Jan Długosz University in Częstochowa
Scientific Issues, Mathematics XVI, Częstochowa 2011

A PICTURE THAT IS SOMETHING MORE THAN JUST A PICTURE

Robert Dyja, Artur Jakubski

Institute of Computer and Information Sciences
Częstochowa University of Technology
ul. Dąbrowskiego 73, 42-201 Częstochowa, Poland
e-mail: robert.dyja@icis.pcz.pl
e-mail: artur.jakubski@icis.pcz.pl

Abstract. Our paper presents the ways of hiding information with the usage of a digital picture. A branch of science that deals with hiding messages in the wider media is called steganography. Due to the rapid expansion of Internet and the associated increase in data exchange, this field appears to be a subject of interest. In this paper we present our own algorithms of hiding data in pictures and their implementations.

1. Introduction

1.1. Introduction to steganography

This article concerns steganography – the field related to cryptography. While cryptography hides the content of the information through its encryption, steganography tries to hide the fact of its occurrence.

There is the following division of steganographic systems: Pure Steganography, Private Key Steganography, and Public Key Steganography.

Kerckhoffs principle says that the cryptosystem should be secure even if all the details of its operation (besides the key) are known. Modification of the least significant bit is a classic representative of methods of replacing (substitution). The last bit (e.g. pixel component values) is replaced by a bit (or bits) from the message.

1.2. Steganography in pictures

Steganography for pictures, as well as for other media, uses a carrier (in this case a picture) to forward a confidential message. Just as in other cases, in steganography it is so important to prepare the carrier (media) that it does not arouse any suspicion of outsiders.

In addition to normal hiding, digital images allow for the usage of specific properties of the graphic format in which the image is saved, in order to provide confidential information. One of the techniques that allow us to hide information in a picture is to modify the least significant bit (Least Significant Bit) [4, 5]. As its name indicates, the technique involves the modified least significant bits of a numerical value describing the intensity of the color at the selected location of a picture.

Generally, with the LSB method in an image, we can hide any information stored in the form of consecutive bits. It is not important, whether it is a text message, the encrypted text message or other image, or any other type of binary file. The only limitation is the capacity of the media. In the example picture of 800x600 pixels in size, assuming the modification of only one bit of blue and one bit of green, we can send a secret message on a maximum of 120 000 bytes.

2. Proposed algorithm for steganography in pictures

2.1. Algorithm

In this chapter we present the algorithm for generating an image with the hidden information in it (a text). There are two approaches to this problem: the first, when a key and a text are hidden together in a picture, and the second, when a key is outside the picture. In the last case, the key can be delivered to the addressee through an encrypted connection, hidden in another image or delivered through another way. The key to our algorithm consists of a pair of numbers. The first number is a generator of multiplicative group Z_p and the second is a prime number p. The group generator is found by using the following theorem.

Theorem 1. *Let p–1 have a decomposition into prime factors p–1= $p_1p_2...p_k$, then $g \in Z_p$ is a generator of multiplicative group Z_p if and only if $g^{p-1/p_i} \neq 1$ for each $1 \leq i \leq k$.*

Theorem 2. *The number of generators of the group Z_p is $\phi(\phi(p))$, where ϕ is the Euler function.*

In the algorithm 1 an input is an array of size of k consisting of primes that are factors of $p - 1$. The algorithm returns the generator of a multiplicative group Z_p.

```
Algorithm 1: Algorithm of creating the generator

  Ingresso: $d[k]$ - array of prime divisors $p-1$
  Uscita: $g$ - generator

1 Choose randomly $g$ from the range $< 2, p-2 >$;
2 per $i \leftarrow 0$ a $k-1$ fai
      se $(g^{p-1/d[i]} = 1)$ allora
          go to step 1;
3 return $g$
```

Suppose that we have generated a prime number p greater than the number of bits of a bitmap and a generator of the group Z_p. Now we show an example of an algorithm to hide a text in the image. If the number of pixels in our bitmap is n, we can put $3n$ bits of text in this picture. This means that we put at most $3n/8$ bytes of text in the bitmap. In our algorithm, we will put 2 pixels in each pixel of the bitmap. The realization will take two approaches here: the first, when the information will be placed in two fixed RGB colors (green, blue), and the second, when the information will be placed in any two of the three colors (in the sequence: red color with green, red with blue and green with blue). Let us assume then that we have k bytes of text ($k < n/4$), which are included in the bitmap.

```
Algorithm 2: Algorithm of hiding text in an image

  Ingresso: $t[k]$ - array of text, $b[n]$ - array of values of bitmap pixels
  Uscita: $b[n]$ - array of values of bitmap pixels

1 per $i \leftarrow 0$ a $k-1$ fai
2     t[i] divide into four blocks of two bits;
3     compute the successive four values:
      $g^{4i} \bmod p, g^{4i+1} \bmod p, g^{4i+2} \bmod p, g^{4i+3} \bmod p$;
4     put the next block of the text in places of the $b[n]$ array determined
      by instruction 3 (from instruction 2).
5 return $b[n]$;

  Note: when some of values calculated in point 3 is bigger than $n-1$ of
  the index of the last bitmap pixel, we calculate the next value.
```

Generating the places on a bitmap, in which the bits of hidden text will be placed, is achieved by a special case of linear pseudo-random number generator. Generators of this type have been proposed by American mathematician Derrick Lehmer [1]. In the generator, pseudo-linear consecutive numbers are calculated based on a recursive pattern: $X_{n+1} = aX_n \bmod N$ ($X_0 < N$ and $a < N$).

