Simultaneous Biometric Verification and Random Number Generation

V. Chandran and B. Chen
Queensland University of Technology, GPO Box 2434, Brisbane QLD 4001, Australia
v.chandran@qut.edu.au

Abstract **A new method of simultaneous biometric verification and generation of a random number for use in authenticated encrypted communication is described. A non-linear transform is applied to a vector derived from a biometric. The output of the same dimension is fed back and the process iterated. At each iteration, the magnitude and phase of a scalar complex-valued inner product of the vector and its displacement from the previous iteration is extracted. It is shown that this product tends towards a certain limit along a trajectory in the complex plane. Both the limit and the trajectory depend on the initial condition which is the biometric vector. For close initial conditions the trajectories will initially remain close but separate exponentially. Magnitude and phase on the trajectory are converted into binary matrices. Entropy criteria are used to weight bits and verify identity. High entropy bits are selected for random number generation. The method was tested using 3D facial images from 61 persons in the Face Recognition Grand Challenge (FRGC) database. An Equal Error Rate of 15.5% was achieved and random numbers of length 512 bits could be generated that satisfied standard tests for randomness. The method can be further developed to generate private keys from low intra-class entropy bits and session keys from the unconditionally random bits on presentation of a biometric without the need to store them.**

I. INTRODUCTION

A number of biometric based encryption methods ([1-3] and references therein) and chaos-based random number generation methods ([4,5] and references therein) have been proposed in the recent past. In this paper, a method for simultaneous biometric feature extraction and random number generation that combines these concepts is presented. The method is validated through verification tests and randomness tests.

For cryptographic applications the output of random number generators and pseudo-random number generators (PRNG) are considered to be strings of zeroes and ones. Each bit produced should ideally appear as the outcome of an independent random event with probability 0.5.

A biometric (for example, face image) acquired using a sensor (CCD camera) usually contains several random noise-like contributions from environmental conditions and presentation (illumination, pose, facial expression). Therefore, it is capable of generating a random number provided the entropy present is distilled appropriately [12]. This may be used directly or as the seed for a PRNG to generate a longer pseudo-random sequence.

Essentially the difference between a random bit and a discriminating bit is that the former has high entropy both within the class and outside it, while the latter must have low entropy within the class. A method is therefore developed that reduces the biometric input into a large matrix of bits in a manner such that discriminating information is retained and entropies can be computed and used. To distil the entropy, a nonlinear system close to chaos and exponentially sensitive to initial conditions is devised.

II. APPROACH

The approach makes use of the following facts:

a) A nonlinear system in chaos exhibits an evolving output that is exponentially sensitive to the initial condition.
b) A discrete-time nonlinear system can be designed to take a vector (derived from the raw or processed biometric) as its initial condition. If the system is iterative (output fed back), it will evolve in time.
c) The output of such a system at any iteration will depend only on the transformation (T) and the initial condition ($x_0(n)$, the biometric vector). A quantization loss may be incorporated into the procedure to improve randomness and irreversibility.
d) A complex-valued scalar quantity can be extracted from the output at any iteration and tracked. The magnitude and phase of this quantity, represented in binary form, will yield a matrix of bits, $P \times R$, where P is the precision to which each quantity is represented and R is the number of iterations.

By virtue of c, it should be possible to extract a feature vector of bits containing information about $x_0(n)$ from the above matrix. If the system is close to chaotic (sensitive to initial conditions), then it should also be possible to generate a random number from the above matrix by appropriately choosing bits.

III. ITERATED INTEGRATED BISPECTRUM

The biometric must first be converted into one or more vectors of real or complex-valued numbers. It is preferable to incorporate desired intra-class invariance properties in this procedure. Facial images, for example, can be normalized, aligned and then expanded into a one-dimensional vector in a row major or column major fashion. Alternately, feature vectors can be extracted and used or Radon projection vectors [7] at different angles can be used.

Let $x_0(n)$, $n = 0,1,2,...., N-1$, where the subscript 0 refers to it being the initial input to an iterative system, be such a vector.

