Feature Based Encryption Technique For Securing Forensic Biometric Image Data Using AES and Visual Cryptography*

Quist-Aphetsi Kester[1234], Laurent Nana[2], Anca Christine Pascu[23], Sophie Gire[2], J. M. Eghan[4], Nii Narku Quaynor[4],

[1]Faculty of Informatics, Ghana Technology University College, Accra, Ghana
[2]Lab-STICC (UMR CNRS 6285), European University of Brittany, University of Brest, 29238 Brest cedex France
[3]HCTI EA 4249 and Lab-STICC (UMR CNRS 6285) European University of Brittany, UBO, France, Brest, 29238 Brest
[4]Department of Computer Science and Information Technology, University of Cape Coast, Cape Coast, Ghana
kquist-aphetsi@univ-brest.fr

***Abstract*—with the current emergence of biometric image data applications on devices such as smart phones, security cameras, personal computers etc, there is a need for securing the image templates obtained from crime scenes as well as such devices before storing them in locations such as the cloud etc. In this paper, we proposed an encryption technique of securing the biometric image data collected from devices with an approach of feature based on encryption technique for securing forensic biometric image data using AES and visual cryptography method.**

Keywords-cryptography ; AES; image; visual cryptography

I. INTRODUCTION

Imaging security and biometrics are two heavily connected areas in present day information security age. The quick evolution of biometrics with its usage in surveillance, verification and access control devices has raised the need of securing biometric data [1]. A majority of this image data from the biometric devices is visual, which has lead to intensive development of image security techniques for biometric applications [2]. Securing life forensic data over a communication channel and storing them must preserve the evidence by avoiding changes to the original image features such as the pixel values [3].

We chose Visual cryptography because it is the technique used to encrypt the data which is in the form of visual information such as images. Since the biometric templates stored in the database is usually in the form of images, the visual cryptography can be efficiently employed to encrypt the templates from attack [4].

* This work is supported by Lab-STICC (UMR CNRS 6285) at UBO France, AWBC Canada, Ambassade de France-Institut Français-Ghana, DCSIT University of Cape Coast,

The AES, Rijndael [5], is an adopted standard for the encryption of data established by the U.S. National Institute of Standards and Technology (NIST) in 2001[6] and it is made up of a set of processes of transformations [7]. The AES-Advanced Encryption Standards has a key schedule for 128-bit, 192-bit, and 256-bit encryption and has all underwent thorough security strength analysis [8][9][10][11][12], application [13][14][15][16][17] and advancement by engaging several hybrid approaches [18][19][20][21][22] with other ciphers in order to test its security strength and also improve its strength. The hybrid approaches of the effective implementation of AES and other strong cryptographic tend to make it stronger and resistive to attacks [23].

This paper proposes an approach of encryption for securing forensic biometric image data using AES and visual cryptography. The encryption is done by engaging visual cryptographic encryption techniques based on image shares and transposition of the share. A key is extracted from the image and then encrypted using AES before using its engagement in the encryption process. At the end of the process, it has been observed that there was no pixel expansion hence there was no loss in image quality. The paper has the following structure: section II Methodology, section III Results and analysis, and section IV concluded the paper.

II. METHODOLOGY

In this paper, we presented an approach of encryption of images using AES and visual cryptography. The key was extracted from the image features and the AES-256 algorithm was used to generate the key used for the image encryption based on the extracted key. The Rijndael algorithm is a symmetric block cipher that supports key sizes of 128, 192 and 256 bits, with data handled in 128-bit blocks [24]. The pixel values of the images to be encrypted were encrypted using n-share visual cryptographic technique. The encryption process experienced no loss of pixel values during the process.

978-1-4799-7600-3/14 $31.00 © 2014 IEEE
DOI 10.1109/AIMS.2014.65

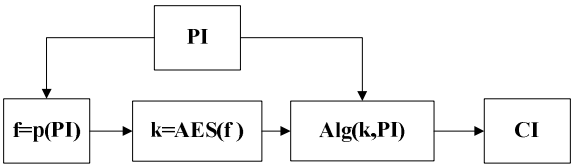

Figure 1. The summary of the processes engaged.

