



Techniques in Image Steganography using Famous Number Sequences

N. Aroukatos¹, K. Manes², S. Zimeras³, F. Georgiakodis⁴

¹Department of Informatics, University of Piraeus, Piraeus, Greece
naroykat@unipi.gr

²Department of Informatics, University of Piraeus, Piraeus, Greece
kmanes@unipi.gr

³University of Aegean, Samos, Greece
zimste@aegean.gr

⁴Department of Informatics, University of Piraeus, Piraeus, Greece
fotis@unipi.gr

ABSTRACT

In the last decades, Steganography techniques have been applied in a variety of data files. The need of copyright protection in Music, in Photography e.t.c pushed the software companies to develop many steganographic systems which they use, in various areas, e.g., in digital assets (DRM). In this paper, we propose a number of methods for image steganography using Catalan numbers and Lucas numbers and we show that they produce better results than the technique using Fibonacci numbers. We are able to use Catalan and Lucas numbers since we have proved that these sets of numbers satisfy similar conditions to those of the Theorem of Zeckendorf.

Indexing terms/Keywords

Zeckendorf; Fibonacci numbers; Steganography; Catalan numbers; Lucas numbers.

Council for Innovative Research

Peer Review Research Publishing System

Journal: International Journal of Computers & Technology

Vol 11, No.3

editor@cirworld.com

www.cirworld.com, member.cirworld.com

INTRODUCTION

The exchange of information is essential for the development of civilization. The discovery and evolution of methods that could make transmission of information secure attracted people since antiquity. Over the centuries people discovered and developed techniques which evolved into the sciences of Cryptography and Steganography. Cryptography disguises the message to be transmitted so that only the intended recipient is able to read it, while steganography hides the message by embedding it within other, seemingly harmless, messages. Steganography dates back to the ancient Greece but only lately (late 20th century) it began being researched for scientific reasons. Today, it is widely used in Telecommunications, Industry [1], Medicine [2] and in the practice of hiding strongly encrypted data. Steganography, in contrast to Cryptography, is not trying to make a message incomprehensible for an invalid person [3], [4], but to hide its existence, using a cover, e.g., by incorporating the message to be transmitted into an image.

The LSB method

In recent years, the LSB (Least Significant Bit) method became one of the most important steganographic methods for hiding data within images [5]. In a $N \times M$ color RGB image, with 8 bit color depth, each pixel assumes an integer value x on the closed interval $[0, 255]$ for each color (Red, Green, Blue). The number x represents the density of the color and it is encoded by an 8 bit binary word $b_7b_6 \dots b_0$, where $x = \sum_{i=0}^7 b_i \cdot 2^i$ and $b_i \in \{0,1\}$. For example $91 = 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 00100011_2$.

This definition of x allows the decomposition of an image into a collection of binary images by separating the bits into 8 bit planes. In the classical LSB embedding methods, the secret message is inserted into the least significant bitplane, i.e., the 8th bitplane, of the cover image, either by directly replacing those bits or by modifying them using a particular "inverse" function [6] (Figure 1). The embedding strategy can also be based on sequential insertion or selective embedding of the message in "noisy" areas or random scattering throughout the image [7].

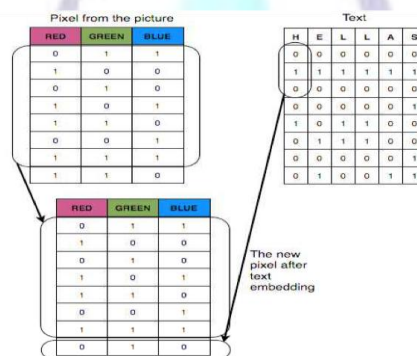


Fig 1: The LSB embedding method

The Fibonacci method

Several research teams have developed and extended the LSB method using different approaches. One of these approaches is presented in [9], and uses the Fibonacci numbers, which are defined by the linear recurrence relation

