x2,y2), and re-sampled to obtain uniform signals length of 512 which is required for the optimum computation efficient for FFT. Finally, the signal is aligned to the origin by subtraction from the centroid.

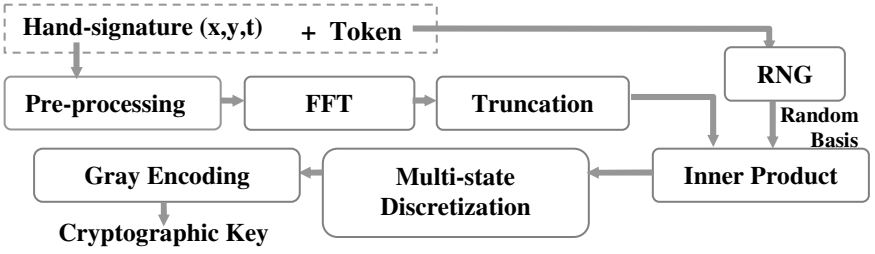


**Fig. 1.** Outline of the proposed method

Each signal $x$ can be represented by a Fourier integral of form (implemented using the FFT algorithm) as $x(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega$ with $j = \sqrt{-1}$. In general, the Fourier transformation is a complex-valued function of $\omega$ and can be expressed in rectangular form as $X(e^{j\omega}) = X_R(e^{j\omega}) + jX_I(e^{j\omega})$ and in the polar form as $X(e^{j\omega}) = |X(e^{j\omega})|e^{\angle X(e^{j\omega})}$ where $|X(e^{j\omega})| = \sqrt{X_R{}^2 + X_I{}^2}$ is the **mag**nitude and $\angle X(e^{j\omega}) = \tan^{-1}\frac{X_I}{X_R}$ is the **ang**le of the Fourier transform. Individual truncation by extracting the 20 most significant amplitudes of the Fourier transforms, followed by concatenation of the various truncated transforms (discussed in Section 5) is computed to obtain a compact biometric vector $B \in \mathbb{R}^n$.

The biometric-token mixing stage (Table1) involves iterative inner products of biometric feature B and random basis vectors defined by user token (T). The random basis are orthonormalized vectors (using Gram-Schmidt algorithm) generated from a random number generator (RNG) eg. X917, that follows the Gaussian distribution of zero mean and unit standard deviation based on T as the seed.

At enrollment, the boundaries are specified for each feature element $D_i$:

- left boundary ($LB_i$) = min($m_{glo,i} - k_{glo}.s_{glo,i}$, $m_{usr,i} - k_{usr}.s_{usr,i}$) and
- right boundary ($RB_i$) = max($m_{glo,i} + k_{glo}.s_{glo,i}$, $m_{usr,i} + k_{usr}.s_{usr,i}$)

**Table 1.** Iterative inner product and discretization steps

| P := **inner product**(T, B): | D := **discretize** (P): |
|---|---|
| 1. for i = 1..n-1<br>2. $r_i = \{RNG(T)_j\}_{j=1+(i-1).n}^{i.n}$<br>3. end for<br>4. orthonormalize $\forall r_i$<br>5. for i = 1..n-1<br>6. $P_i = \langle r_i, B \rangle$<br>7. end for | 1. for i = 1..n-1<br>2. $D_i = \text{gray}\left\lfloor \frac{P_i - LB_i}{w_i} \right\rfloor$<br>3. end for |

with m being the mean, s being the standard deviation and k a configurable parameter, subscript glo denoting population-wide norms and usr denoting user-specific (in our case, trained on 10 reference signatures) norms. The correct user state is within the region of $m_{usr} \pm k_{usr}.s_{usr}$. We specify only the $LB_i$ and width, $w_i$ ($=2.k_{usr}.s_{usr,i}$) of each $D_i$ for storage. To guarantee that the Hamming distances between consecutive states are 1, we modify the original algorithm to encode the index of the state using Gray code ie. distant states will have higher Hamming distances compared to states that are nearer to the authentic state. The discretization algorithm proceeds as shown in Table1.

## 4 Experiments and Discussion

The proposed method was tested on the Signature Verification Competition 2004 (SVC2004)[20] Task 1 database consists of 40 users with 20 genuine and 20 skilled forgery samples. The two error rates that we are interested to measure in our experiments are for: (1) skilled forgery where a non-genuine user replicates the genuine signature by imitation and (2) random forgery where a non-genuine user uses his signature. The different dynamic features from the positional, velocity and acceleration are then combined to extract longer bit-strings from (1) 80-feature vector V1= [mag(x),mag(y),mag(x1),mag(y1)], (2) 120-feature V2= [mag(x),mag(y),mag(x1), mag(y1),mag(x2),mag(y2)], and (3) 160-feature V3=[mag(x),mag(y),mag(x1), mag(y1),mag(x2),mag(y2),ang(x),ang(y)]. Fig 2-4 show the effect of varying $k_{glo}$ and $k_{usr}$ on the different combination of dynamic features used between the genuine and skilled forgery without any mixing of tokens. The optimal configuration is observed when $k_{usr}$=1.5 and $k_{glo}$=20. Using V1 provides the best result but shortest key length.