PI= Plain Image
f= extracted feature
k=key obtained from the extracted feature using the AES
Alg= visual encryption algorithm employed
CI= Ciphered Image

A. *The Feature extraction*

The features extracted from the image were the entropy and the geometrical mean.

Let *I*= an image=*f (R, G, B)*
I is a color image of *m x n x 3* arrays

$$\begin{pmatrix} R & G & B \\ r_{i1} & g_{i2} & b_{i3} \\ \vdots & \vdots & \vdots \\ r_{m1} & g_{m2} & b_{m3} \end{pmatrix}$$

(R, G, B) = m x n
Where *R, G, B* $\in I$
(R o G) i j = (R) ij. (G) ij
Where *r_11* = first value of *R*
r= [ri1] (i=1, 2... m)
$x \in r\_i1 : [a, b] = \{x \in I: a \le x \ge b\}$
a=0 and *b=255*
R= r= I (m, n, 1)
Where *g_12* = first value of *G*
g= [gi2] (i=1, 2... m)
$x \in g : [a, b] = \{x \in I: a \le x \ge b\}$
a=0 and *b=255*
G= g= I (m, n, 1)
And *b_13* = first value of *B*
g= [bi3] (i=1, 2... m)
$x \in b\_i1 : [a, b] = \{x \in I: a \le x \ge b\}$
a=0 and *b=255*
B=b= I (m, n, 1

The geometric mean is

$$m = \left[\prod_{i=1}^{n} x_i\right]^{\frac{1}{n}} \qquad (1)$$

Where $x \in b\_i1 : [a, b] = \{x \in I: a \le x \ge b\}$

δ(xi) = entropy(I) which is the entropy of the plain and it is is a scalar value representing the entropy of grayscale image The entropy is a statistical measure of randomness that can be used to characterize the texture of the input image.

Entropy is defined as

$$\delta(\text{xi}) = -\sum_{\eta=0}^{\varepsilon-1} \Psi(xi).\ log2\ (\Psi(xi)) \qquad (2)$$

Where:
δ= Entropy of image
ε = Gray value of an input image *(0-255).*
$\Psi(\eta)$ = Probability of the occurrence of symbol η

B. *The Rijndael-Advanced Encryption Standard Algorithm*

All the input values, output values and cipher key values for the Advanced Encryption Standards are sequences containing the following bits values 128, 160, 192, 224 or 256 bits with the same cipher block size. The AES has a fixed block size at 128 bits. The number *'i'* associated with a bit will hence be in one of the five ranges $0 \le i < 128$, $0 \le i < 160$, $0 \le i < 192$, $0 \le i < 224$ or $0 \le i < 256$. All the bit sequences are interpreted as one-dimensional arrays of 8-bit bytes where byte *n* consists of the sub-sequence 8*n* to 8*n* +7. The arrays are denoted by *a*, the *n*'the byte will be referred to as a_n or *a[n]*, where *n* is in one of the ranges $0 \le n < 16$, $0 \le n < 20$, $0 \le n < 24$, $0 \le n < 28$ or $0 \le n < 32$ [25]. The bit order within a byte has the value 7-*k*, where *k* is the bit's index, and the bit with order in a byte *b* will be denoted by b_i. Internally bytes are polynomial representations of finite field elements:

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0 = \sum_{i=0}^{7} b_ix^i \qquad (3)$$

The state array, s, and each distinct byte has two indexes: row number *r* with range $0 \le r < 4$, and column number *c*, with range $0 \le c < Nc$. Hence we have s_{rc} *or s[r,c]* and the range for c is $0 \le c < 4$ has a fixed value of 4.

in_0	in_4	in_8	in_{12}	…
in_1	in_5	in_9	in_{13}	…
in_2	in_6	in_{10}	in_{14}	…
in_3	in_7	in_{11}	in_{15}	…

(a) Cipher input bytes

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$	…
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	…
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$	…
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$	…

(b) Cipher state array

out_0	out_4	out_8	out_{12}	…
out_1	out_5	out_9	out_{13}	…
out_2	out_6	out_{10}	out_{14}	…
out_3	out_7	out_{11}	out_{15}	…

(c) Cipher output bytes

Figure 2. Input(a), and output(c) from the cipher state(b) array.