$$F_n = F_{n-1} + F_{n-2}, n > 1, \text{ with } F_0 = 0 \text{ and } F_1 = 1.$$

In the LSB scheme, one bit is embedded in each pixel color of the image. To increase the amount of data, we could embed them in higher bitplanes. This however causes noticeable distortions in the image. To avoid this problem, the Fibonacci method uses a new representation of the pixel value which increases the number of available bitplanes. According to Zeckendorf's Theorem [8], every positive integer can be uniquely represented as a sum of distinct, nonconsecutive Fibonacci numbers. More specifically, Zeckendorf's Theorem states that for every $x \in \mathbb{N}^*$ there exists a finite sequence c_1, c_2, \dots, c_n of positive integers with $c_{i+1} > c_i + 1, 1 \leq i < k$, such that $x = \sum_{i=1}^k F_{c_i}$.

This sum is called the Zeckendorf representation of x . Equivalently, given that $F_k \leq F_{k+1}$, for some $k \geq 2$, we have that $x = \sum_{i=1}^k w_i F_{i+1}$, where $w_i \in \{0, 1\}$, $w_k = 1$ and there is no i such that $w_i = w_{i+1} = 1$. The sequence $w_n w_{n-1} \dots w_1$ is a binary word with no consecutive 1's and it is called the Fibonacci encoding of x (with respect to n). By convention, 0 is encoded by the binary word $00 \dots 0$, so that the encoding is extended to all nonnegative integers. Strictly speaking, this encoding is a bijection from the set of integers $\{0, 1, 2, \dots, F_{n+2} - 1\}$ onto the set of binary words of length n , having nonconsecutive 1's. For this reason, in the literature, these words are often referred to as Fibonacci words. Consequently, each value of each pixel color, can be written from now on, using this new basis. Since this value ranges from 0 to 255, we only need the terms F_2, F_3, \dots, F_{13} for its encoding, that is the elements of the set

$$F_{(12)} = \{1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233\}$$



consisting of all F_n , where $2 \leq n \leq 13$.

For example, the number 39 can be written as a sum of elements of the set $F_{(12)}$ as

$$39 = 0 \cdot 1 + 1 \cdot 2 + 1 \cdot 3 + 0 \cdot 5 + 0 \cdot 8 + 0 \cdot 13 + 0 \cdot 21 + 1 \cdot 34 + 0 \cdot 55 + 0 \cdot 89 + 0 \cdot 144 + 0 \cdot 233,$$

or

$$39 = 0 \cdot 1 + 0 \cdot 2 + 0 \cdot 3 + 1 \cdot 5 + 0 \cdot 8 + 0 \cdot 13 + 0 \cdot 21 + 1 \cdot 34 + 0 \cdot 55 + 0 \cdot 89 + 0 \cdot 144 + 0 \cdot 233,$$

or

$$39 = 0 \cdot 1 + 0 \cdot 2 + 0 \cdot 3 + 1 \cdot 5 + 0 \cdot 8 + 1 \cdot 13 + 1 \cdot 21 + 0 \cdot 34 + 0 \cdot 55 + 0 \cdot 89 + 0 \cdot 144 + 0 \cdot 233.$$

These three sums can be represented respectively by the binary words:

$$000010000110F_{12}, 000010001000F_{12}, 000001101000F_{12}.$$

Therefore, the number 39 has more than one representations, in terms of Fibonacci numbers. From all these representations, we choose the one which is derived from Zeckendorf's Theorem, i.e., the one corresponding to the binary word $000010001000F_{12}$, (which is the only one containing no consecutive 1's). In this way, we produce 12 bitplanes for embedding data and so we can increase the stego capacity.

An extension of Zeckendorf's Theorem

In this section, we give an extension of Zeckendorf's theorem, which allows us to use other integer sequences for the representation of bytes. This way, we improve the previously described Fibonacci method. Some of these results were presented in [10].

Theorem 1 (Extension of Zeckendorf's Theorem). Let $(a_n)_{n \in \mathbb{N}^*}$ be a strictly increasing sequence of positive integers, with $a_1 = 1$, $a_2 = 2$, and $a_n + a_{n+1} \geq a_{n+2}$ and $n \in \mathbb{N}^*$. Then, every positive integer x with $a_n \leq x < a_{n+1}$, $n \in \mathbb{N}^*$ can be represented as a sum of different and nonconsecutive terms of the sequence (a_n) , with the restriction that the term a_n appears in the sum.