Using the optimal configuration found earlier, we obtain the results in Table 3 for the cases of using **no token**, forged signature with **stolen** token and with adversary own (**substitution**) token. The consistency of random forgery case EER2~0% confirms the clear separation between the genuine and random forgery distribution while

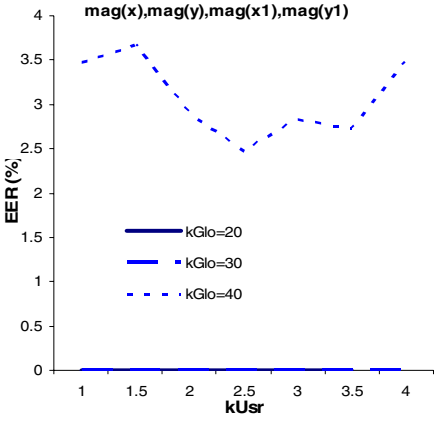


**Fig. 2.** Various $k_{Glo}$ and $k_{Usr}$ settings on V1

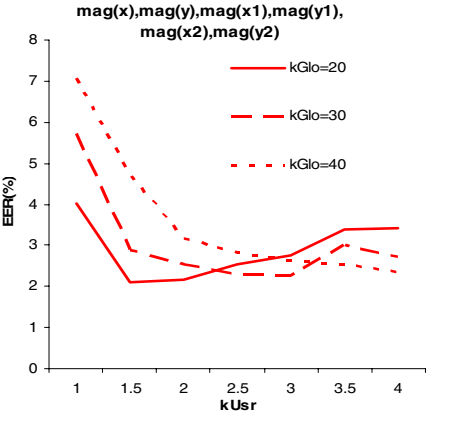


**Fig. 3.** Various $k_{Glo}$ and $k_{Usr}$ settings on V2

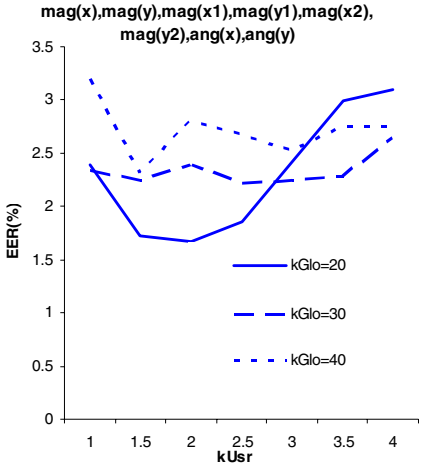


**Fig. 4.** Various $k_{Glo}$ and $k_{Usr}$ settings on

**Table 3.** EERs (in %) for no token, stolen token and substitution token case

| Feature Vector | | V1 | V2 | V3 |
|---|---|---|---|---|
| Bits, N | | 320 | 480 | 640 |
| No Token | EER1 | 0.00 | 2.44 | 1.61 |
| | EER2 | 0.00 | 0.00 | 0.00 |
| Stolen Token | EER1 | 4.25 | 6.64 | 4.66 |
| | EER2 | 0.00 | 0.00 | 0.00 |
| Substi-tution-Token | EER1 | 0.02 | 0.05 | 0.06 |
| | EER2 | 0.00 | 0.00 | 0.00 |

results of skilled forgery EER1 <6.7% are also encouraging. The good separation for substitution token forgery is because each (including skilled forgery) user performs his own token-governed transformation, and hence the keys are all projected to different random subspaces, resulting in different keys. For stolen token scenario, the skilled forgery cases have higher EERs because similar signature vectors are projected onto the same subspace using the stolen token. We will discuss the security implication of these results in the Section 6.

## 5 Security Analysis and Discussion

### 5.1 Exhaustive Search

In this case, we assume that the attacker has no knowledge of the random token, key statistics or signature. Let N be the effective size of the bit-strings generated in our experiment. Using brute force attack, $2^N$ number of attempts is required.

### 5.2 Known Key Statistics Attack

In this scenario, the attacker has access to the statistical parameters used in MSD. The number of guesses he can make for each feature element is given by $\sum_{i=1}^{n-1} \frac{RB_i - LB_i}{w_i}$. For our experiment, each element can be represented by an average of 4 bits, hence the number of attempts made will be at least $2^{4(n-1)}$. It must be noted though the key statistics are not reflective of the actual biometric features but on the inner product values, hence are replaceable and will not pose a permanent security risk.

### 5.3 Stolen Token Attack

This is the case of an adversary using **stolen genuine token** and **forged signature** of the genuine user. The EER1 of <6.7% shows that it is of comparable performance to existing protocols. The best result is achieved using V1 signature features which provided EER1=4.25. In fact, our proposed scheme has a longer key length of effective bits as compared Feng-Chan (43 elements) and Vielheur et al (24 elements).