Hence the input array is copied to the state array according to the schemes:

$$\left.\begin{array}{l} s[r,c] = in[r+4c] \\ out[r+4c] = s[r,c] \end{array}\right\} \quad 0 \le r < 4 \text{ and } 0 \le c < Nc \qquad (4)$$

The 128 bit key is then expanded as an array of 44 entries of 32 bits words; 4 distinct words serve as a round key for each round; key schedule relies on the S-box. The method composed of three layers– Linear Diffusion, Non-linear Diffusion and Key Mixing. The Cipher is as follows:

Cipher(byte in*[4*Nc]*, byte out*[4*Nc]*, word *k[Nn+1,Nc], Nc, Nn)*
Begin
byte state*[4,Nc]*
state = in
XorRoundKey(state, *k[0,-], Nc)*
for *round = 1* step *1* to *Nn – 1*
SubBytes*(state, Nc)*
ShiftRows*(state, Nc)*
MixColumns*(state, Nc)*
XorRoundKey*(state, k[round,-], Nc)*
end for
SubBytes*(state, Nc)*
ShiftRows*(state, Nc)*
XorRoundKey*(state, k[Nn,-], Nc)*
out = state
end

The notation *k[Nn+1,Nc]* above implies that the array *k* contains *Nn + 1* individual round keys that are each arrays of *Nc* words.

The SubBytes(State, S-box) above is as follows:

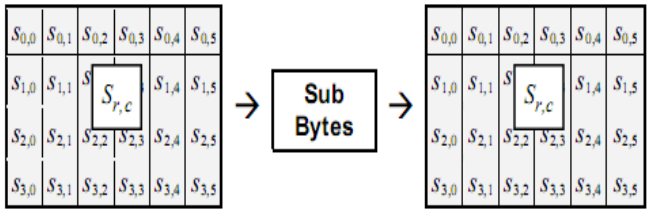

Figure 3. SubBytes acts on every byte in the state in isolation.

The transformation is defined by

$$b_i' = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i \qquad (5)$$

Where $0 \le i < 8$ and bi is the bit i of the byte and ci is the bit i of a byte c.

ShiftRows(State) above is as follows:

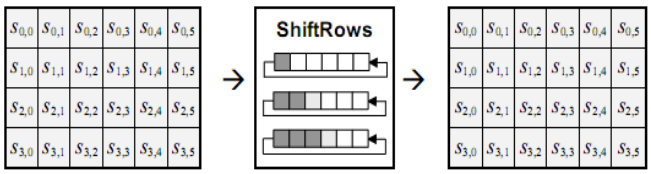

Figure 4. ShiftRows acts independently on rows in the state.

$$s'_{r,c} = s_{r,[c+h(r,Nc)] \bmod Nc} \text{ for } 0 \le r < 4 \text{ and } 0 \le c < Nc \qquad (6)$$

Where the shift amount *h(r,Nc)* depends on row *r* number and block length and block size, *Nc*, is fixed at 4

The MixColumns(State) above is represented as follows: The MixColumns transformation operates separately on every column of the state and treats each column as a four-term polynomial

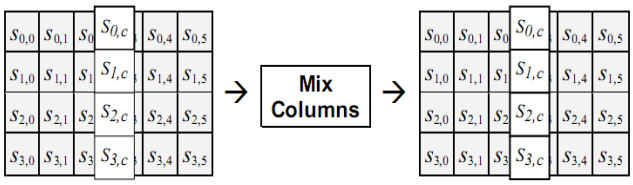

Figure 5. MixColumns acts independently on each column in the state.

