Biometric Keys Based On Pseudorandom Sequences

Mofreh A. Hogo
Electrical Engineering Technology Department,
Faculty of Engineering Benha, Benha University,
Benha, Egypt.
Email: mofreh_hogo@ hotmail.com.

***Abstract*—this paper introduces a new technique for Key Generation based on fingerprints, Genetic Algorithms, and pseudorandom generator. The proposed technique can be applied in block ciphering techniques to enhance its performance (increase its complexity space). The paper introduces the different stages of the key generation sub-system; including: preprocessing and features extraction to represent the fingerprints signature (ID). Secondly the computation of genetic biometric signature ID, the third stage is computation of the pseudorandom generator to generate the different keys for 3DES. Results analysis proved that; the generated keys were strong when compared with the weak or semi weak keys. The proposed key generation technique increases the key space as well as it increases the total overheads on the encryption system due to the extra operations added but these overheads can be neglected when compared with the strong generated keys and its high complexity space.**

Keywords - DES, Biometric Signature, Minutiae, Bifurcation, Genetic Algorithm, Pseudorandom Generator.

I. INTRODUCTION

Biometric security system (BSS) is a security system based on the biometric features. The BSS are not widely used, because people do not trust them yet. Fingerprints as a kind of human biometrics on fingertips have been widely used for personnel recognition in the commercial and forensic areas because of its uniqueness and immutability [1-2]. Fingerprint is composed of interleaved parallel ridge and valley flows. Fingerprint features can be categorized into three different levels; level 1features capture the macro details of fingerprint such as the ridge flow structures, pattern types, orientation field and singular points (core and delta points). These coarse level features are not adequate to determine the uniqueness of a fingerprint. Level 2 features capture the minute details of ridge flow patterns such as the ridge bifurcations and endings (minutiae points). Theoretical studies over a large number of fingerprint samples show that these features can provide a huge amount of discriminatory information to determine the individuality of a fingerprint [3]. Level 3 features include all dimensional attributes of ridge details such as ridge pores edge contours, incipient ridges, creases, scars and etc. Currently, most commercial AFIS work on 500-ppi images so their recognition algorithms primarily use level 1 and 2 features. In general, high level (levels 2 and 3) features are often used to provide more discriminatory information to estimate the individuality of fingerprint. The multi-level matching scheme can improve the efficiency and accuracy of large scale fingerprint identification [4]. In [5] a hybrid method is proposed to combine structural and statistical approaches to extract features from orientation field for classifying fingerprints into six classes. The hybrid orientation features can compensate for the limitations of the respective single feature extraction strategy and give a more comprehensive representation. Instead of using the raw orientation data, a fingerprint orientation model based on 2D Fourier expansions (FOMFE) is proposed to reconstruct orientation field and the model coefficients are used to construct a feature vector for fingerprint indexing [6]. In [7] several biometric applications have been adopted in civilian, commercial, and forensic areas. Traditionally, the physical characteristics used for human recognition include fingerprints [8-9], iris [10], retinal [11], and facial [12], while the behavioral ones include signature [13], voice [14], and gait [15]. Among all these biometric characteristics, fingerprints are considered one of the most reliable for human recognition due to their individuality and persistence [16]. The fingerprint's individuality means that it is unique across individuals and across fingers of the same individual, even in identical twins [17]. On the other hand, the fingerprint's persistence means that the basic fingerprint characteristics do not change with time. Automatic fingerprint recognition often involves four steps [18]: (1) acquisition, (2) classification, (3) identification, and (4) verification. To design a balanced system, in which the FAR (False Acceptance Rate) and FRR (False Rejection Rate) rates were very low, is goal of the BSS designers. The rest of the paper is organized as following: Section II introduces the theoretical overview of the DES block cipher algorithm. Section III the extraction of minutiae, statistical features, and hybridization between them. Section IV presents the computation of genetic keys. Section V presents the generation for pseudorandom sequence. Section VI presents the comparison with the semi weak and weak DES.

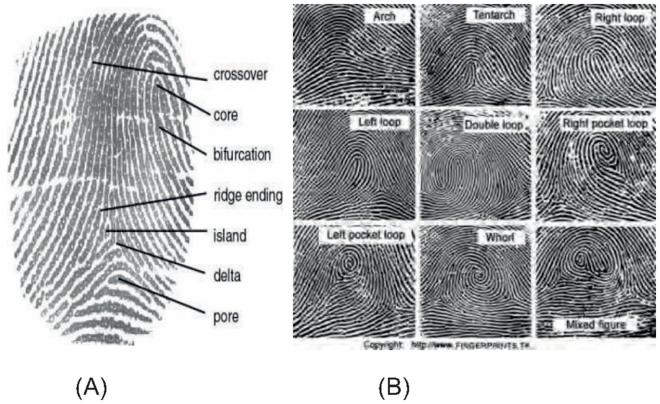
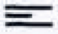
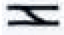
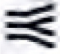
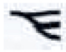
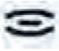
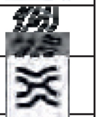
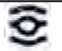
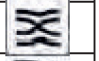
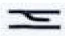
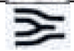


Figure 1. (A) Fingerprints Parts, (B) Types Of Fingerprints

978-1-4673-2451-9/12/$31.00 ©2012 IEEE

TABLE I. BASIC AND COMPOSITE RIDGE

Minutiae	Example	Minutiae	Example
Ridge Ending		Bridge	
Bifurcation		Double Bifurcation	
Bifurcation.		Short Ridge (Dot).	
Island (Short Ridge)		Opposed Bifurcations	
Lake (Enclosure)		Ridge Crossing	
Hook (Spur)		Opposed Bifurcation/Ridge Ending	

Finally section VII is reserved for the conclusion.