### 5.4 Substitution Token Attack

This is the case of an adversary using his **own token** and **forged signature** of the authentic user. EER1=0% in Table 2 confirmed that the scheme is extremely unyielding against such attacks.

In summary, the key security advantages our solution provides are (1) longer key space, (2) good separation between the genuine and skilled forgery curve and (3) perfect separation for the random forgery case. Another important improvement of our scheme as compared to DTW-based approaches is the non-requirement of template storage which could deter an adversary to reproduce the signature without even the actual signing action.

## 6 Concluding Remarks

Our experimental results have established that the proposed method of combining random token and biometric data is able to generate sufficiently long and distinguishing bit-strings. In particular, we have found that the method is comparable with existing schemes even for the more difficult case of a skilled forger using an authentic token. The use of MSD using user key statistics provides the error tolerance to accommodate intra-signal differences. By incorporating randomness via the iterative inner product, the keys generated are replaceable thus providing better management. The one-wayness of the inner product mixing, and the key statistics which are based on the token-based projected biometrics (instead on the plain biometrics) ensure that the biometric features are not compromised even if multiple keys stolen.

## References

[1] Goh, A. & Ngo, D.C.L.: Computation of Cryptographic Keys from Face Biometrics, Seventh IFIP TC-6 TC-11 Conference on Communications and Multimedia Security, Springer-Verlag LNCS 2828 (2003)

[2] Chang, Y.C., Zhang, W. & Chen, T.: Biometric-based Cryptographic Key Generation, IEEE Conference on Multimedia and Expo, Taiwan (2004)

[3] Vielhauer, C., Steinmetz, R. & Mayerhorf, A.: Biometric Hash based on Statistical Features of Online Signatures, Proc. of the 16th Intl. Conference on Pattern Recognition (2002)

[4] Feng, H. & Chan, C.W.: Private Key Generation from On-line Handwritten Signatures, Information Management and Computer Security, MCB UP Limited (2000) 159-164.

[5] Davida, G., Frankel, Y., Matt, B.J. & Peralta, R.: On the Relation of Error Correction and Cryptography to an Off Line Biometric Based on Identification Scheme, WCC99, Workshop on Coding and Cryptography (1999)
[6] Soutar, C., Roberge, D., Stoianov, A., Gilroy, R. & Kumar, B.V.K.V.: Biometric Encryption Using Image Processing. SPIE 3314 (1998) 178-188
[7] Monrose, F., Reiter, M.K., Li, Q. & Wetzel, S.: Cryptographic Key Generation from Voice, Proc. of the 2001 IEEE Symp. on Security and Privacy (2001)
[8] Monrose, F., Reiter, M.K., Li, Q., Lopresti, D.P. & Shih, C.: Toward Speech-Generated Cryptographic Keys on Resource Constrained Devices, Proc. of the 11th USENIX Security Symposium (2002)
[9] Juels, A. & Wattenberg, M.: A Fuzzy Commitment Scheme, in Proc. 6th ACM Conf. Computer and Communications Security, G. Tsudik, Ed. (1999) 28-36
[10] Juels, A. & Sudan, M.: A Fuzzy Vault Scheme, in Proc. IEEE Int. Symp. Information Theory, A. Lapidoth & E. Teletar, Eds. (2002) 408
[11] Clancy, T.C., Kiyavash, N. & Lin, D.J.: Secure Smartcard-based Fingerprint Authentication, in Proc. ACM SIGMM 2993 Multimedia, Biometrics Methods and Applications Workshop (2003) 45-52
[12] Boyen, X.: Reusable Cryptographic Fuzzy Extractors, 11th ACM Conference on Computer and Communications Security (CCS 2004), ACM Press (2004) pp. 82-91
[13] Sakoe, H. & Chiba, S.: Dynamic Programming Algorithm Optimization for Spoken Word Recognition, IEEE Trans. on Acoustics, Speech & Signal Processing, Vol.ASSP-26, No.1 (1978)
[14] Hastie, T. & Kishon, E.: A model for Signature Verification, AT&T Bell Laboratories Technical Report (1992)
[15] Kholmatov, A.A.: Biometry Identity Verification Using On-line and Off-line Signature Verification, Master of Science Thesis, Sabanci University (2003)
[16] Plamondon, R. & Srihari, S.: On-line and Off-line Handwriting Recognition: A Comprehensive Survey, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1 (2000)
[17] Feng, H. & Chan, C.W.: Online Signature Verification using a New Extreme Points Warping Technique, Pattern Recognition Letters 24 (2003) 2943-2951
[18] Martinez, J.C.R., Lopez, J.J.V. & Rosas, F.J.L.: A Low-cost System for Signature Recognition, Int. Congress on Research in Electrical and Electronics Engineering (2002)
[19] Chan, F.L. & Kamins D.: Signature Recognition through Spectral Analysis, Pattern Recognition, Vol. 22, Issue 1 (1989) 39-44
[20] SVC 2004: First International Signature Verification Competition, http://www.cs.ust.hk/svc2004/