$$\begin{bmatrix} s_{3,c} \\ s_{2,c} \\ s_{1,c} \\ s_{0,c} \end{bmatrix}' = \begin{bmatrix} \{02\} & \{01\} & \{01\} & \{03\} \\ \{03\} & \{02\} & \{01\} & \{01\} \\ \{01\} & \{03\} & \{02\} & \{01\} \\ \{01\} & \{01\} & \{03\} & \{02\} \end{bmatrix} \begin{bmatrix} s_{3,c} \\ s_{2,c} \\ s_{1,c} \\ s_{0,c} \end{bmatrix} \text{ for } 0 \le c < Nc \qquad (7)$$

XorRoundKey above is presented as follows: with this the transformation words from the key schedule are each added into the columns of the state as shown in (8).

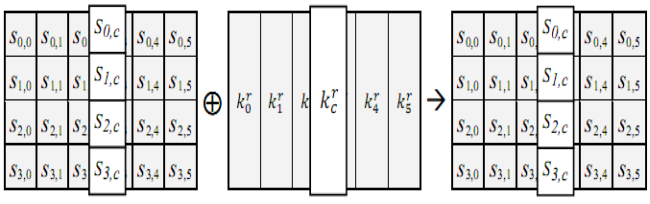

Figure 6. Words from the key schedule are XOR'd into columns in the state.

$$[b_{3c}, b_{2c}, b_{1c}, b_{0c}]' = [b_{3c}, b_{2c}, b_{1c}, b_{0c}] \oplus [k_{round,c}] \text{ for } 0 \le c < Nc \qquad (9)$$

Where the round key words $k_{round,c}$ (as K^c_r in the diagram above). The round number, round, is in the range $0 \leq round < Nc$, with the value of 0 being used to denote the initial round key that is applied before the round function.

Let Sk=f(k ,Ar) where k is the Secret Passphrase and Ar is the AES result obtained from the encryption results.

C. *The Visual Cryptographic Encryption Process*

The image encryption process engaged *Alg(k,PI)* in ciphering the image and displacing the pixel values using a visual cryptographic technique.

Engagement *k* was used encrypt the plain image.

Start
Reading the image data,
Let PI= f (R, G, B)
new_image=imread(PI);
PI is a color image of m x n x 3 arrays
(R, G, B) = m x n
Where R, G, B $\in$PI
(R o G) i j = (R) ij. (G) ij
Where r_11 = first value of R
r= [ri1] (i=1, 2… m)
x $\in$ r_i1 : [a, b]= {x $\in$ I: a $\leq$ x $\geq$ b}
a=0 and b=255
R= r= I (m, n, 1)
Where g_12 = first value of G
g= [gi2] (i=1, 2... m)
x $\in$ g : [a, b]= {x $\in$ I: a $\leq$ x $\geq$ b}
a=0 and b=255
G= g= I (m, n, 1)
And b_13 = first value of B
g= [bi3] (i=1, 2... m)
x $\in$ b_i1 : [a, b]= {x $\in$ I: a $\leq$ x $\geq$ b}
a=0 and b=255
B=b= I (m, n, 1)
kn=abs(mod((c*x(k)),p))
for i: Δi:*kn*
Let *t'(i,j)* =Transpose of r(i,j)
t'(i,j)= f(r',c,p);
Let *y'(i,j)* =Transpose of y(i,j)
y'(i,j)= f(g',c,p);
Let *u'(i,j)* =Transpose of u(i,j)
u'(i,j)= f(b',c,p);
end
Transformion of *t'(i,j) into f(t'(i,j)*,c,p))
r= f(t'(i,j),c,p))*=f(r, c, p)*
Transformion of *y'(i,j)into f(y'(i,j)*,c,p))
g= f(y'(i,j),c,p))*=f(g, c, p)*
Transformion of *u'(i,j) into f(u'(i,j)*,c,p))
b= f(u(i,j),c,p))*=f(b, c, p)*
CI= f (3,r,g,b);
end