II. THEORETICAL REVIEW OF DES

DES: *Data Encryption Standard* was the first encryption standard to be recommended by NIST (National Institute of Standards and Technology). DES is 64 bits key size with 64 bits block size. The same DES algorithm and key are used for both encryption and decryption. DES operates on a 64-bit block of plaintext. After an initial permutation, the block is broken into a right half and a left half, each 32 bits long. Then there are 16 rounds of identical operations, called Function *f*, in which the data are combined with the key. After the 16 round, the right and left halves are joined, and a final permutation (the inverse of the initial permutation) finishes off the algorithm. In each round the key bits are shifted, and then 48 bits are selected from the 56 bits of the key, the right half of the data is expanded to 48 bits via an expansion permutation, combined with 48 bits of a shifted and permuted key via an XOR sent through 8 S-boxes producing 32 new bits, and permuted again. These four operations make up Function *f*. The output of Function *f* is then combined with the left half via another XOR. The result of these operations becomes the new right half; the old right half becomes the new left half. These operations are repeated 16 times, making 16 rounds of DES. If Bi is the result of the i^{th} iteration, L_i and R_i are the left and right halves of B_i, K_i is the 48-bit key for round i, and f is the function that does all the substituting and permuting and XORing with the key, then a round looks like: $L_i = R_{i-1}$, and $R_i = L_{i-1}\ f\ (_{Ri-1}, K_i)$. DES/3DES applied in virtual private networking *(VPN),* secure extranet protocols, secure Storage devices as smartcards, in XML encryption, and for secure E- business.

III. FINGERPRINT BIOMETRIC SIGNATURE

Fingerprints remain constant throughout life, no two fingerprints have ever been found to be alike, not even those of identical twins. Fingerprint identification involves comparing the pattern of ridges and furrows on the fingertips, as well as the minutiae points.

A. Recognition Using Minutiae Features

Identification by fingerprints relies on pattern matching followed by the detection of certain ridge characteristics, points of identity, or minutiae, and the comparison of the relative positions of these minutiae points with a reference print, usually an inked impression of a suspect's print. There are three basic ridge characteristics, the ridge ending, the bifurcation and the dot (or island). A single rolled fingerprint may have as many as 100 or more identification points that can be used for identification purposes. There is no exact size requirement as the number of points found on a fingerprint impression depends on the location of the print. As an example the area immediately surrounding a delta will probably contain more points per square millimeter than the area near the tip of the finger which tends to not have that many points. Fig. 1(a) shows the part of fingerprint. Fig. 1(b) shows examples of fingerprints. Table I, illustrates the basic and composite ridge characteristics.

Fingerprint recognition refers to the automated method of verifying a match between two human fingerprints. Fingerprints are one of many forms of biometrics used to identify an individual and verify their identity. The analysis of fingerprints for matching purposes generally requires the comparison of several features of the print pattern. These include patterns, which are aggregate characteristics of ridges, and minutiae points, which are unique features found within the patterns [19]. *Patterns:* The three basic patterns of fingerprint ridges are the arch, loop, and whorl. An arch is a pattern where the ridges enter from one side of the finger, rise in the center forming an arc, and then exit the other side of the finger. The loop is a pattern where the ridges enter from one side of a finger, form a curve, and tend to exit from the same side they enter. In the whorl pattern, ridges form circularly around a central point on the finger. Scientists have found that family members often share the same general fingerprint patterns, leading to the belief that these patterns are inherited. The major Minutiae features of fingerprint ridges are: ridge ending, bifurcation, and short ridge (dot). The ridge ending is the point at which a ridge terminates. Bifurcations are points at which a single ridge splits into two ridges. Short ridges are ridges which are significantly shorter than the average ridge length on the fingerprint. Minutiae and patterns are very important in the analysis of fingerprints, where most algorithms are using minutiae, the specific points like ridges ending, bifurcation. Only the position and direction of these features are stored in the signature for further comparison. The steps to extract and recognize minutiae are:

1) Read Fingerprints' Image & Enhancement: identification/verification system would be robust with respect to the quality of the images.

2) Thinning:

3) Minutiae: The thinned ridge map is filtered by the filter minutiae. Compute the number of one-value of each 3x3 window:

   a. If the central is 1 and has only one-value neighbor, then the central pixel is a termination.

   b. If the central is 1 and has 3 one-value neighbors, then the central pixel is a bifurcation.

   c. If the central is 1 and has 2 one-value neighbors, then the central pixel is a usual pixel.

TABLE II. SAMPLE OF TERMINATIONS =21; BIFURCATIONS=9

X	Y	$\frac{y}{x}$	$\frac{dy}{dx}$	Angle	Angle^2
52	26	0.50	0.5	0	0
58	29	0.50	0.5	3.14	9.8596
42	43	1.02	-0.875	-2.62	6.8644
154	58	0.38	0.133928571	2.36	5.5696
52	59	1.13	-0.009803922	0.52	0.2704
180	79	0.44	0.15625	-1.05	1.1025
61	92	1.51	-0.109243697	-2.09	4.3681
93	98	1.05	0.1875	-1.57	2.4649
137	116	0.85	0.409090909	-1.57	2.4649
151	116	0.77	0	-1.05	1.1025
108	117	1.08	-0.023255814	-2.09	4.3681
70	124	1.77	-0.184210526	-0.79	0.6241
162	126	0.78	0.02173913	-1.57	2.4649
79	133	1.68	-0.084337349	2.36	5.5696
108	142	1.31	0.310344828	-2.36	5.5696
32	146	4.56	-0.052631579	-1.05	1.1025
44	170	3.86	2	3.14	9.8596
97	175	1.80	0.094339623	-2.62	6.8644
75	176	2.35	-0.045454545	3.14	9.8596
113	180	1.59	0.105263158	0.52	0.2704
119	180	1.51	0	0	0
154	180	1.17	0	1.05	1.1025
49	186	3.80	-0.057142857	-2.62	6.8644
71	186	2.62	0	3.14	9.8596
∑ Angle^2= 98.4462 **MAV=√ 98.4462= 9.9220058**					

4) Termination and Bifurcation processing: A lot of spurious minutiae have to be processed using a specific threshold distance d as: If ((distance between a termination and a bifurcation) OR (distance between two bifurcations) OR (distance between two terminations)) < d) then remove this minutiae.