Proof. We will use induction on x . If $1 \leq x \leq 3$, then the claim obviously holds, since $a_1 = 1$, $a_2 = 2$, $a_3 = 3$. Let $x > 3$, and assume that the claim holds for all positive integers less than x . Then, there exists a unique $n > 2$, such that $a_n \leq x < a_{n+1}$. If $a_n = x$, then the claim obviously holds. If $a_n < x < a_{n+1}$, then, setting $y = x - a_n$, by the induction hypothesis, we have that y is represented as a sum of different and nonconsecutive terms of the (a_n) sequence. Let $y = a_{b_\lambda} + a_{b_{\lambda-1}} + \dots + a_{b_1}$ be a representation of y , where (b_n) is a strictly increasing sequence of nonconsecutive positive integers, and $\lambda \in \mathbb{N}^*$. Therefore, given the restriction that the term a_n must appear in the representation of x , we obtain the representation $x = a_n + y$ of x . Furthermore, we have that

$$x < a_{n+1} \Rightarrow y < a_{n+1} - a_n$$

and since $a_{n+1} - a_n \leq a_{n-1}$, it follows that $y < a_{n-1}$. The representation of y gives that $a_{b_\lambda} \leq y$, thus $a_{b_\lambda} < a_{n-1}$. Finally, since the sequence (a_n) is strictly increasing, we have $a_{b_\lambda} < a_{n-1} \Rightarrow b_\lambda < n - 1$, so that the representation of x contains no consecutive terms.

According to Theorem 1, given a sequence (a_n) satisfying the above requirements, any $x \in \mathbb{N}$ is represented as

$$x = \sum_{i=1}^n w_i \cdot a_i$$

where $w_i \in \{0,1\}$, $w_n = 1$ and there is no i , such that $w_i = w_{i+1} = 1$. The number n is the unique positive integer satisfying $a_n < x < a_{n+1}$. Therefore, each representation corresponds to a unique Fibonacci word $w_n w_{n-1} \dots w_1$, so that each $x \in \mathbb{N}$ corresponds to at least one Fibonacci word. By choosing the lexicographically greatest corresponding word, we define an encoding for the elements of \mathbb{N} . This is equivalent to applying recursively the restriction of the Theorem. The implementation for this is trivial and, therefore, the process of encoding and decoding each integer x is straightforward.

For example, the sequence $(1, 2, 3, 5, 7, 9, 11)$ is a sequence of length 7 which encodes all integers in the interval $[0, 22]$. (Note that 22 is obtained as the maximum sum of nonconsecutive terms of the sequence, i.e., $22 = 11 + 7 + 3 + 1$.) Following the restrictions of Theorem 1, the number 18 is represented as

$$18 = 11 + 7 \text{ or } 18 = 11 + 5 + 2.$$

These representations correspond to the Fibonacci words $w = 1010000$ and $u = 1001010$ respectively. Since w is greater than u , the number 18 is encoded by w .

Corollary 2. Let $(a_n)_{n \in \mathbb{N}^*}$ be a strictly increasing sequence of positive integers, such that the Fibonacci numbers F_2, F_3, F_4, \dots form a subsequence of (a_n) . Then, every positive integer x , with $a_n < x < a_{n+1}$, $n \in \mathbb{N}^*$ can be represented as a sum of different and nonconsecutive terms of the sequence (a_n) , with the restriction that the term a_n appears in the sum.