III. RESULTS AND ANALYSIS

AES is a block cipher supporting any combination of data and key size of 128, 192, and 256 bits. However, AES merely allows a 128 bit data length that can be divided into four basic operation blocks. Each cipher produced by AES uses several rounds of fixed operations to achieve desired output which determines its security level which is measured in terms of diffusion (strict avalanche criterion (SAC)) and confusion hence the number of rounds are chosen such that the algorithm provides the SAC value.

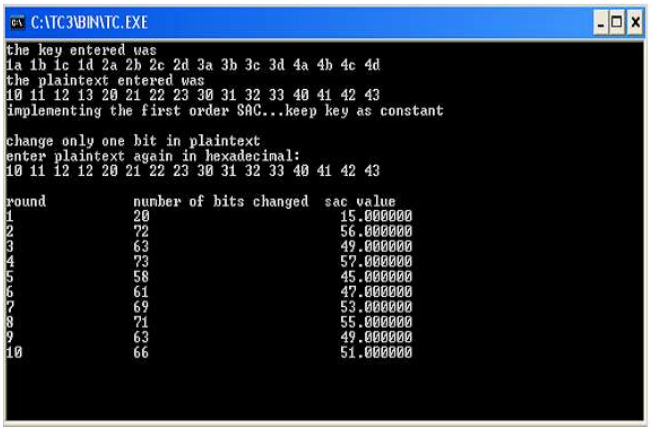

Figure 7. Results of Avalanche effect on AES

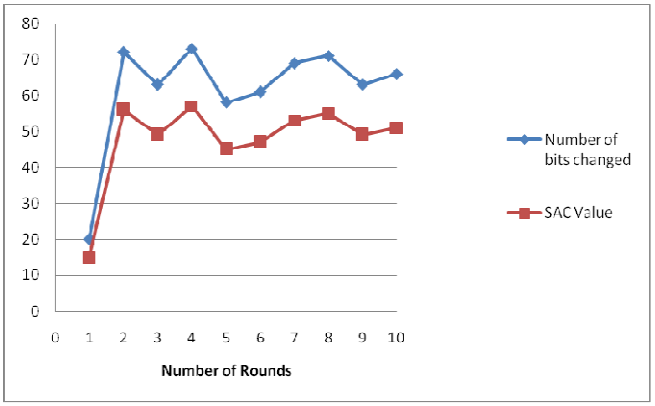

Figure 8. A graph of the results in figure7.

In figure 7 and as seen in the graph as well in figure 8, the end of 1st round, 20 bits of cipher value have changed out of 128-bit cipher text. This resulted in an SAC value of 15% and the SAC at the end was 51%.

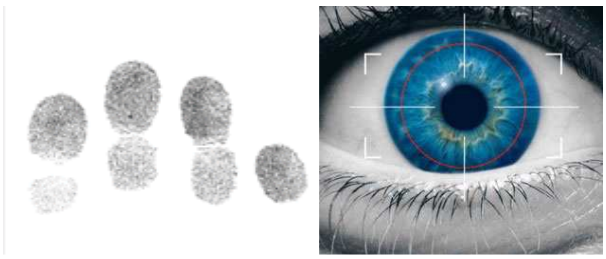

Figure 9. A biometric finger print image and an iris image.

Figure 10. A biometric finger print image.