5) Orientation: find the orientation of each minutia.

6) The $\frac{dy}{dx}$ rate of the Y position change relative to rate of X position change for the Minutiae and bifurcation, it is another new proposed feature that can be used for login and key generation. These extracted features are illustrated in Tables II and III.

7) The select the features vector:

a. For each angle find the square value of it (Angle^2).

b. Find sum^2 =∑of all squared values obtained in step (a).

c. MAV =√ sum^2 For the Bifurcations consider only the 3 angles' values; compute the values of the Magnitude Angle Vector as computed in steps (a,b,c); the three Angles Angle$_1$, Angle$_2$, Angle$_3$.

d. The final vector obtained consists of the following attributes:

- MAV for the terminations Angles.
- MAV$_1$, MAV$_2$, and MAV$_3$ for the three bifurcation angles.

TABLE III. SAMPLE OF BIFURCATIONS

x	y	Angle$_1$	Angle$_2$	Angle$_3$	Angle$_1$$^{\wedge 2}$	Angle$_2$$^{\wedge 2}$	Angle$_3$$^{\wedge 2}$
1.1025	0.2704	6.8644	1.05	-0.52	-2.62	88	59
2.4649	0.2704	6.8644	1.57	-0.52	-2.62	114	80
4.3681	0	6.8644	-2.09	0	2.62	125	92
1.1025	0	9.8596	1.05	0	3.14	128	80
4.3681	0	5.5696	-2.09	0	2.36	276	116
2.4649	0	5.5696	-1.57	0	2.36	285	177
∑ Angl12= 41.592 MAV1= √ 41.592 = 6.449186			∑ Angl22= 0.5408 MAV2=√0.5408 = 0.7353911		∑ Angl32= 15.871 MAV3=√15.871 = 3.9838424		

TABLE IV. MINUTIAE & BIFURCATIONS FEATURES

Sample#	MAV	MAV$_1$	MAV$_2$	MAV$_3$
1	8.5604965	7.349204	5.2125905	1.9457646
2	9.9220058	6.449186	0.7353911	3.9838424
3	7.6921193	5.1149585	3.6967553	0.7353911
4	10.556792	4.1239567	2.2137345	3.9838424

e. $\frac{dy}{dx}$ *Rate of y/x change for the minutiae and bifurcations will be included instead of x and y coordinates.*

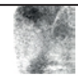
Minutiae Identification (Minutiae match): Given two set of Minutiae of two fingerprints signatures, the Minutiae match algorithm determines; whether the two Minutiae sets are from the same finger or not. Two steps: Alignment stage and Match stage. This section introduces the results obtained from the fingerprint signatures identification based on minutiae. The matching algorithm also presented. Samples from these features are shown in Table IV. The test of our system is carried on using the NIST-4 database [20], which consists of 4,000 images (512 x 512); we use 1000 images only. Impressions scanned at 500 dpi; each finger in the database has two impressions. The images in the NIST-4 database are stored with numerical order as f0001 to f2000 and s0001 to s2000. All of the data are used as testing. The matching algorithm considers dy/dx and the angles of the termination points that will be computed (processed). The recognition and matching process is dependent on the distance measure between the unknown fingerprint Minutiae vector and the stored fingerprint vectors database. Find the minutiae for the unknown fingerprint. Construct the vector MAV, MAV$_1$, MAV$_2$, and MAV$_3$. Find the nearest matched vector from the database of the fingerprint, using the three Euclidean distances measures using the following formulas.

$$D_{IJmav} = \sqrt{\begin{matrix}(MAV_I - MAV_J)^2 + (MAV_{I1} - MAV_{J1})^2 + \\ (MAV_{I2} - MAV_{J2})^2 + (MAV_{I3} - MAV_{J3})^2\end{matrix}}$$

$$D_{IJy/x} = \sqrt{\begin{matrix}\left(dy/dx_{1I} - dy/dx_{1_J}\right)^2 + \left(dy/dx_{2I} - dy/dx_{2_J}\right)^2 + \\ +(dy/dx_{nI} - dy/dx_{n_J})^2\end{matrix}}$$

$$D_{totla} = \sqrt{\left(D_{IJmav}\right)^2 + \left(D_{IJy/x}\right)^2}$$

TABLE V. SAMPLES OF STATISTICAL FEATURES

Sample#				
	1	2	3	4
Mean	157.1	122.8	179.6000	224.6
Var	4670.7	2390.2	980.5840	1030.21
St_Dev	68.300	48.9	31.3143	32.1
Smooth	1	1	0.9999	1
Kurt	2.0	2.6	1.8608	2.2
Skewness	-0.6	0.1	-0.2082	-0.7

B. Recognition Based on Statistical Features

This section introduces the second proposal for recognizing the fingerprints. The proposed method based on the extraction of a robust set of statistical features as: Mean, Standard deviation, Variance, Smoothness, Skewness, and the kurtosis.

1) *Statistical Moments Features:*

2) The expression for computing the n^th moment about the mean given by: $\mu_n = \sum_{i=0}^{L-1}(z_i -$ *mnpzi;* where Mean is given by: $m = \sum_{i=0}^{L-1} Z_i P(Z_i),$

3) *Standard Deviation:* $\sigma = \sqrt{\mu} Z_i = \sqrt{\sigma^2},$

4) *Smoothness:* $R = 1 - \frac{1}{(1+\sigma^2)}$

5) *Third Moment:* $\mu_3 = \sum_{i=0}^{L-1}(Z_i - m)^3 F(Z_i),$ Measures the Skewness of a histogram this measures is 0 for symmetric histograms, positive by histograms skewed to the right (about the mean) and negative for histograms skewed to the left. Values of this measure are brought into a range of values comparable to the other five measures by dividing m_3 by $(L - 1)^2$ also, which is the same divisor we used to normalize the variance [21].