Proof. By the definition of (a_n) , it follows that $a_1 = F_2 = 1, a_2 = F_3 = 2$ and $a_3 = F_4 = 3$. Hence, by Theorem 1, it suffices to prove that $a_n + a_{n+1} \geq a_{n+2}$. This is obviously true for $n = 1$. If $n \geq 2$, then by the definition of (a_n) , it follows that there exists a unique $k \in \mathbb{N}^*$, where $2 \leq k \leq n$, such that

$$F_k \leq a_n < a_{n+1} \leq F_{k+1}$$

If, $a_{n+1} \leq F_{k+1}$ then $a_{n+2} \leq F_{k+1}$ so that

$$F_k \leq a_n < a_{n+1} < F_{k+1} \Rightarrow a_n + a_{n+1} > F_{k-1} + F_k = F_{k+1} \geq a_{n+2}$$

On the other hand, if $a_{n+1} = F_{k+1}$, then $a_{n+2} \leq F_{k+2}$, so that

$$F_k \leq a_n < a_{n+1} = F_{k+1} \Rightarrow a_n + a_{n+1} > F_k + F_{k+1} = F_{k+2} \geq a_{n+2}$$

Thus, in both cases we have that $a_n + a_{n+1} > a_{n+2}$

The Lucas numbers

The mathematician Francois Edouard Anatole Lucas (1842 - 1891), studied the Fibonacci numbers and the related sequence that is named after him. The Lucas sequence is defined as follows:

$$L_n = \begin{cases} 2, & n = 0, \\ 1, & n = 1, \\ L_{n-1} + L_{n-2}, & n > 1. \end{cases}$$

As in the Fibonacci method, we define the set

$$L_{(12)} = \{1, 2, 3, 4, 7, 11, 18, 29, 47, 76, 123, 199\} \text{ that consists of all } L_n, \text{ where } 0 \leq n \leq 11.$$

In accordance to Theorem 1, each positive integer in the closed interval $[0, 255]$ can be uniquely represented as a sum of different, nonconsecutive Lucas numbers. For example, the number 39 can be written as a sum of elements of the set $L_{(12)}$ as

$$39 = 0 \cdot 1 + 0 \cdot 2 + 1 \cdot 3 + 0 \cdot 4 + 1 \cdot 7 + 0 \cdot 11 + 0 \cdot 18 + 1 \cdot 29 + 0 \cdot 47 + 0 \cdot 76 + 0 \cdot 123 + 0 \cdot 199,$$

$$39 = 1 \cdot 1 + 1 \cdot 2 + 0 \cdot 3 + 0 \cdot 4 + 1 \cdot 7 + 0 \cdot 11 + 0 \cdot 18 + 1 \cdot 29 + 0 \cdot 47 + 0 \cdot 76 + 0 \cdot 123 + 0 \cdot 199.$$

The above two sums are encoded respectively by the binary words: $000010010100_{L_{12}}$, $000010010011_{L_{12}}$.

As in the case of Fibonacci numbers, we use the representation $000010010100_{L_{12}}$ which is the lexicographically greatest.

The method using Catalan numbers.

The Catalan numbers are named after the Belgian mathematician Eugene Charles Catalan (1814 - 1894). The Catalan numbers appear in a variety of counting problems. The n -th Catalan number is given explicitly in terms of binomial coefficients by

$$C_n = \frac{1}{n+1} \binom{2n}{n} = \frac{(2n)!}{(n+1)n!}, n \in \mathbb{N}.$$

The first 11 Catalan numbers are 1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796. Clearly, not every integer in the interval $[0, 255]$ can be represented as a sum of distinct Catalan numbers. For example, $18 = 14 + 2 + 2$.

We define the set $C_{(6)} = \{1, 2, 5, 14, 42, 132\}$, consisting of the Catalan numbers which are less than or equal to 255. Furthermore, we consider the sets

$$CF = C_{(6)} F_{(12)} = \{1, 2, 3, 5, 8, 13, 14, 21, 34, 42, 55, 89, 132, 144, 233\}$$

and

$$CL = C_{(6)} L_{(12)} = \{1, 2, 3, 4, 5, 7, 11, 14, 18, 29, 42, 47, 76, 123, 132, 199\}.$$

We use the sets CF and CL for the representation of the color values of each pixel. Using the set CF for the representation of the color values of each pixel, we create 15 virtual bitplanes, 3 more than the number of bitplanes produced by the Fibonacci method. Using the set CL , we create 16 virtual bitplanes, 4 more than the Fibonacci and the Lucas representation. In this way, more stego data can be embedded into the image.

