8th. World Congress on Computational Mechanics (WCCM8)
5th. European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2008)
June 30 – July 5, 2008
Venice, Italy

FINGERPRINT ENCRYPTION USING FRACTIONAL FOURIER TRANSFORM

***Juan M. Vilardy[1], Cesar O. Torres[2] and Lorenzo Mattos[3]**

[1] Universidad Popular del Cesar and COLCIENCIAS – Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología "Francisco José de Caldas" Valledupar, Cesar - Colombia
vilardy.juan@unicesar.edu.co

[2] Universidad Popular del Cesar
Valledupar, Cesar - Colombia
cesartorres@unicesar.edu.co

[3] Universidad Popular del Cesar
Valledupar, Cesar - Colombia
mattos.lorenzo@caramail.com

Key Words: *Fingerprint, Phase Encryption/Decryption, Continue Fractional Fourier Transform, Discrete Fractional Fourier Transform, Random Phase Masks.*

ABSTRACT

In the present paper a digital algorithm was developed to make digital encryption applied to fingerprints using the fractional Fourier transform. The encryption process used [1], take the grayscale image (fingerprint) and it's placed as the phase of a complex exponential, then is transformed five times and multiplied in intermediate steps by four random phase masks statistically independent, thus to obtain the encrypted fingerprint. The fractional orders applied in the transforms are decimal numbers between zero and four, generated from a key alphanumeric of six to ten characters. In the decryption process for the coding fingerprint, the encryption procedure is applied in the inverse sense to the conjugated complex of the encrypted fingerprint, then is taken the negative of the phase of the resulting image of the decryption process and the original fingerprint is obtained this way that had been encrypted. In the implemented cryptographic algorithm nine keys are used, constituted by five fractional orders and four random phase masks, all these keys are necessary for a correct decryption providing a high level of security to fingerprint for a given application.

ENCRYPTION AND DECRYPTION PROCESS

A. Encryption Process

FIGURE. 1 shows the encryption process [1, 2, 3, 4] (DFrFT, Discrete Fractional Fourier Transform [5]). Upon carrying out the encrypting process of the fingerprint, the encrypted fingerprint hides the totality of the information contained therein, as seen in figure 2(b) and 2(c), the distribution of intensities of the encrypted fingerprint varies when changing the keys (fractional orders, e.i key alphanumeric) and the matrixes that contains the real and imaginary part of the encrypted fingerprint are saving with 16 bits (quantization level) in a *RGB* image; When the decryption process is done with the keys and masks correct, we recover the original fingerprint with loses not visible to the human eye, as shown in figure 2(d). If the keys used in the decryption process are not equal to the keys used in the encryption process, the fingerprint will not be recovered, see figure 2(e). And lastly, if the masks used in the decryption process are not the same

masks used in the encryption, or if this are the same but are placed in different intermediate steps in the decryption, the fingerprint will not be recovered, as it shows in the figure 2(f).

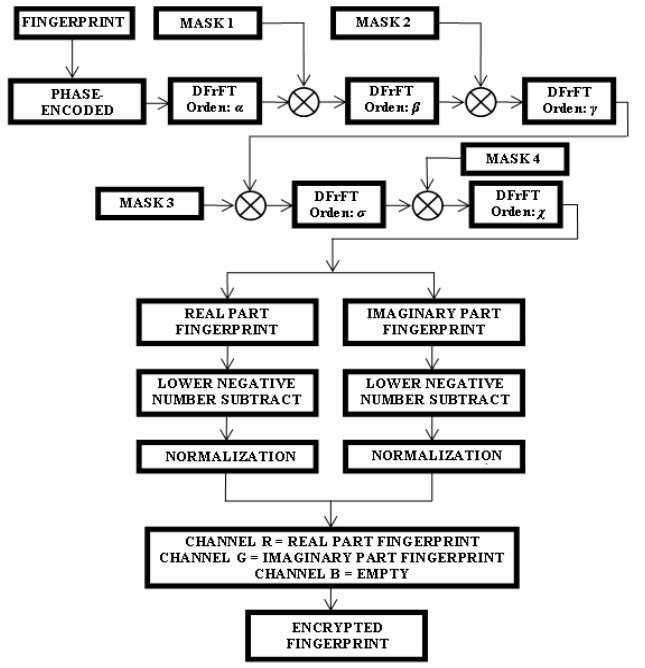

FIGURE 1. Block diagram of the fingerprint encryption process.

The Mean Square Error (MSE) between the input fingerprint and our decrypted fingerprint is calculated to validate the reliability of this algorithm. The Mean Square Error can be defined by the difference between the energy of the fingerprint decrypted and encrypted, i.e.

$$MSE = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left| I_1[i,j] - I[i,j] \right|^2 \tag{1}$$

Where $I(i, j)$ and $I_1(i, j)$ are the matrixes element of the input fingerprint and our decrypted fingerprint at the pixel (i, j), respectively, and $M \cdot N$ is the size of the fingerprint.