6) *Kurtosis:* Statistics kurtosis measures of the flatness of a curves [21]; and calculated by: $\mu_4 = \sum_{i=0}^{L-1}(Z_i - m)^4 P(Z_i)$

7) *Uniformity:* It measures the uniformity and its maximum when all gray levels are equal (maximally uniform) and decreases from there [21], it can be computed by:

$$U = \sum_{i=0}^{L-1} P^2(Z_i),$$

8) *Entropy:* A measure of randomness and calculated by: $e = -\sum_{i=0}^{L-1} P(Z_i) \log_2 P(Z_i)$; where z_i is a random variable indicating intensity $P(z_i)$ is the histogram of the intensity level in a region, L is the number of possible intensity levels and, $m = \sum_{i=0}^{L-1} Z_i P(Z_i) \mu_n = \sum_{i=0}^{L-1}(Z_i - m)^n P(Z_i)$

The extraction of the proposed statistical features is implemented using MATLAB.

The steps to extract and identify these features are:

Initialization Step

Image = imread ('imagefilename.bmp');

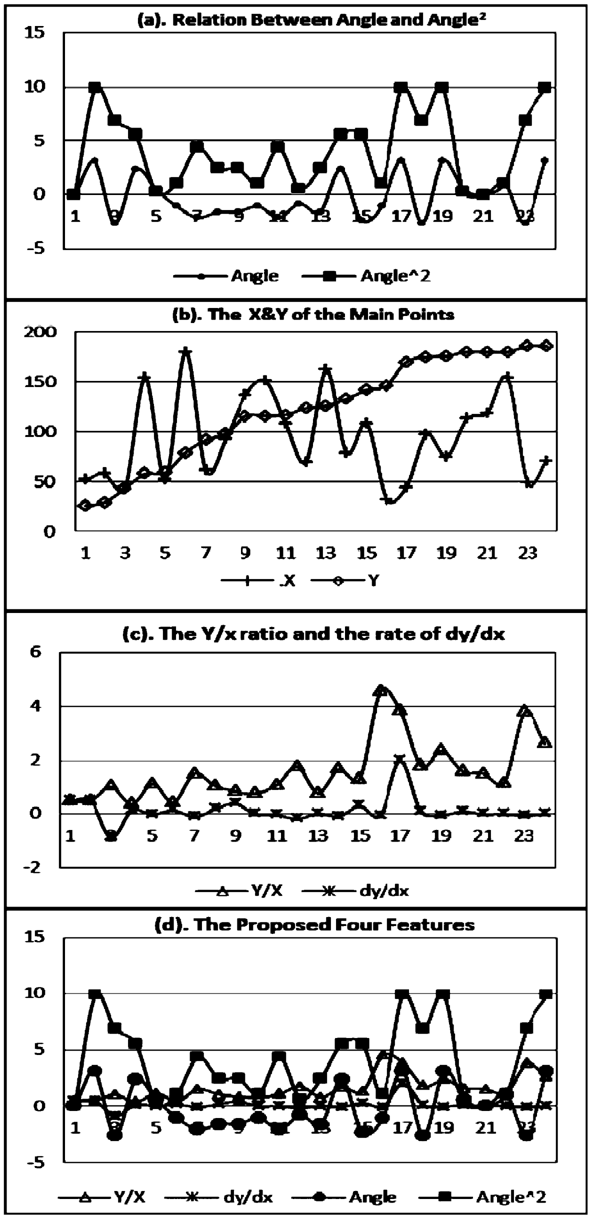


Figure 2. Extended Minutiae Features

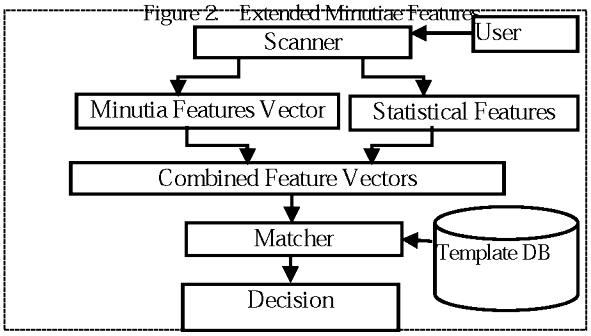


Figure 3. Combination Of Minutiae And Statistical Features

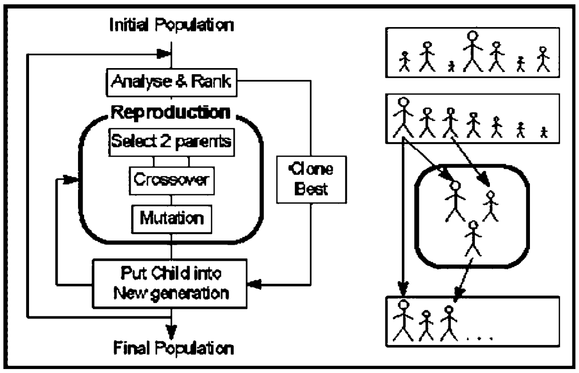


Figure 4. Genetic Algorithm Steps

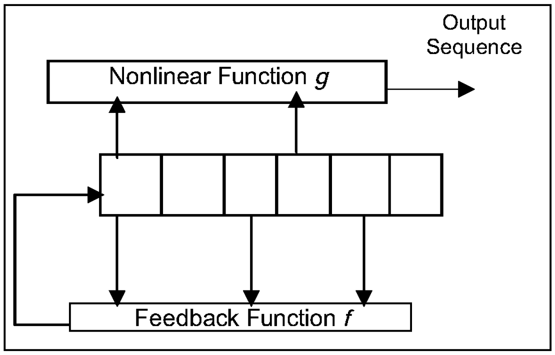


Figure 5. Nlffsr Structure

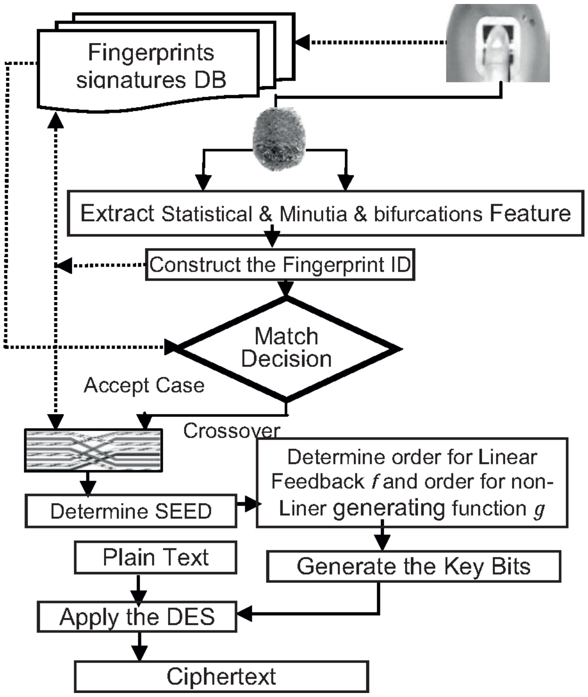

Fingerprints signatures DB
Extract Statistical & Minutia & bifurcations Feature
Construct the Fingerprint ID
Match Decision
Accept Case
Crossover
Determine SEED
Determine order for Linear Feedback f and order for non-Liner generating function g
Plain Text
Generate the Key Bits
Apply the DES
Ciphertext

Figure 6. Complete Structure Of The System

```
Image =rgb2gray (image); [m,n]=size (image);
    Imaged= double (reshape (image, 1,m*n));
    l=length (x 1);
```

```
1.  Calculate F 1: Mean
    s=0;
    for i= 1:l
    s=s+x 1(i);
    end
    Avx 1=s /l;
    Xim=mean (x 1);
        2   Calculate F2 Variance
v=0;
for i= 1:l
v=v+ (x 1(i)-Avx 1) ^2; end

    vx 1=v /(l- 1);
    xiv=var (x 1);
        3   Calculate F3 Standard Deviation
    stdx 1=sqrt (vx 1);
    sgmx 1=std (x 1);
    Calculate F4 Smoothness
    Rx 1= 1- 1/( 1+vx 1);
    rx 1= 1- 1/( 1+xiv);
        4   Calculate F5  Skewness
    sk=0;
    for i= 1:l
    sk=sk+ (x 1(i)-Avx 1) ^3;
    end
    skx 1=sk /(l*stdx 1^3);
