Wu-Lee Steganographic Algorithm on Binary Images Processed in Parallel

Silvana GRECA[1], Edlira MARTIRI[2]

[1] Faculty of Natural Sciences, Computer Science, University of Tirana, ALBANIA
[2] Faculty of Economy, Applied Informatics, University of Tirana, ALBANIA
silvana.greca@fshn.edu.al, edlira.martiri@unitir.edu.al

***Abstract*— Data security is nowadays one of the most active fields of study in Informatics and Computer Science. Author right for intellectual property is a real challenge, especially when information is processed and transmitted. One of the electronic forms of digital data is images. They are widely used in organizations, research institutions, and in environments where high resolution is needed. Here we come across to another kind of processing: parallelism.**

Information hiding and watermarking techniques are essential in addressing the problem of author copyright. This is done by means of steganographic algorithms. The aim of this paper is to analyze and present those steganographic algorithms which can be parallelized to a coarse-grain size. We will consider one typical steganographic algorithms for binary images, Wu-Lee and will analyze the parallelization of this algorithm.

***Index Terms*— Steganography, Wu-Lee method, binary image, parallel processing**

I. Presentation and Definition of Steganography in Images

NOWADAYS the growth of information stored in digital forms and the development of new multimedia services, security-related issues are becoming more and more important. The acceptance of the new services that may be offered depends on whether they are associated by safe techniques to protect the interests of several parties: at least to the service provider and its user. Moreover, the nature of the data holder (image, text, audio or video) is threatened for some reasons related to his digital forms:

a. make a copy of them is quite easy, unfortunately, we can say it is perfect (the copy does not change at all from the original)

b. their mode of transmission is also distress: if only a pirated copy is made, it can be accessed by anyone who needs it.

c. the plasticity of digit holders threatens their contents. A malicious user can transform an image so putting at risk the techniques for protecting their intellectual property.

For many reasons, it is critical that the protection system of copyright to be conceived in such a way as to minimize the above risks. For this, many authors will not be encouraged to distribute their works, the health organizations will minimize the use of image scanning, and the video and music industry would not have that distribution which they have, without using the steganographic techniques and watermarking.

The examples that we mentioned above are typical examples of images which are used and provided in a very high resolution or that have a high rendering execution time. At the same time, with the increasing strength of computer graphics systems, even Steganography (hiding digital data on digital medium), has increased rapidly. Before describing what is Steganography it is better to understand the image architecture. According to D. Sellar, "for a computer, the image is a table of numbers that represent the light intensities at different pixels". These pixels represent the image data. The main typical files are those of 8-bits and 24-bits per pixel. Both these formats have their advantages and disadvantages when are used in steganographic techniques. 8-bit images are better because of their size, but have a caveat because there are used only 256 possible colors. Such images are those with extension .GIF. 24-bit images offer much more flexibility. The high number of colors (over 16 million) makes the discovery of a hidden message to be very difficult to catch after his encryption. Images can also be compressed by two techniques: lossy and lossless [1]. Compression with lossy (as the .JPG word indicate), reduces the size of an image by removing redundant data of the image [1]. Compression lossless (.GIF and .BMP) indicated the save of the image as it is and for this reason this type of compression is the selected techniques in Steganography.

Steganography is the science of hiding data [2]. In the case of images, the most used techniques are those that modify the least significant bit (LSB). Exploiting LSB of each byte (8 bits) in the image can be stored 3 bits of data in each pixel for 24 bits images [3]. Depending on the color table which is used (i. e. Grey Palette) we can gain two LSB from one byte and the changes cannot be cached by the human eye. However, there are other known methods for hiding messages [4,5,6,7]. The only problem with LSB technique is its vulnerability of being attacked such as for example the changes of the image and the type converting (e.g. from .GIF to .JPF). This vulnerability would be eliminated by using watermarking, by adding specific algorithms in data filtering and masking. Those new techniques which are used in complex steganographic algorithms in data covering and in encryption of images are more secure. These techniques will be discussed in the next section of this paper.

II. Steganographic Algorithms and Their Implementation in Binary Images

Regarding to steganographic algorithms implemented in the images, it is worthy to mention that many of them have starting from the replace of LSB and then have been developed in other complex and secure methods. As we

mentioned in the second section, it is needed a well-defined area of study, in ensuring as much as more efficient parallelization, so for that here we are providing a summary of those techniques by describing algorithms which stands for each of them.