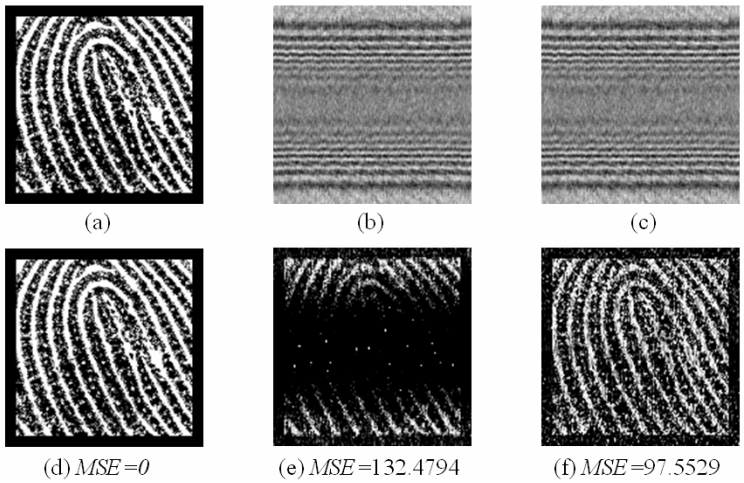

FIGURE 2. (a) The input fingerprint; encrypted fingerprint with the key alphanumeric ***'vilardy121' (α=1.4149, β=1.715, γ=1.0849, σ=1.0521, y χ=1.18001)*:** (b) real part, (c) imaginary part; the decrypted result: (d) with the key alphanumeric and masks corrects, (e) with one wrong on key alphanumeric **(γ) *'vijardy121' (α=1.4149, β=1.715, γ=1.0649, σ=1.0521, y χ=1.18001),*** and corrects masks, (f) with wrong masks and corrects keys.

B. Decryption Process

The decryption process is the same encryption process, but in the inverse sense applied to the complex conjugate of the encrypted fingerprint and lastly we take the negative of the phase of the matrix resulting from this process, thus to obtain the fingerprint that initially was coding.

EXPERIMENT RESULTS

For the implementation of the digital algorithm in Matlab® v.7.4, we used a Pentium 4 IBM computer with a 2.23 GHz processor and 256 MB RAM, obtaining the following encryption and decryption times for a fingerprint of 400x300 pixels, Encryption Time: 0.17233 Seconds. Decryption Time: 0.17465 Seconds.

Finally for the key space analysis of the algorithm implemented, it considered only printable characters of the ASCII code (codes from 32 to 126 in decimal, known as printable character, and represent the character space, letters, digits, punctuation signs and several symbols), now as the key alphanumeric used in the encryption algorithm has 10 ASCII characters, to succeed in a brute force attack on the fingerprint encrypted there are $(95)^{10}$ possibilities, only to the key alphanumeric and assuming known the four random phase masks. **For the test fingerprint used in this paper, the decryption time is *0.17233 seconds* and therefore, is take *10.31803 x 10^{18} seconds* to have a successful bruteforce cracking, this is equivalent to *3.27182 x 10^{11}* years!.**

CONCLUSION

Using fractional Fourier transform in the digital fingerprint encryption greatly increases safety parameters in the encrypted fingerprint, due to the sensitivity of the key alphanumeric (fractional orders) used, and moreover the four random phase masks utilized, increase security much more for any cryptanalyst trying to decrypt the fingerprint without authorization.

REFERENCES

[1] J. Vilardy, J. Calderon, C. Torres, L. Mattos, "Digital Images Phase Encryption Using Fractional Fourier Transform", *Proceedings of the Electronics, Robotics and Automotive Mechanics Conference*, IEEE, ISBN: 0-7695-2569-5/06, 2006.

[2] K. N. Naveen, J. Joby, S. Kehar, "Fully phase-encrypted memory using cascade extended fractional Fourier transform", *ELSEIVER, Optics and Lasers in Engineering* 42 (2004) 141-151.

[3] C. Candan, "dFRT: The Discrete Fractional Fourier Transform, A Matlab Program, 1998", http://www.ee.bilkent.edu.tr/~ haldun/dFRT.m.

[4] A. Bultheel, H. Martínez, "Computation of the Fractional Fourier Transform", *ELSEIVER, Applied and Computational Harmonic Analysis,* 16 (2004) 182-202.

[5] C. Candan, M. A. Kutay, H. M. Ozaktas, "The Discrete Fractional Fourier Transform", *IEEE Transactions on signal processing,* VOL. 48, No.5, MAY 2000.