    x 1sk=skewness (x 1);
        5   Calculate F6 Kurtosis
    kr=0;
    for i= 1:l
    kr=kr+ (x 1(i)-Avx 1) ^4;
    end
    krx 1=kr /(l.*stdx 1. ^4);
    x 1kr=kurtosis (x 1);
    fet=[Avx 1 vx 1 stdx 1 Rx 1 skx 1 krx 1]
```

6. *The output features vector is listed below:*

FV= [Avge, Varia, St_Dev, Smoo, Skew, Kurt]
= [F1, F2, F3, F4, F5, F6]

The vector consists of 6 features: Mean, Variance, Standard Deviation, Smoothness, Kurtosis, and Skewness. But the smoothness feature was rejected; where it has no effect on the identification step as well as $\mathbf{sin\theta_4} = \frac{\mathbf{F_4}}{\mathbf{magnitude}}$ was rejected where it based on $\mathbf{F_4}$. Therefore we consider only 5 attributes.

7. *Steps of Identification Using Statistical Features were listed below (For unknown Fingerprints, find the following :)*

a. Find the magnitude of the statistical feature vector using: $\boldsymbol{mag} = \sqrt{\boldsymbol{f_1^2} + \boldsymbol{f_2^2} + \boldsymbol{f_3^2} + \boldsymbol{f_5^2} + \boldsymbol{f_6^2}}$

b. Find the orientation of this vector using: $\mathbf{sin\theta_1} = \frac{\mathbf{F_1}}{\mathbf{magnitude}}$; $\mathbf{sin\theta_2} = \frac{\mathbf{F_2}}{\mathbf{magnitude}}$; $\mathbf{sin\theta_3} = \frac{\mathbf{F_3}}{\mathbf{magnitude}}$; $\mathbf{sin\theta_5} = \frac{\mathbf{F_5}}{\mathbf{magnitude}}$; $\mathbf{sin\theta_6} = \frac{\mathbf{F_6}}{\mathbf{magnitude}}$.

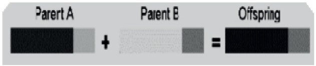
c. Use the Euclidian distance for Matching as following:

d. Construct and post process the of the statistical features vector as following:

TABLE VI. CROSSOVER OPERATIONS

<table>
<tr><td>

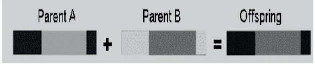
Single Point Crossover



11001011+11011111 = 11001111

</td></tr>
<tr><td>

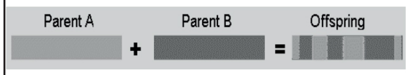
Two Point Crossover



11001011 + 11011111 = 11011111

</td></tr>
<tr><td>

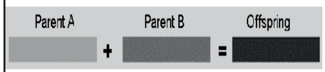
Uniform Crossover



11001011 + 11011101 = 11011111

</td></tr>
<tr><td>

Arithmetic Crossover



(AND)

11001011 + 11011111 = 11001011

</td></tr>
</table>

- Magnitude mag, $(\sin\theta_1)$, $(\sin\theta_2)$, $(\sin\theta_3)$, and $(\sin\theta_5)$.

e. Table V shows samples from these computes selected statistical features.