The input vector is normalized by the maximum of the magnitudes of the elements of the vector and the mean value is removed. Let $x_i(n)$ denote the input to the system at the i-th iteration. Let $X_i(k)$ be the N-point DFT of $x_i(n)$. The magnitude spectrum, $|X_i(k)|$ is computed and only the positive frequency half is retained. It is zero padded to length *N*. Discarding the Fourier phase makes the process irreversible for mixed phase signals, and results in a controlled information loss improving unpredictability and randomness. The resulting sequence $y_i(n) = \begin{cases} |X_i(n)|, n = 1,2,..., N/2-1 \\ 0, \quad n = N/2,...N-1 \end{cases}$ is real-valued and the imaginary part is set to zero. It is again Fourier transformed and the deterministic bispectrum ([6,7] and references therein)

$$B_i(k_1, k_2) = Y(k_1)Y(k_2)Y^*(k_1 + k_2) \tag{1}$$

is computed. The bispectrum retains phase information from the signal unlike the power spectrum and it is sensitive to asymmetry (or irreversibility with respect to the time axis) of the signal. $B_i(k_1, k_2)$ is complex-valued in general with non-zero imaginary components [6,7]. The bispectrum is integrated along radial slices in the bifrequency plane,

$$V_i(a) = \int_{k_1=0+}^{1/(1+a)} B_i(k_1, ak_1)dk_1 \tag{2}$$

where $a = \frac{1}{N}, \frac{2}{N}, ..., 1$.

Frequencies are normalized by the Nyquist frequency (one half of the sampling frequency). The zero frequency component (or average signal) is eliminated from the above computation and a is the slope of the line in bifrequency (k_1, k_2) space along which the integral is computed. The bispectrum is bilinearly interpolated to compute the integral in equation 2 as a summation. This procedure has been used for feature extraction and described in reference 6. It has not been iteratively applied in the manner described below previously.

The integrated bispectrum is fed back to the system as a complex-valued vector of length N for the next iteration,

$$x_{i+1}(n) = V_i\left(\frac{n}{N}\right) \tag{3}$$

Let the entire transformation be represented by *T*. Then

$$x_{i+1}(n) = T[x_i(n)] \tag{4}$$

and the output after R iterations is related to the initial input, the biometric data, as

$$x_R(n) = T^R[x_0(n)] \tag{5}$$

When the system described above is subject to many iterations, $x_i(n)$ as $i \to \infty$ will tend towards a complex sequence which is the integrated bispectrum of the positive frequency portion of its own magnitude spectrum. The normalization step guarantees that this limit is not the zero sequence and forces the system to be BIBO stable.

After each iteration, a measure of the change is extracted as follows. The complex valued inner product of the difference between the previous and present outputs with the previous output is taken.

$$D_i(n) = \sum_{n=0}^{N-1}[x_{i-1}(n) - x_i(n)]x_{i-1}(n) = M_i \exp(j\phi_i) \tag{6}$$

Since the input vector is normalized by the maximum of the magnitudes of its elements on each iteration, the difference is prevented from tending to zero. It is observed experimentally that the measure increases for the first two iterations as the input moves from zero imaginary part to non-zero imaginary part, and then the magnitude M_i tends towards a finite limit. The phase ϕ_i can exhibit limiting behaviour or fluctuate depending on the input vector and its length. The limit of the magnitude is dependent on the initial biometric vector.

The trajectory of $D_i(n)$ with iterations i can be plotted in a polar coordinate system and as expected, it is observed to be sensitive to the input biometric vector. To illustrate the principle, 1024 sample sequences from the file "gong.wav" in Matlab are used. Trajectories are shown for 3 different sequences in figure 1. Sequences that are just 2 samples (starting at sample 13022 and 13024, respectively) apart show close (green) trajectories, while the third sequence which is 10,000 samples away (starting at sample 23024) exhibits a significant deviation. The logarithm of the magnitude is plotted in the figures below in order to visualize them better and therefore the maximum value of unity actually maps to the origin. Each trajectory starts there and then branches out to a limit as shown. The limits, although globally "close" are in fact distinct.