The procedure for this method is as follows. Firstly, each pixel value is represented by its decimal value. Then, this value is converted using CF (or CL) numbers, to 15 (or 16) bitplanes. So, the message can be embedded in the last bitplane (as in LSB) as well as in higher bitplanes.

Measures and results

We examine the effectiveness of our methods, by comparing the quality of the corresponding (resulting) images. The implementation of our method is done using our own application (Crypto ver. 1.2), which is powered by MATLAB software. As test image, we use the three grayscale images: baboon, airplane and pepper. (Figure 2).



Fig 2: Test Images

We create four stego images, each one using a different method (LSB, Fibonacci, Lucas, Catalan-Fibonacci, Catalan-Lucas method).

We use two metrics to compare the various image techniques: The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR). A lower value for MSE means lower error, and as seen from the inverse relation between MSE and PSNR, this translates to a high value of PSNR. MSE is the cumulative squared error between the stego image and the original image and is defined as follows:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2,$$

where I and K are two $m \times n$ monochrome images, where one of the images is considered to be a noisy approximation of the other.

PSNR is given by the formula:

$$PSNR = 10 \log_{10} \left[\frac{MAX_I^2}{MSE} \right],$$

where MAX_I is the maximum possible pixel value of the image. When the pixels are represented using 8bits per sample the value of MAX_I is 255.

We present the measures of PSNR, when embedding data in more bitplanes, in tables 1, 2, 3, 4, 5, 6 and 7. For pages other than the first page, start at the top of the page, and continue in single-column format.

In the next tables and in columns 2, 3, 4, 5 and 6, we see PSNR measurements for each method. We can see that our method improves the performance of the image quality, when compared to the simple LSB and the Fibonacci methods. Moreover, the Lucas sequence and the sets CF and CL give better results than the LSB and Fibonacci methods, when we embed data in higher bitplanes. Of all the methods, the one using the set CL seems to give better results. More specifically, the PSNR value in the Lucas method and in the Catalan-Lucas method is increased by about 2.5% and 13% respectively, compared to the Fibonacci method, while the Catalan-Lucas method improves the PSNR value of the Catalan-Fibonacci method by about 5%. In figure 5, we can see an average PSNR comparison and in figure 6, we see what happens to the picture when we overflow the image with stego data

Table 1. Measures for image using 1 bitplane (Last).

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	76.29	76.29	76.29	76.29	76.29
324	73.37	73.37	73.37	73.37	73.37
576	71.19	71.19	71.19	71.19	71.19
784	69.44	69.44	69.44	69.44	69.44
1024	67.99	67.99	67.99	67.99	67.99



Table 2. Measures for image using 2 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	72.21	72.21	72.21	72.21	72.21
324	69.16	69.16	69.16	69.16	69.16
576	67.06	67.06	67.06	67.06	67.06
784	67.06	67.06	67.06	67.06	67.06
1024	67.06	67.06	67.06	67.06	67.06

Table 3. Measures for image using 3 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	64.99	69.16	69.16	69.16	69.16
324	61.88	66.07	66.07	66.07	66.07
576	59.83	64.01	64.01	64.01	64.01
784	58.19	62.35	62.35	62.35	62.35
1024	56.80	60.96	60.96	60.96	60.96

Table 4. Measures for image using 4 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	61.01	64.99	66.85	64.99	66.85
324	57.88	61.88	63.74	61.88	63.74
576	55.84	59.83	61.68	59.83	61.68
784	54.20	58.19	60.04	58.19	60.04
1024	52.82	56.80	58.65	56.80	58.65

Table 5. Measures for image using 5 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	54.01	61.01	52.82	61.01	64.99
324	50.88	57.88	59.02	57.88	61.88
576	48.84	55.84	56.98	55.84	59.83
784	47.21	54.20	55.34	54.20	58.19
1024	45.83	52.82	53.96	52.82	56.80

Table 6. Measures for image using 6 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	48.50	56.86	58.27	57.46	62.56
324	45.36	53.76	55.14	54.76	59.67
576	43.33	51.68	53.10	52.84	56.98
784	41.70	50.07	51.47	51.03	55.48
1024	40.32	48.65	50.09	49.14	53.97