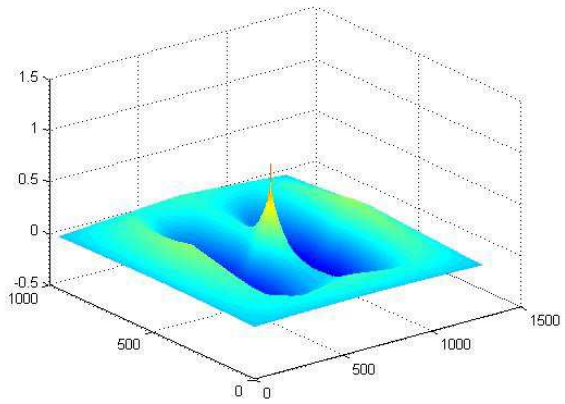

Figure 11. The graph of the normalized cross-correlation of the matrices of the plain image in figure 10.

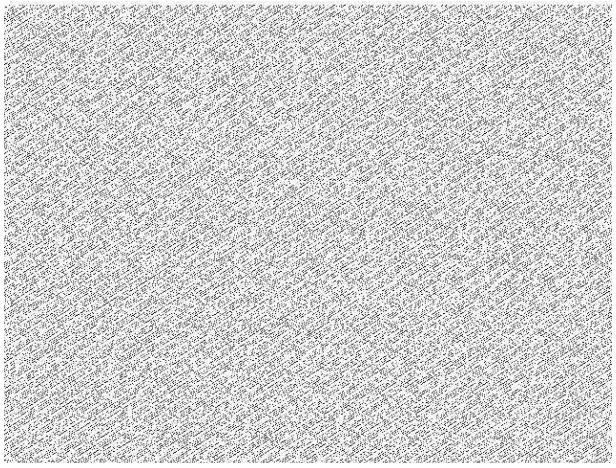

Figure 12. A ciphered biometric finger print image of figure 10.

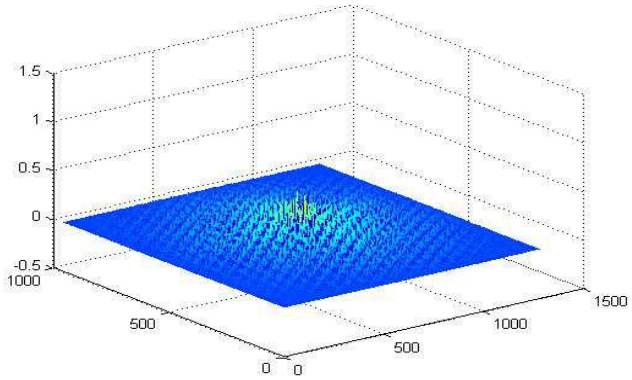

Figure 13. The graph of the normalized cross-correlation of the matrices of the ciphered image in figure 11.

The normalized cross-correlation of the matrices of is

$$\gamma(u,v)=\frac{\sum_{x,y}\left[f(x,y)-\bar{f}_{u,v}\right]\left[t(x-u,y-v)-\bar{t}\right]}{\left\{\sum_{x,y}\left[f(x,y)-\bar{f}_{u,v}\right]^2\sum_{x,y}\left[t(x-u,y-v)-\bar{t}\right]^2\right\}^{0.5}} \quad (10)$$

f is the mean of the template, $\bar{t}$ is the mean of in the region under the template. $\bar{f}_{u,v}$ is the mean of $f(u,v)$ in the region under the template.

TABLE I. GEOMETIRIC MEAN AND ENTROPY VALUES

xi	*m*	δ(xi)
PI(R)	2.5910	214.0351
PI(G)	2.5910	214.0351
PI(B)	2.5910	214.0351
CI(R)	2.5910	214.0351
CI(G)	2.5910	214.0351
CI(B)	2.5910	214.0351

The plain image in figure 10 was encrypted by implementing the algorithm in MATLAB. The graph of the normalized cross-correlation of the matrices of the plain image was plotted as shown in figure 10 as well as its ciphered image as shown in figure 13. Table 1 has the geometric mean value and the entropy value of the respective RGB values for both the plain image and the ciphered image.

IV. CONCLUSION

The basic design and strength of an encryption algorithm such as AES depends on diffusion and confusion. This engaged the use of Advanced Encryption algorithm and visual cryptography in securing forensic biometric images.