Steganographic techniques are divided in three groups:

1. Replacing technique: by using those techniques we replace the redundant or unneeded bits of a medium with the bits from the message which will transmit. The main example is the LSB method.

2. Transformation techniques

2. a. Discrete Cosine Transformation: this technique tends to divide the image in pieces according to the quality of image.

2. b. Discrete Fourier Transformation: this technique transforms one image from spatial domain to frequency domain.

3. Spread Spectrum Encryption: this method hides a small message in a large [8].

If we mean to apply the LSB method in binary images, or even other techniques, the embedded data are most likely to be intercepted by the human eye. For this reason there exist some special techniques. We will mention one of them: Wu-Lee.

The Wu-Lee algorithm is designed to embed the bits of the message in a m x n block by changing the value of at most one pixel in each block. The chosen pixel will be near borders and by doing so, the possibility that it will be intercepted is lower than in LSB [9].

III. The Evolution of Images in Parallel Programming

Parallel image processing, incorporated with steganographic algorithms shows us one interlinked discipline. Both of them are important directions in computer science research in the academic world. This cooperation derives as effect of two main developments: advancement and improvement of hardware devices from one side, and as a need for protection of intellectual authentication of digital information, which is become more and more important in data processing and transmit.

According to Eijkhout (2010):

"Parallel processing is found in the crossroad of many disciplines and study fields. In order to be more successful in parallel processing in science, one must have good knowledge in all these fields, since calculations are fruit of the context of their application".

Meanwhile, Pacheco in "*An introduction to parallel programming*" is answered three more important questions [10]:

1. Why we need more and more higher performance?

2. Why we are headed and building parallel systems?

3. Why we need implemented parallel programs and in the same time do not to have one method to convert serial (executable in a machine with one processor) codes into parallel code?

The development of processors has made us to profit from their speed but for other physical constraints, the graphical performance in conventional computers is in decline. Moore's Law is not any more valid for the new situation created by hardware development, due to the fact of the integration of processing units in the same chip. It is also true that do not exist programs that could convert the serial codes in parallel codes, for the reason of physical and software complexity of the issues. That is why software developers are required to create parallel programs and also to have a picture of the corresponding problem.

IV. Review of the Existing Wu-Lee Method

Hiding in a binary image is very difficult due to the fact that changing one bit in a binary image is easy to detect as it changes the color from black to white or the opposite. However, there are many techniques for hiding a message in a binary image [11,12,13]. Methods such as: Zhao-Koch, Wu-Lee, CPT and TP are presented in sequential mode. Actually does not exits any work about the parallelization of these algorithms. Let us discuss the pseudo-code of one the main steganographic method which is use for binary images: Wu-Lee.

Before dealing with the algorithm, we will explain:

K: secret key used to embed the message. It has a size of m *x* n, and contains 0 and 1.

B: the block with size m *x* n.

F: cover image used to hide data.

Algorithm

Input: cover image (F), secret key (K), secret message d
Output: stego-image
Begin
divide F into blocks B, each of size m *x* n.
If 0 < SUM(B AND K) < SUM(K) then
if SUM(B AND K) mod 2 = d then keep B intact.
else if SUM(B AND K) = 1 then
randomly pick a bit B_{ij} = 0 such that K_{ij} = 1 and change B_{ij} to 1.
else if SUM(B AND K) = SUM(K) - 1 then
randomly pick a bit B_{ij} = 1 such that K_{ij} = 1 and change B_{ij} to 0.
else
randomly pick a bit B_{ij} such that K_{ij}= 1 and complement B_{ij}.
EndIf;
Else
select next block.
EndIf;
End [9]

This method uses a key K for added security and logical operations. The image is partitioned into blocks B of *m x n* pixels each, and the key, which is also an *m x n* block of bits, is used to embed at most one data bit *d* in each block. It is assumed that the image size is an integer multiple of the block size. The main advantage of the method is that the data bit is embedded in the block by changing the value of at most one pixel.

V. PARALLELIZATION AND ANALYZES OF THE WU-LEE ALGORITHM

Nowadays with the new development in hardware and software it is possible to process the images in parallel to achieve high performance. The above algorithm is showing a high scale of parallelization. Parallel system will implement those very well because of matrix structure of the images. We will show a detailed version of this algorithm.