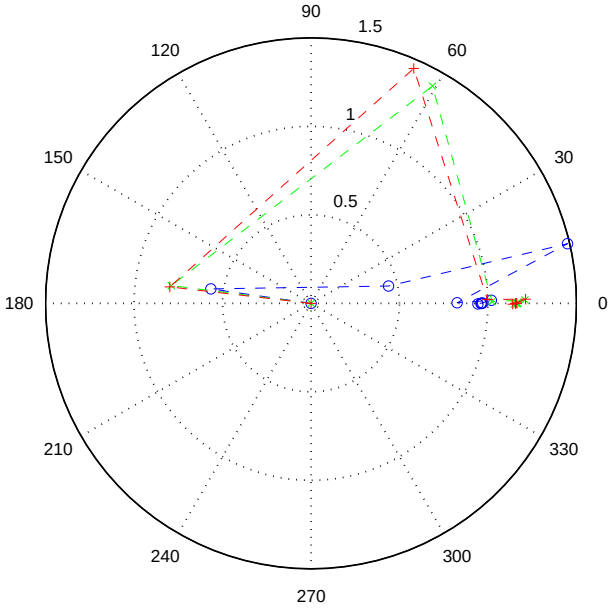

Figure 1. Log-magnitude vs angle trajectories of $D_i(n)$ for three different 1024 sample vectors from the Matlab sound file "gong.wav". The close trajectories are for windows 2 samples apart, while the significantly different trajectory is from a window 10,000 samples away.

For these inputs, the phase angles tend towards $\pm\pi$ very rapidly. If the window length is reduced to 768 samples, however, the phase angle does not tend to a limit and instead decreases exponentially. The transition in behaviour was observed at greater than 842 samples and surprisingly this seemed to be independent of the input!

Log-Magnitude and phase plots are compared in figures 2 and 3. If figure 1 be considered a "phase" plot in dynamical system terminology, it can be observed from figures 2 and 3 that one state variable is bounded (the Log-magnitude) while the other (angle) is changing exponentially when the vector dimension is 768. This is a characteristic of chaotic systems [4,5,8].

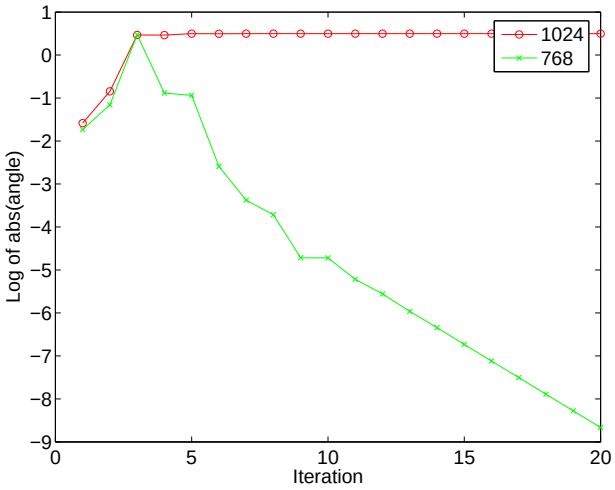

Figure 2. The logarithm of the absolute value of the phase of $D_i(n)$ vs iterations for N=1024 (upper) and N=768 (lower) long vectors from "gong.wav".

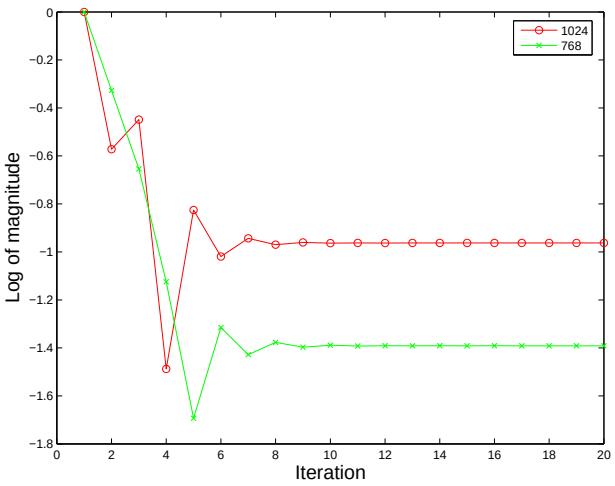

Figure 3. Log-magnitude of $D_i(n)$ vs iteration for N=1024 and N=768 sample vectors from "gong.wav" . The limit is input dependent.

The sensitivity of trajectory can be exploited to generate random numbers by expanding the magnitude and phase into a binary representation at each iteration such that a matrix of bits is produced and then selecting bits based on entropy. Similarities between the trajectories will also result in some bits exhibiting low entropy within a class such as biometrics acquired from the same person but under different acquisition conditions. These bits can be used for verification of identity or generation of a private cryptographic key which need not be stored on a system.

IV. BIT MATRICES

Magnitude and phase of $D_i(n)$ can be expanded into binary representations at each iteration generating a bit matrix in a number of ways. The magnitude is bounded and finite because of the normalization of each input. It has a maximum value of unity because of the normalization after a trajectory is extracted. The phase is bounded by $\pm\pi$ when not unwrapped.