- Apply the Euclidian formula to find the nearest matched vector from the stored database of the fingerprints with unknown Fingerprint signature; mag, $(\sin\theta_1)$, $(\sin\theta_2)$, $(\sin\theta_3)$, $(\sin\theta_5)$, and $(\sin\theta_6)$, using the 5-dimensional Euclidean distance. As for two N-D points, FI=(F_{I1},F_{I2},F_{I3}, F_{I5},F_{I6}) and FJ=(F_{J1},F_{J2},F_{J3},F_{I},F_{J6}), the distance is computed by D_{IJ} by:

$$D_{IJ} = \sqrt{\begin{array}{c}(Mag_{I1} - Mag_{J1})^2 + (sin(\theta)_{I1} - sin(\theta)_{J1})^2 \\ +(sin(\theta)_{I2} - sin(\theta)_{J2})^2 \\ +(sin(\theta)_{I3} - sin(\theta)_{J3})^2 + (sin(\theta)_{I5} - sin(\theta)_{J5})^2\end{array}}$$

C. Identification Based On Minutiae & Statistical Features

A novel modest contribution in this work is the hybridization between the minutiae and statistical features, the goal of this hybridization is to provide a robust biometric key from fingerprints signature. The complete structure of the matching system is shown in Fig. 3, which illustrates the identification steps as following:

1) Combine the five features obtained from the minutiae with the five features obtained from the statistical features analysis.
2) The vector of hybridized features consists of 9 attributes plus those for the as: *[Mag. $(\sin\theta_1)$, $(\sin\theta_2)$, $(\sin\theta_3)$, $(\sin\theta_5)$,*
3) MAV, MAV1, MAV2, and MAV3], plus those for $\frac{dy}{dx}$: Rate of y/x change for the minutiae and bifurcations will be included instead of x and y coordinates.
4) Using the Euclidean distance to verify the fingerprints ID.

The idea of combination between the two set of features produces a classifier with 100% accuracy rate of classification when tested with 1000 samples using the NIST-4 database. This obtained accuracy gives the power to complete the development of the rest of the system of encryption using the proposed algorithm.

TABLE VII. (A) DES SEMIWEAK KEY PAIRS

01FE	01FE	01FE	01FE	and	FE01	FE01	FE01	FE01
1FE0	1FE0	0EF1	0EF1	and	E01F	E01F	F10E	F10E
01E0	01E0	01F1	01F1	and	E001	E001	F101	F101
1FFE	1FFE	0EFE	0EFE	and	FE1F	FE1F	FE0E	FE0E
011F	011F	010E	010E	and	1F01	1F01	0E01	0E01
E0FE	E0FE	F1FE	F1FE	and	FEE0	FEE0	FEF1	FEF1

(B) DES POSSIBLY WEAK KEYS

Alternating ones + zeros (0x0101010101010101)
Alternating 'F' + 'E' (0xFEFEFEFEFEFEFEFE)
'0xE0E0E0E0F1F1F1F1'
'0x1F1F1F1F0E0E0E0E'
If an implementation does not consider the parity bits, the corresponding keys with the inverted parity bits may also work as weak keys:
all zeros (0x0000000000000000)
all ones (0xFFFFFFFFFFFFFFFF)
'0xE1E1E1E1F0F0F0F0'
'0x1E1E1E1E0F0F0F0F'

IV. GENETIC FINGERPRINTS KEYS

A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems [22]. Genetic algorithms are categorized as global search heuristics. Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as mutation, selection, and crossover (also called recombination). Fig. 4, Shows the main steps of the GA.

The complete steps of GA are not needed in this work; just we apply the crossover operations in its various forms as. Many crossover techniques exist for organisms which use different data structures to store themselves. The crossover operations are implemented using C++ to find the different types of crossover operations. The different types of crossover operations are illustrated in Table VI.

1. The used type of crossover in this work is the arithmetic crossover using (XOR logical operation). The mechanism of the crossover operations are illustrated in Table IX (A), as well as the steps of Genetic Fingerprints ID computation.

V. GENETIC PSEUDORANDOM SEQUENCE

A pseudorandom generator is a function *g*: $\{0, 1\}^1 \rightarrow \{0, 1\}^*$ that expands a short seed into a bit sequence of arbitrary length. In order to be of cryptographic interest, *g* has to be computable by an efficient algorithm. Components of such a generator are:

1. An inner state $S_i \in$ *g*: $\{0, 1^1$,
2. An update function *f*: $\{0, 1\}^1 \rightarrow \{0, 1\}^1$ that modifies the inner state between two outputs, and

3. An output function g: $\{0, 1\}^v \rightarrow \{0, 1\}$, $v \leq l$ that computes the next output bit from (part of) the current inner state.

The seed value S_0 and the relation $S_0 = f\ (S_{i-1})$ form a recurrence, defining the sequence of all inner states that the generator assumes over time. Also note that the generator can assume at most 2^l different inner states, yielding an upper bound on the least period of 2^l.

NLFFSR: a mechanism for generating extremely well pseudorandom binary sequence. The steps for generating the Key as: Firstly generate the genetic-biometric key then pass it to Non-Linear Feedback Shift Register (NLFFSR) presented in [23] to generate the pseudorandom sequence. Fig. 5 shows a general model of NLFFSR. It is a nonlinear forward feedback shift register with a feedback function *f* and nonlinear function *g*. Binary sequence is generated with very good randomness and statistical properties. The only signal necessary for the generation of the binary sequence is a clock pulse. With each clock pulse a bit of the binary sequence is produced.

The most important part is the use of the order bits from crossover as a seed for the feedback function *f* and repeated order of the digits obtained from crossover are used a seed for the nonlinear function g. The usefulness of these sequences depends in large part on their having nearly randomness properties. Therefore such sequences are termed as pseudorandom binary sequences. The balance, run and correlation properties of these sequences make them more useful in the selection of secret keys [23].

The proposed NLFFSR concept is shown in Fig. 5. The steps illustrate the computing of the linear feedback and nonlinear function g. The overall structure of the new proposed Fingerprints identification and key generating systems is shown in Fig. 6. *The sequence of 3DES encryption:* CIPHERTEXT = EK3 (DK2 (EK1 (plaintext))); i.e., DES encrypts with K1, DES decrypt with K2, and then
DES encrypt with K3. *Decryption Steps using the proposed Genetic Biometric Keys: The decryption* is a reverse of the encryption. Plaintext = DK1 (EK2 (DK3 (ciphertext))); decrypt with K3, encrypt with K2, then decrypt with K1. The general sequence of the encryption process done as follows:

1. Capture and preprocess the fingerprint of the user.
2. Extract the minutia and statistical features and combine them to find the fingerprints signature or fingerprints ID.
3. Construct the genetic fingerprint ID (arithmetic crossover operations).
4. Generate the genetic pseudorandom sequence to find the seed for the liner feedback *f* and nonlinear function *g*.
5. Use the NLFFSR to generate the three keys K1 K2 K3; sequentially and periodically change the time slice for example generate 6 keys and select the odd keys as 1,3,5; generate 9 keys and select the odd with skip as 3,5,7; and so on. Or changing the scheme for selecting the 3 keys feed the K1, K2, and K3 to the 3DES.
6. The concluded features that will be the input to the genetic (Fingerprints ID) crossover operations are shown in Table VIII A,B,C,D.
7. Table IX (A) shows the computation of the genetic fingerprint ID.
8.