We have in total 5 lines of sequential instructions and 3 lines of instructions which can be processed in parallel. In total we have 8 lines of processing instructions. Let us see the algorithm and then try to analyze it.

```
Begin
        Read image(m,n)
        Read message(k)
        Read mb, nb                     //block sizes
        binary_message_size=k*8
        binary_message(k*8)=Binary_Conversion(message(k))
        For i=1 to n, step i+nb
          For j=1 to m, step j+mb
            If j+mb>m CONTINUE
            Else
                    If SUM(B AND K) mod 2 = d CONTINUE
                    Else if SUM (B AND K) = 1 then
                            index_i=random(B,0)
                            index_j=random(B,0)
                            image(index_i, index_j)= 1
                    Else if SUM (B AND K) = SUM (K)-1
                            index_i=random(B,1)
                            index_j=random(B,1)
                            image(index_i, index_j)= 0
                    Else
                            index_i=random(B,NULL)
                            index_j=random(B,NULL)
                            image(index_i,index_j)=
                            =Complement(image(index_i,index_j)
                    EndIf
                  EndIf
          EndFor
        EndFor
End
```

The figures 1 and 2 show a binary image and its blocks.

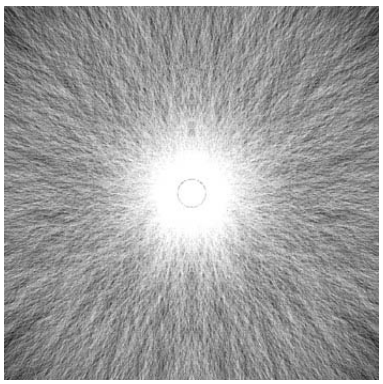

Fig. 1. Example of a binary image

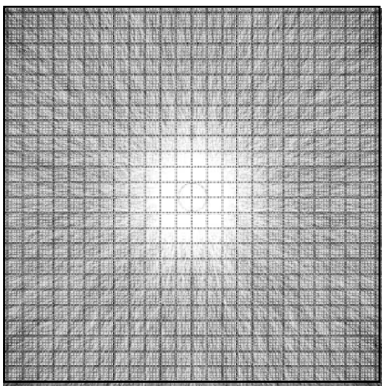

Fig. 2. Blocks in a binary image

The minimized number of processing instructions comes from the simplicity of the data structure used here. We considered the image as a two-dimensional array of data, which content would be rather 1 or 0.

In order to show the advantages of parallel computations of the Wu-Lee Method, we will show in this section some of the main factors which influence the performance of the abovementioned algorithm in a parallel system.

a. *Concurrency*

Definition: an algorithm is considered concurrent in the case where a large number of instructions can be parallelized.

b. *Grain Size*

Definition: it is a measure of the parallelism size. It refers to the number of instructions which can be executed in parallel

c. *Speed-Up*

Definition: Is the ratio between the sequential computation time and the parallel computation time.

d. *Efficiency*

Definition: Efficiency of an algorithm which is executed on p-processors is defined as the ration between Speed-Up and the number of processors.

e. *Effictivity*

Definition: it is one of the most important factors. An algorithm is considered effective if it maximizes the Sp*Ep product. [14,15,16]

Let us suppose an image of size 300 x 200. In total we will have 60,000 pixels. In order to be a multiple of these sizes, we choose the size block 2 x 3. So:

Total no. of pixels: 60,000
Block size in pixels: 3*3 = 6
Total no. of blocks: 60,000/6=10,000
Total no. of symbols: 10,000/8bit = 1250
Supposed message: 100 symbols

We can improve the algorithm by adding a stepping counter. Since we can hide in total 1250 symbols, but our message is only 100 symbols, this means that we can calculate only 1/10 of the total blocks. By doing so, we have:

No. of iterations of inner For-loop: 30
No. of iterations of outer For-loop: 20

No. of instructions in inner For-loop: 3
Total no. of instructions: 30*20*3=1,800

Total no. of sequential instructions: 1804

The table below shows the values of Tp, Sp, Ep and Sp * Ep according to the given numbers of Processing Units (PU).

PU	Tp	Sp	Ep	Sp* Ep
1	1804	1	1	1
5	364	4.956	0.9912	4.912
10	184	9.804	0.9804	9.612
100	22	82	0.82	67.24
300	10	180.4	0.6103	110.098
600	7	257.7	0.4295	110.68
1,000	5.8	311.03	0.311	96.74
1,800	5	360.8	0.20	72.16

According to the function Sp * Ep, there is a maximization at number of PUs equal to 600. This means that for this situation, 300 is the optimal number of PCs, shown in figure 3.