The logarithm to base 2 of the magnitude and angle values are stored as signed numbers to sixteen significant decimal digits from a 64 bit floating point representation. The exponent is normalized and the mantissa is adjusted. The decimal point is removed from the significand and the 16 digit integer is converted into a 60 bit binary value with zero padding in front of the MSB if it is of insufficient length. These binary sequences are stored in two matrices, one each for the angle and the magnitude. Different bits in the matrices have different probability distributions over any class of inputs.

V. TESTS WITH FACE IMAGES

Data collected for the Face Recognition Grand Challenge (FRGC) [9,10] conducted by the National Institute of Standards and Technology and the University of Notre Dame, USA, was used to test the performance of the method. The face images in this database are of two types – intensity and range. Only the range (from the 3D) images were used in these tests. All 61 persons in the database for whom more than 20 images were available were chosen. The images were acquired in weekly sessions over a period of one and half years in 2002-4. The images are aligned and normalized using eye to eye distance and eye positions and represented in 150x130 pixel frames (figure 4).

The Radon transform is used to reduce the initial input length of 19,500 (vector form of a 150x130 image) into 6 separate 203 length vectors at angles in multiples of 30 degrees between 0 and 180. A circular mask of radius 60 was applied to each image prior to performing the Radon transform.

Each of the separate 203 length vectors is passed through the iterated transform over 25 iterations producing two sets of 6 separate output matrices (one for angle another for magnitude).

These are then appended together to form two overall matrices for the image. These two matrices are then converted into binary form with 9000 bits in each.

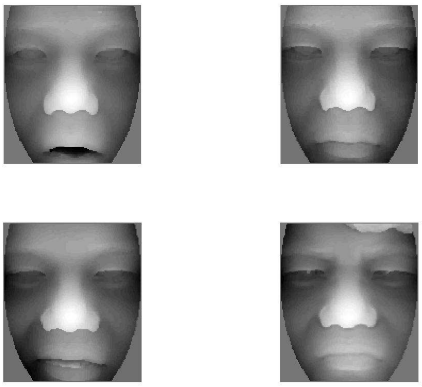

Figure 4. 3D face (range) images before circular masking.

A. VERIFICATION

Entropies of the bits are then calculated to weight bits for verification and select bits for random number generation. To train the system both the intra-class entropy and the extra-class entropy (outside of that class) of each bit must be calculated for each person. The expected values of the bits for each class must also be determined. These entropies are used to rank the usefulness of each bit.

TABLE 1. BIT PROPERTIES AND ENTROPY.

Desirable Property	Entropic Representation
1. The bit is constant for this user and also constant over all other users BUT is of the opposite value.	Low entropy intra-class Low entropy outside-class but opposite expected value (very useful for verification)
2. The bit is constant for this user and is random over all other users.	Low entropy intra-class High entropy outside-class (useful for verification)
Undesirable Property	
3. The bit is random for this user and is constant over all the other users	High entropy intra-class Low entropy outside-class (highly unlikely in practice)
4. The bit is random for all users	High entropy intra-class High entropy outside-class (not useful for verification but useful for RNG)
5. The bit is constant for all users	Low entropy intra-class Low entropy outside-class (not useful)

Each bit is given a weight between 0 and 1, calculated using functions based on extra- and intra- class entropies for that bit. The weight is a representation of how valuable the bit is in identifying this specific user.

The weight for each bit can be broken up into two parts:

1. Intra-class weight of the bit – From table 1 it can be seen that ideally the bit should be constant within the class (Low entropy η).

$$w_1 = 1 - \eta_{\{within-class\}} \quad (7)$$

2. Inter-Class weight of the bit – There are two cases. If the intra- and outside-class expected values of the bits are different then low outside-class entropy is desired (Property 1).

$$w_{2\{different\}} = (0.5)^{\eta_{\{outside-class\}}} \quad (8)$$

In the ideal situation where outside-class entropy is 0, the weight would be 1 above. If the expected intra-class value of the bit is the same as the expected outside-class value then high outside-class entropy is desirable (Property 2).

$$w_{2\{same\}} = 0.5 * \eta_{\{outside-class\}} \quad (9)$$

Since the outside-class entropy will only serve to reduce false acceptances by 50% at best, the weight is chosen to be a maximum of 0.5 above. Note that 'expected' value here is either 1 or 0 and not the statistical mean as a real number.

The overall weight for that bit is chosen as the product of the intra-class weight and appropriate inter-class weight.