The Entropy value and the arithmetic mean value of the pixels of the plain image and the ciphered image was found to be 2.5910 and 214.0351 respectively, this means that there was no pixel expansion after the encryption process. Hence the quality of the image will remain the same after decryption. The advantage of the engagement of the Rijandel is that, it is resistant to linear and differential cryptanalysis .

ACKNOWLEDGMENT

This work is supported by Lab-STICC (UMR CNRS 6285) at UBO France, AWBC Canada and the DCSIT, and UCC. And also special thanks go to Dominique Sotteau (formerly directeur de recherche, Centre national de la recherche scientifique (CNRS) in France and head of international relations, Institut national de recherche en informatique et automatique, INRIA) and currently, she is the Scientific counselor of AWBC.

REFERENCES

[1] Ion Marqués and Manuel Graña. 2012. Image security and biometrics: a review. In Proceedings of the 7th international conference on Hybrid Artificial Intelligent Systems - Volume Part II (HAIS'12), Emilio Corchado, Václav Snášel, Ajith Abraham, Michał Woźniak, and Manuel Graña (Eds.), Vol. Part II. Springer-Verlag, Berlin, Heidelberg, 436-447. DOI=10.1007/978-3-642-28931-6_42 http://dx.doi.org/10.1007/978-3-642-28931-6_42.

[2] Charles L. Wilson, Patrick J. Grother, and Ramaswamy Chandramouli. 2007. SP 800-76-1. Biometric Data Specification for Personal Identity Verification. Technical Report. NIST, Gaithersburg, MD, United States.

[3] Kester, Q. A., Nana, L., Pascu, A. C., Gire, S., Eghan, J. M., & Quaynnor, N. N. (2014). A hybrid encryption technique for securing biometric image data based on feistel network and RGB pixel displacement. In Recent Trends in Computer Networks and Distributed Systems Security (pp. 530-539). Springer Berlin Heidelberg.

[4] Aeloor, Deepak, and Amrita A. Manjrekar. "Securing Biometric Data with Visual Cryptography and Steganography." Security in Computing and Communications. Springer Berlin Heidelberg, 2013. 330-340

[5] "Announcing the ADVANCED ENCRYPTION STANDARD (AES)". Federal Information Processing Standards Publication 197. United States National Institute of Standards and Technology (NIST). November 26, 2001.

[6] Daemen, Joan; Rijmen, Vincent (9/04/2003). "AES Proposal: Rijndael". National Institute of Standards and Technology. p. 1.

[7] AlMarashda, K.; AlSalami, Y.; Salah, K.; Martin, T., "On the security of inclusion or omission of MixColumns in AES cipher," Internet Technology and Secured Transactions (ICITST), 2011 International Conference for , vol., no., pp.34,39, 11-14 Dec. 2011

[8] Chong Hee Kim, "Differential Fault Analysis against AES-192 and AES-256 with Minimal Faults," Fault Diagnosis and Tolerance in Cryptography (FDTC), 2010 Workshop on , vol., no., pp.3,9, 21-21 Aug. 2010doi: 10.1109/FDTC.2010.10

[9] Floissac, N.; L'Hyver, Y., "From AES-128 to AES-192 and AES-256, How to Adapt Differential Fault Analysis Attacks on Key Expansion," Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on , vol., no., pp.43,53, 28-28 Sept. 2011doi: 10.1109/FDTC.2011.15

[10] Ali, S.S.; Mukhopadhyay, D., "A Differential Fault Analysis on AES Key Schedule Using Single Fault," Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on , vol., no., pp.35,42, 28-28 Sept. 2011doi: 10.1109/FDTC.2011.10

[11] Yongzhuang Wei, Jiqiang Lu, and Yupu Hu. 2011. Meet-in-the-middle attack on 8 rounds of the AES block cipher under 192 key bits. In Proceedings of the 7th international conference on Information security practice and experience (ISPEC'11), Feng Bao and Jian Weng (Eds.). Springer-Verlag, Berlin, Heidelberg, 222-232.