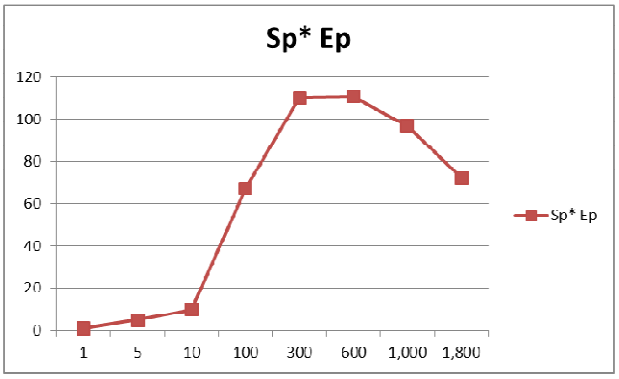

FIG. 3. GRAPH OF SP*EP AND ITS MAXIMAL VALUE

VI. CONCLUSIONS AND FURTHER WORK

We presented in this paper an analysis of steganographic algorithms applied in binary images, especially in the case when these images are processed in parallel. The main algorithms in this case are Wu-Lee and Zhao-Koch, but our focus was on the former one. Parallel execution dramatically reduces response time for data-intensive operations.

We showed that this algorithm has a coarse-grain size of parallelization. We analyzed the different factors related to parallelism such as speed-up, effectively, etc and showed the optimal number of processors of the system where the Wu-Lee algorithm could run in parallel.

Today we need a higher level of performance so that we can multiply the number of computations that an application can handle. In today's computing environment, there's really only one way to get there: utilize a parallel architecture to run multiple tasks at the same time. Especially in image processing we think that parallel systems must be adopted as soon as possible in developing countries in this era of technology where our region needs the fastest and complete solutions.

We will develop more these ideas and implement their parallelization. For this accomplishment we will use GPUs, moreover, NVIDIA and its framework based on CUDA.

REFERENCES

[1] Neil F. Johnson, Stefan C. Katzenbeisser, "A survey of steganographic techniques" - Chapter 3.
[2] Bret Dunbar, "A Detailed look at Steganographic Techniques and their use in an Open-Systems Environment", SANS Institute, 2002.
[3] Mehdi Kharrazi, Husrev T. Sencar, and Nasir Memon, "Image Steganography: Concepts and Practice", WSPC, 2004
[4] K. Stefan and A. Fabien "Information hiding techniques for steganography and digital watermarking," Artech House, 2000
[5] F. Petitcolas, R. Anderson and M. Kuhn "Information Hiding", A Survey Proceedings of the IEEE, special issue on protection of multimedia content, July 1999.
[6] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," IBM Systems Journal, Vol. 35, No. 3 and 4, pp. 313-336, 1996.
[7] M. Celik, G. Sharma, and A. M. Tekalp, "Reversible data hiding" Proceeding IEEE ICIP, Rochester, NY, Sept. 2002.
[8] H. Motameni, M. Norouzi, M. Jahandar, and A. Hatami, "Labeling Method in Steganography", World Academy of Science, Engineering and Technology, 2007
[9] D. Salomon, "Coding for data and computer communications", Springer, 2005
[10] P. Pacheco, "An introduction to parallel programming", Elsevier, 2011
[11] Y. Chen, H. Pan, and Y. Tseng, "A secure Data Hiding Scheme for Two-Color Images," IEEE Symposium On Computers and Communications, 2000
[12] M. Wu, E. Tang, B. Liu: "Data Hiding in Digital Binary Image", IEEE International Conference on Multimedia & Expo (ICME'00), New York City, 2000.
[13] M. Wu and J. Lee, "A Novel Data Embedding Method for Two-Color Facsimile Images", Proeeding of International Symposium on Multimedia Information Processing, Taiwan, December 1998.
[14] Fatmir Hoxha, "Elemente te njehsimit paralel", Textbook, SHBLU, Tirana, 2004.
[15] Blaise Barney, Lawrence Livermore, "Introduction to Parallel Computing", National Laboratory.
[16] Nicholas J. Hopper John Langford Luis von Ahn, "Provably Secure Steganography", 2006.
[17] Edlira Martiri, Gloria Tuxhari, Albana Gorishti, "Parallel Computing: still missing in the Albanian reality", International Conference, Economic & Social Challenges and Problems, Faculty of Economy, University of Tirana, December 2010.
[18] Fatmir Hoxha, Andrea Kotro, "Matematike e Zbatuar", Textbook, SHBLU, Tirana, 2002.