Given an image of a user in the system he or she can be verified using that given image and the stored information on the verifying system, namely the inter and intra-class weights of each bit as well as the expected values of the bits that this user should produce. The same process is used on the supplied image to produce a pair of binary matrices one from the angle and another from the magnitude information.

The binary values in the matrix are compared with the stored expected values using a form of weighted Hamming distance as given in equation 7. A mismatch will decrease the score by the same amount that a match would have increased it. A total score is produced by multiplying this value with the overall weight of that bit and summing over all the bits in the matrix.

$$s = \sum_{i \in \{\text{all bits}\}} w_i \, signum[(b_{i(source)} \overline{\oplus} b_{i(destination)}) - 0.5] \quad (10)$$

where w_i represents the total weight of the bit, b_i its value at the source and the destination, respectively, and $\overline{\oplus}$ the exclusive NOR operation.

Weights are calculated only from the training set images. The system was trained using 15 images from each of 61 subjects and the performance of the system was tested over the remaining images. Only the angle information is used here. Three permutations of train/test splits were used – with the first 15, middle 15 and last 15 images being used for training in each case. Performance of the method is shown in figure 5.

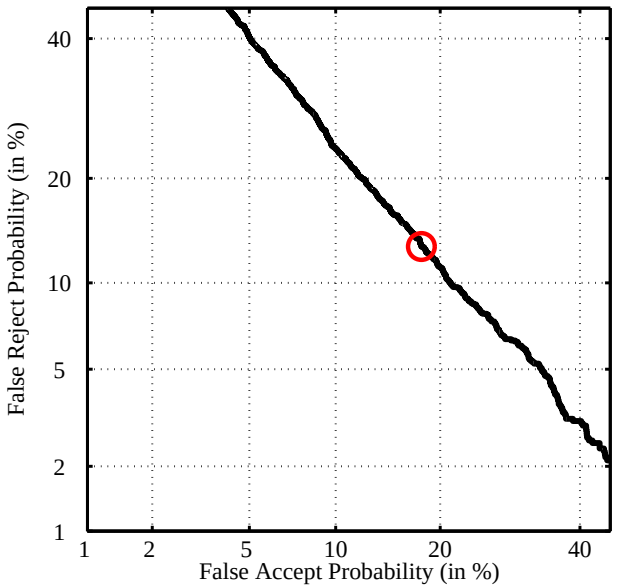

Figure 5. Detection Error Trade-off plot for the method. The equal error rate point is marked with a circle.

The equal error rate is seen above to be 15.5%. Although this is rather high and the best results that have been reported on the FRGC data [10,14] are around 2% EER on the 3D data, it serves to illustrate the usefulness of the method. We are using down-sampled data and only 6 Radon projections with considerable room for improvement. Improvements that are incorporated in other biometric feature extraction techniques such as extraction of features separately from parts of the face [14-16] and particular frequency bands can also be applied with this method. False rejections at a given false acceptance can also be lowered with multimodal or pseudo-multimodal systems [13]. They will be considered in future work.

B. RANDOM NUMBER GENERATION

For random number generation, bits that show high entropy over all inputs were selected. The word extracted from each input was varied in size from 32 to 512 bits. All the words were appended to form a large stream and this stream was passed through six tests for statistical randomness (frequency test, serial test, poker test (4-bit test), runs test, autocorrelation test, and DFT spectral

analysis). These tests were implemented in Matlab according to the algorithms supplied in the Handbook of Applied Cryptography [11] and the NIST paper on pseudo-random number generators [12]. All tests except for the DFT spectral analysis and autocorrelation should follow a Chi-squared distribution with significance level of 0.05. Autocorrelation and spectral analysis tests follow a standard normal distribution. Acceptable thresholds for the tests are presented in table 2 and the test results for different random word sizes are shown in table 3.

TABLE 2. ACCEPTABLE THRESHOLDS FOR THE RANDOMNESS TESTS.

Test	Degrees of Freedom	Acceptable Threshold
Frequency test	1	3.8415 max
Serial test	2	5.9915 max
Poker test (m = 4)	7	24.9958 max
Runs test (runs of greater than 16 equals 16) (k = 16)	30	43.7730 max
Autocorrelation test (shift = 4)		1.96 max
DFT Spectral Analysis		0.01 min

Descriptions of the tests can be found in references 11, 12.

TABLE 3. RANDOMNESS TEST RESULTS.