[12] H. Gilbert, M. Minier, A collision attack on 7 rounds of Rijndael, in Proceedings of the Third AES Candidate Conference (AES3), New York, USA (2000), pp. 230–241

[13] Dino Oliva, Rainer Buchty, and Nevin Heintze. 2003. AES and the cryptonite crypto processor. In Proceedings of the 2003 international conference on Compilers, architecture and synthesis for embedded systems (CASES '03). ACM, New York, NY, USA, 198-209. DOI=10.1145/951710.951738

[14] Wen-Ai Jackson and S. Murphy. 2007. Projective aspects of the AES inversion. Des. Codes Cryptography 43, 2-3 (June 2007), 167-179. DOI=10.1007/s10623-007-9059-4

[15] Thomas Baignères and Serge Vaudenay. 2005. Proving the security of AES substitution-permutation network. In Proceedings of the 12th international conference on Selected Areas in Cryptography (SAC'05), Bart Preneel and Stafford Tavares (Eds.). Springer-Verlag, Berlin, Heidelberg, 65-81. DOI=10.1007/11693383_5

[16] Sheikh Muhammad Farhan, Shoab A. Khan, and Habibullah Jamal. 2009. An 8-bit systolic AES architecture for moderate data rate applications. Microprocess. Microsyst. 33, 3 (May 2009), 221-231. DOI=10.1016/j.micpro.2009.02.013

[17] Mao-Yin Wang; Chih-Pin Su; Chia-Lung Horng; Cheng-Wen Wu; Chih-Tsun Huang, "Single- and Multi-core Configurable AES Architectures for Flexible Security," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.18, no.4, pp.541,552, April 2010doi: 10.1109/TVLSI.2009.2013231

[18] Vishnu, M. B.; Tiong, S.K.; Zaini, M.; Koh, S. P., "Security enhancement of digital motion image transmission using hybrid AES-DES algorithm," Communications, 2008. APCC 2008. 14th Asia-Pacific Conference on , vol., no., pp.1,5, 14-16 Oct. 2008

[19] Kshirsagar, R.V.; Vyawahare, M.V., "FPGA Implementation of High Speed VLSI Architectures for AES Algorithm," Emerging Trends in Engineering and Technology (ICETET), 2012 Fifth International Conference on , vol., no., pp.239,242, 5-7 Nov. 2012doi: 10.1109/ICETET.2012.53

[20] Ganesh, E.S.; Velayutham, R.; Manimegalai, D., "A secure software implementation of nonlinear AES S-box with the enhancement of biometrics," Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on , vol., no., pp.927,932, 21-22 March 2012doi: 10.1109/ICCEET.2012.6203796

[21] Sandeep Kumar Namini. 2012. A Secure Communication for Wireless Sensor Networks: Through Hybrid (AES +Ecc) Algorithm. LAP Lambert Academic Publishing, , Germany.

[22] Xiang Li; Junli Chen; Dinghu Qin; Wanggen Wan, "Research and realization based on hybrid encryption algorithm of improved AES and ECC," Audio Language and Image Processing (ICALIP), 2010.

[23] Kester, Q.-A.; Nana, L.; Pascu, A.C., "A Novel Cryptographic Encryption Technique for Securing Digital Images in the Cloud Using AES and RGB Pixel Displacement," Modelling Symposium (EMS), 2013 European , vol., no., pp.293,298, 20-22 Nov. 2013 doi: 10.1109/EMS.2013.51.

[24] [33] Jamil, T., "The Rijndael algorithm," Potentials, IEEE , vol.23, no.2, pp.36,38, April-May 2004 doi: 10.1109/MP.2004.1289996

[25] Gladman, Brian. "A specification for Rijndael, the AES algorithm." at fp. gladman. plus. com/cryptography_technology/rijndael/aes. spec 311 (2001): 18-19.