Table IX (B) shows the computation steps to compute the Genetic Pseudorandom fingerprints keys as: (1). Compute the equivalent order for the final the child results from crossover. (2). Find the total polynomial (16 terms). (3). Select the polynomial orders from the 16 (unrepeated terms). (4). Remove the redundancy. (5). Select final order of the polynomial. (5). Compute the SEED of NLFFSR (Feedback function f and nonlinear g; to generate key bits WEAK & SEMI WEAK DES KEYS

In cryptography, a weak key is a key; which when used with a specific cipher, makes the cipher behave in some undesirable way. Weak keys usually represent a very small fraction of the overall key space, which usually means that A cipher with no weak keys is said to have a flat, or linear, key space. The block cipher DES has a few specific keys termed "weak keys" and "semi-weak keys". These are keys which cause the encryption mode of DES to act identically to the decryption mode of DES (albeit potentially hat of a different key). In operation, the secret 56-bit key is broken up into 16 subkeys according to the DES key schedule; one subkey is used in each of the sixteen DES rounds. The weak keys of DES are those which produce sixteen identical subkeys. Tables VII (A) and (B); illustrate both of semiweak DES keys and weak DES keys. The comparison step of the generated biometric genetic pseudorandom sequence keys with the weak and semiweak DES keys is a critical step to ensure the strength of the generated keys and these keys are neither semi weak nor weak keys. The comparison step was carried using the generated keys and the key shown in Table VII (A) and (B). The results were assisting the strength of the generated keys. A simple matching program was implemented for this comparison step and the matching results were zero matches.

VI. CONCLUSION

The work introduced a modest contribution in securing DES using a new technique to generate biometric genetic pseudorandom sequence key. The generated key is based on Biometric, Genetic algorithms, pseudorandom sequence key, and periodical change of the sequence selection from the pseudorandom sequence. The work also introduced the combination between the minutiae (Gabor) and statistical features to identify/verify fingerprints. The proposed system was tested with 1000 samples from the NIST-4 database, the results were 100% classification accuracy, and the generated keys were strong when compared with the weak and semiweak DES keys. The proposed Key generating system proved its capabilities to be applicable in critical systems. Moreover the time-complexity of the proposed algorithm is more than the traditional DES due to the extra added operations.

TABLE I. BIOMETRIC SIGNATURE GENERATION

(A) MINUTIAE AND BIFURCATIONS FEATURE

Samples	MAV	MAV1	MAV2	MAV3
	F1	**F2**	**F3**	**F4**
1	8.5604965	7.349204	5.2125905	1.9457646
2	9.9220058	6.449186	0.7353911	3.9838424
3	7.6921193	5.1149585	3.6967553	0.7353911
4	10.55679234	4.1239567	2.2137345	3.9838424

(B) STATISTICAL FEATURE

Mag	Sin θ_1	Sin θ_2	Sin θ_3	Sin θ_4
F_5	**F_6**	**F_7**	**F_8**	**F_9**
4673.841	0.033613	0.014613	0.000428	0.000128
2393.855	0.051298	0.020427	0.001086	0.000418
997.3914	0.18007	0.031396	0.001875	0.000209
1054.903	0.212911	0.030429	0.002086	0.000664

(C) MINUTIAE AND BIFURCATIONS ID COMPUTATION

Samples	F1	F2	F3	F4	Sum	Square	Square* F5
	8.56049	7.3492	5.2125	1.94576	23.068	532.1352	2487115.265
1	9.9220058	6.449186	0.7353911	3.9838424	21.090425	444.8060393	1064801.16
2	7.6921193	5.1149585	3.6967553	0.7353911	17.239224	297.190851	296415.599
3	10.55679234	4.1239567	2.2137345	3.9838424	20.878326	435.9044941	459836.958

(D) STATISTICAL ID COMPUTATION

Samples	F6	F7	F8	F9	Sum	Square	Square* 10^6
	0.033613	0.014613	0.000428	0.000128	0.0487	0.00238	2379.683524
1	0.051298	0.020427	0.001086	0.000418	2393.928	5730892	5.73089E+12
2	0.18007	0.031396	0.001875	0.000209	997.605	995215.6	9.95216E+11
3	0.212911	0.030429	0.002086	0.000664	1055.149	1113340	1.11334E+12

TABLE IX (A) GENERATING GENETIC FINGERPRINT ID

2487115265	0011	0010	0011	0100	0011	1000	0011	0111	0011	0001	0011	0001	0011	0101	0011	0010
2379683524	0011	0010	0011	0011	0011	0111	0011	1001	0011	0110	0011	1000	0011	0011	0011	0101
Crossover (arithmetic) Using XOR	0001	0010	0111	0011	1011	0111	0100	1001	0010	0110	0010	1000	0110	0011	0001	0101

(B) COMPUTING SEEDS POLYNOMIAL FOR LINEAR & NONLINEAR FUNCTIONS KEY (NLFFSR)