Table 7. Measures for image using 7 bitplanes.

stego bits	PSNR (LSB)	PSNR (Fibonacci)	PSNR (Lucas)	PSNR (Catalan-Fibonacci)	PSNR (Catalan-Lucas)
196	38.86	52.68	54.17	56.19	59.27
324	36.75	49.54	50.88	53.08	56.14
576	34.93	49.54	48.84	51.02	54.10
784	33.62	45.87	47.21	49.39	52.47
1024	31.26	44.50	45.83	48.01	51.09

In figure 3, we can see an average PSNR comparison and in figure 4, we see what happens to the picture when we overflow the image with stego data

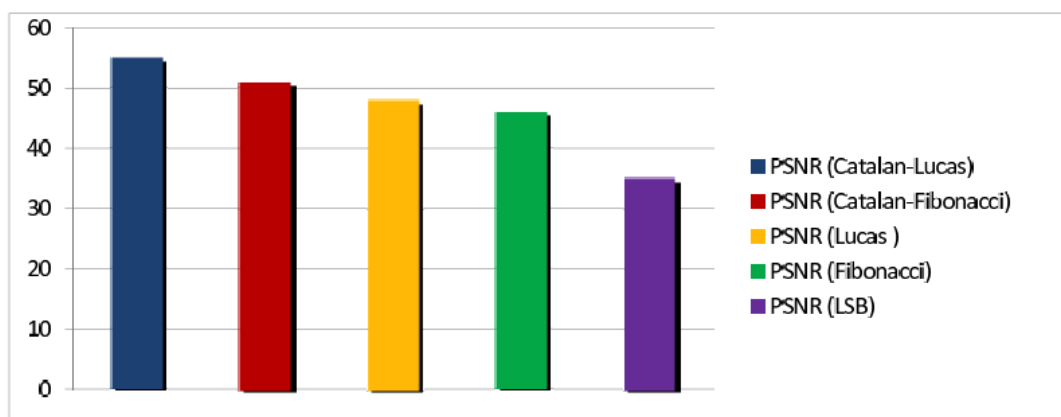


Fig 3: Average PSNR comparison

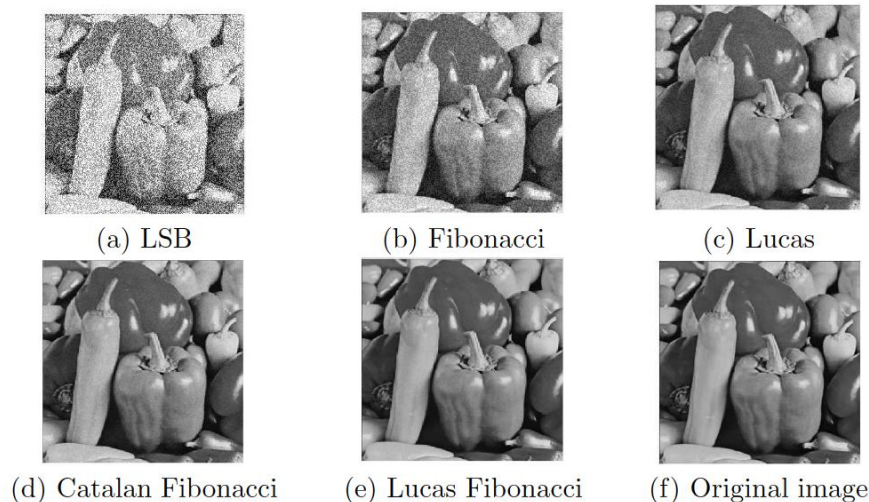


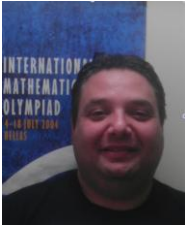
Fig 4:Worst cases comparison

Conclusions

Our methods using Lucas, Catalan-Fibonacci and Catalan-Lucas numbers are superior over the Fibonacci data hiding technique. In the classical LSB technique it is only possible to embed secret data just in the first few bitplanes, since image quality becomes radically deteriorated when embedding data in higher bitplanes. Battisti et al. (2006), Sandipan et al (2007) proposed an improvement over this by using Fibonacci embedding technique and our method, using a greater set of virtual bitplanes, increases the number of stegobits that can be embedded when an image should be regarded as a stego image. Besides, apart from the LSB method, in which the tracking of data is easy (if there is suspicion of course), the Fibonacci, Lucas, CF and CL methods offer a kind of encryption, through the way that they conceal the bits in more bitplanes. Particularly, our method enables a large number of bitplanes, offering not only more space for our stego data but also increased security towards steganalysis software.