Word size	One Bit	Serial (2 bit)	Poker (4 bit)	Runs	Auto Corr.	DFT
32	PASS 0.27	PASS 1.27	FAIL 41.1	PASS 17	PASS -1.3	PASS 0.869
64	PASS 0.461	PASS 0.664	FAIL 25.7	PASS 15.5	PASS -0.17	PASS 0.466
128	PASS 0.46	PASS 0.95	PASS 14.6	PASS 31	PASS 0.082	PASS 0.6
256	PASS 0.95	PASS 0.12	PASS 7.6	PASS 24.7	PASS 0.589	PASS 0.74
512	PASS 0.28	PASS 1.34	PASS 7.07	PASS 17.9	PASS 0.672	PASS 0.254

VI. CONCLUSION

A method of simultaneously generating a reliable feature vector for biometric verification and a random number is proposed. The method can be further developed to generate keys for cryptography. The core concept is of extracting bit matrices through an iterated chaotic nonlinear transformation while preserving feature invariance or robustness and using entropy criteria to select bits. The method can be applied to any biometric in raw data form.

ACKNOWLEDGMENT

The authors acknowledge the use of FRGC face data and assistance from Mr. Jamie Cook and Prof. S. Sridharan.

REFERENCES

[1] P.K. Janbandhu and M.Y. Siyal, “Novel biometric digital signatures for Internet based applications,” Info. Management and Computer Security, vol. 5, no. 9, 205-212, 2001.

[2] S. Hoque, M. Fairhurst, G. Howells and F. Deravi, “Feasibility of generating biometric encryption keys,” Electronics Letters, vol. 41, no. 6, pp. 309-311, 2005.

[3] U. Uludag and A. K. Jain, “Multimedia content protection via biometric encryption,” Proc. of Intl. conf. on Multimedia and Expo (ICME’03), vol. 3, pp. 237-240, 2003.

[4] T. Stojanovski and L. Kocarev, “Chaos-based random number generators – part 1: Analysis (Cryptography),” IEEE Trans. On Circuits and Systems, vol. 48, no. 3, pp. 281-288, 2003.

[5] S. Callegari, R. Rovatti and G. Setti, “Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos,” IEEE Trans. On Signal Processing, vol. 53, no. 2, pp. 793-805, 2005.

[6] V. Chandran and S.L. Elgar, “Pattern Recognition Using Invariants Defined from Higher Order Spectra—One-Dimensional Inputs”, IEEE Transactions on Signal Processing, Vol. 41, No. 1, pp. 205—212, January 1993.

[7] V. Chandran, *et al.*, “Pattern Recognition Using Invariants Defined from Higher Order Spectra: 2-D Image Inputs”, IEEE Transactions on Image Processing, Vol. 6, No. 5, pp. 703—712, May 1997.

[8] V. Chandran, S. Elgar and C. Pezeshki, “Bispectral and Trispectral Characterization of Transition to Chaos in the Duffing Oscillator,” Intl. Journal of Bifurcation and Chaos, vol. 3, no. 3, pp. 551-557, 1993.

[9] P. J. Phillips, P. J. Flynn, T. Scruggs, Kevin W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the face recognition grand challenge,” in Proc. of IEEE Comp. Society Conf. on Computer Vision and Pattern Recognition (CVPR’05)-Vol. 1,Washington, DC, USA, pp. 947–954, 2005.

[10] K.W. Bowyer, K. Chang and P. Flynn, “A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition,” Computer Vision and Image Understanding, vol. 101, no. 1, pp. 1-15, 2006.

[11] A. Menezes, P. van Oorschot and S. Vanstone, Handbook of Applied Cryptography, USA: CRC Press, pp 180, 1997.

[12] "A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications", NIST Special Publication 800-22, 2001.

[13] V. Chandran**,** “Biometrics: New Perspectives on Multimodal and Client-centred Systems”, Proc. Intl. Workshop on Recent Advances in Biometrics, IIT Kanpur, India, pp. 77-89, April 15-16, 2005.

[14] J. Cook, V. Chandran and C. Fookes, “3D Face Recognition using Log-Gabor Templates,” Proc. of the British Machine Vision Conference 2006, accepted.

[15] C. Sanderson and K. K. Paliwal, “Fast features for face authentication under illumination direction changes,” Pattern Recognition Letters, vol. 24, no. 14, pp. 2409-2419, 2003.

[16] S. Lucey and T. Chen, “A GMM parts based face representation for improved verification through relevance adaptation,” Proc. of Computer Vision and Pattern Recognition (CVPR) 2004, vol. II, pp. 855-861, 2004.