The Equivalent Order For The Final Child After Crossover	1	2	7	3	B	7	4	9	2	6	2	8	6	3	1	5
The Total Polynomial 16	X^{1}	X^{2}	X^{3}	X^{4}	X^{5}	X^{6}	X^{7}	X^{8}	X^{9}	X^{10}	X^{11}	X^{12}	X^{13}	X^{14}	X^{15}	X^{16}
The Selected Polynomial	X^{1}	X^{2}	X^{7}	X^{3}	X^{11}	X^{7}	X^{4}	X^{9}	X^{2}	X^{6}	X^{2}	X^{8}	X^{6}	X^{3}	X^{1}	X^{5}
Remove The Redundancy	X^{1}	X^{2}	X^{7}	X^{3}	X^{11}	X^{7}	X^{4}	X^{9}	X^{2}	X^{6}	X^{2}	X^{8}	X^{6}	X^{3}	X^{1}	X^{5}
The Polynomial Without Redundancy	X^{1}	X^{2}	X^{7}	X^{3}	X^{11}		X^{4}	X^{9}		X^{6}		X^{8}				X^{5}
The Final Ordered Polynomial	X^{1}	X^{2}	X^{3}	X^{4}	X^{5}	X^{6}	X^{7}	X^{8}	X^{9}	X^{11}						
Feedback Function f To	X^{1}	+	X^{2}	+	X^{3}	+	X^{4}	+	X^{5}	+	X^{6}	+	X^{7}	+	X^{8}	+ X^{9} + X^{11}
nonlinear function g to generate key stream bits (the rest of the order selected digits order)	X^{10}	+	X^{12}	+	X^{13}	+	X^{14}	+	X^{15}	+	X^{16}					
nonlinear function g (the repeated bits)	X^{1}	+	X^{2}	+	X^{3}	+	X^{6}	+	X^{7}							

■ Repeated Polynomial Remove the Repeated + XOR Operation

VII. REFERENCES

[1] Manhua Liu, Pew-ThianYap, " Invariant representation of orientation fields for fingerprint indexing", Pattern Recognition Vol. 45, pp 2532–2542, 2012.

[2] A.K. Jain, J. Feng, K. Nandarkumar, "Fingerprint matching", IEEE Computer Vol. 43 (2), pp 36–44, 2010.

[3] S. Pankanti, S. Prabhakar, A.K. Jain, "On the individuality of fingerprints", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 24 (8) pp 1010–1025, 2002.

[4] N.K. Ratha, S. Chen, K. Karu, A. Jain, "A real-time matching system for large fingerprint databases", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 18 (8) pp 799–813, 1996.

[5] K.A. Nagaty, "Fingerprints classification using artificial neural networks: a combined structural and statistical approach",Neural Networks Vol. 14 pp 1293–1305, 2001.

[6] Y. Wang, J. Hu, D. Phillips, "A fingerprint orientation model based on 2d fourier expansion and its application to singular-point detection and fingerprint indexing", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 29 (4) pp 573–585, 2007.

[7] Javier A. Montoya Zegarra, Neucimar J. Leite , Ricardo da Silva Torres, "Wavelet-based fingerprint image retrieval" Journal of Computational and Applied Mathematics Vol.227 pp 294–307, 2009.

[8] A.K. Jain, A. Ross, S. Prab, "An introduction to biometric recognition", IEEE Transactions on Circuits and Systems for Video Technology Vol.14 (1),pp 4–20, 2004.

[9] A.K. Jain, L. Hong, S. Pankanti, R. Bolle, An identity-authentication system using fingerprints, Proceedings of the IEEE, Vol 85 (9) pp1365–1388, 1997.

[10] R.P. Wildes, Iris recognition: "An emerging biometric technology", Proceedings of the IEEE Vol. 85 (9) pp 1348–1363, 1997.

[11] R.B. Hill, “Retina Identification, Biometrics: Personal Identification in Networked Society”, Kluwer Academic, 1999.

[12] W. Zhao, R. Chellappa, P.J. Phillips, A. Rosenfeld, “Face recognition: A literature survey”, ACM Computing Surveys Vol. 35 (4) pp x 399-458, 1997.

[13] V.S. Nalwa,”Automatic on-line signature verification”, Proceedings of the IEEE Vol. 85 (2) pp 215-239, 1997.

[14] A. Eriksson, P. Wretling, “How flexible is the human voice? A case study of mimicry”, Proceedings of EUROSPEECH Vol. 2, pp 1043-1046, 1997.

[15] B. Chiraz, C. R.G, D. L.S, “Gait recognition using image self-similarity”, EURASIP Journal on Applied Signal Processing pp 572-585, 2004.

[16] S. Pankanti, S. Prabhakar, A.K. Jain, “On the individuality of fingerprints”, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 24 (8) pp 1010-1025, 2002.

[17] A.K. Jain, S. Prabhakar, S. Pankanti, “On the similarity of identical twin fingerprints”, Pattern Recognition Vol. 35 (11), pp 2653-2663, 2002.

[18] A.K. Jain, L. Hong, R.M. Bolle, “On-line fingerprint verification”, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 19 (4) pp 302-314, 1997.

[19] 19 Jain, L.C. et al. "Intelligent Biometric Techniques in Fingerprint and Face Recognition", Boca Raton, FL: CRC Press, 1999.

[20] R. Mukundan and K. R. Ramakrishnan, “Moment Functions in Image Analysis Theory and Applications”, World Scientific Publishing Co. Pte. Ltd. ISBN 981-02-3524-0, pp.81-85, 1998.

[21] Holland, John H., “Adaptation in Natural and Artificial Systems”, University of Michigan Press, Ann Arbor. 1975.

[22] Ralph Merkle, Martin Hellman, "On the Security of Multiple Encryption", Communications of the ACM, Vol. 24, No 7, pp 465-467, July 1981.

[23] William Stallings, "cryptography and Network Security: Principles and Practice", Prentice Hall, 2003.

VIII. VITA

The author is an associate professor in computer science at Benha University, Egypt. He holds a PhD in Informatics Technology from Czech Technical University in Prague, 2004. He is an author of many papers that have been published in many refereed international Journals. His areas of interest include but not limited to Digital Image Processing, Multimedia Networks, Intrusion detection, Data Mining, Data Clustering and classification, pattern Recognition, character recognition, fuzzy clustering, artificial Neural Networks, Expert systems, and Software Engineering.