REFERENCES

- [1] Luo Xiangyang, Liu Bin and Liu Fenlin, Improved RS method for detection of LSB Steganography. Int. Workshop on Inf. Security and Hiding, Singapore (2005).
- [2] Cheddad A., Condell J., Curran K. and McKeivitt P., Digital image steganography: Survey and analysis of current methods, Signal Processing, 90, pp. 727–752 (2010)
- [3] Jagpal G., Steganography in digital images, Thesis, Cambridge University Computer Library (1995).
- [4] Tanako S., Tanaka K. and Sugimura T., Data hiding via steganographic image transformation, IEICE Trans. Fundamentals, E83-A, pp. 311–319 (2000).
- [5] Shao-Hui C., Tian-Hang Y. and Wen Hong-Xun G., A variable depth LSB data hiding technique in images, Conference on Machine Learning and Cybernetics, 7, pp. 3990–3994 (2004).
- [6] Chandramouli R., and Memon N., Analysis of LSB based image steganography techniques, Proc. of the IEEE International Conference on Image Processing, pp.1019-1022 (2001).
- [7] Sandipan D., Ajith A. and Sugata S., An LSB Data Hiding Technique Using Prime Numbers, The Third International Symposium on Information Assurance and Security, Manchester, UK, IEEE CS press (2007).
- [8] G.M. Phillips, Zeckendorf representation, in Hazewinkel, Michiel, Encyclopaedia of Mathematics, Springer, ISBN 978-1556080104, Picione (2001).
- [9] De Luca, Battisti F., Carli M., Astola J. and Egiazarian K., A Fibonacci LSB data hiding technique, 14th European signal processing conference, EUSIPCO 2006 (2006).
- [10] Aroukatos N., Manes K., Zimeras S. and Georgiacodis F., Data Hiding Techniques in Steganography using Fibonacci and Catalan numbers, 9th International Conference on Information Technology – New Generations, Las Vegas, Nevada, USA (2012).



Mr. Nikolaos Aroukatos holds a BSc. on Mathematics from University of Athens, a MSc of Mathematics from G.O. University and a MSc of Computer Science from University of Piraeus .He is a PhD candidate in Department of Informatics, University of Piraeus. His research areas include Steganography, Steganalysis, Cryptography , Image Processing and Number Theory.



Mr. Kostas Manes is a PhD candidate in Department of Informatics, University of Piraeus. His research areas include Combinatorial Enumeration, Combinatorics in Lattice paths and Asymptotic Enumeration.

Dr. Stelios Zimeras holds a BSc. (1st class Hons) on Statistics and Insurance Sciences from University of Piraeus and Ph.D. on Statistics from the University of Leeds, U.K. Since 2008, he is a full time staff member (Assistant Prof) on statistics and probabilities at the Department of Mathematics Direction: Statistics and Financial-Actuarial Mathematics, University of the Aegean, Samos, Greece. His published material and presentations are on a number of topics of Statistical Simulations, Image analysis, Medical Image analysis, Spatial Statistics, Telemedicine, and Statistical Modeling with applications in biology, ecology, and medicine, statistical epidemiology, and Bayesian statistics.

Professor F.Georgiakodis has taught undergraduate and postgraduate courses at the University of Bradford , University of Piraeus, University of Aegean ,University of Patras and the University of Athens. His research interests include .Biostatistics, Biometry, Survival Analysis ,Reliability ,Data Analysis , Data Mining and Image Processing. He has published a number of papers in international and Greek scientific Journals and is the author of books in Statistics.