Theorem 3. *If $N = p$ is prime, the linear generator has a maximum period equal to p if and only if a is a primitive root of p.*

In our case the places, where the bits of hidden text are put, are analogous to that of the generator for $a = g$ and $N = p$.

2.2. Implementation

In order to present the method and effectiveness of the proposed solutions, we presented an example implementation of one of the variants of the proposed algorithm. A hidden message is the Latin text 'Lorem ipsum ...' This text was taken from publicly available sources that the reader can find at `http://lipsum.com`.

Carrier of secret information is a photograph of 768x1024 pixels and saved as 24-bit bitmap in BMP format. Comparison of the appearance of the image before encoding and after hiding the information is presented in Figure 1.

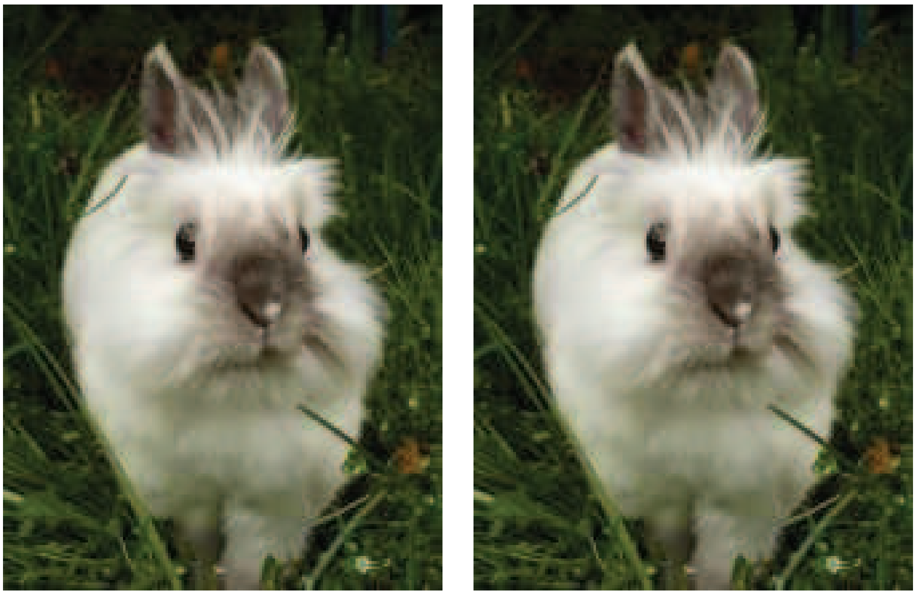

Figure 1: The original carrier (without modification) is shown on the left, while the image-bearer of hidden information counting 1000 words is presented on the right.

As we can see, comparing ‘with the naked eye’ the two pictures has a small chance of detecting any suspicious elements in any of them. The hidden message in this photo consists of 1,000 words, which use 6717 characters (bytes). Modification of the least significant bit of color is virtually undetectable to the human eye. It should be noted that in the real situation the original image should be removed in order to hinder the direct comparison. In a situation, where an attacker has the original image, he can make a simple ‘byte by byte’ comparison of both images, which immediately reveals that the picture shown in Figure 1 on the right has undergone some editing.

3. Steganoanalysis – χ^2 test

Frequently used attack is a test based on chi-square test. It assumes that in an ordinary image the distribution of least significant bits of the value 1 is not the random normal distribution. Introduction of the message may change the distribution of ones in such a way that it is close to a normal distribution.

Preparation for this test begins from counting the number of occurrences of all possible values of certain color (this is to create a histogram). Then, the values are grouped in pairs of different values in the last bit (each pair has a value of even and odd). In a typical photograph, each pair of the numbers of odd and even values should deviate from equilibrium. In the case of pictures with a hidden message, this number will move towards equality (if the image is analyzed with noise only). The most important advantage is that the attacker does not need to have an original picture.

Chi-square test is based on verifying the hypothesis that the last bits of the image have normal distribution. If there is no such distribution, the picture probably does not contain any hidden information. If not, the picture may contain a hidden message.

Implementation of chi-squared test involves drawing up statistics:

$$\chi^2 = \sum_i \frac{(o_i - e_i)^2}{e_i},$$

where o is the observed number of ones on the last bit, while e is an expected number estimated on the basis of the normal distribution [2], [3], [4].

Checking the picture with the message in Figure 1 (right) by using the test does not qualify the picture as a suspected one, the same the original picture in Figure 1 (left). However, filling in this picture the last bits by the random value (noise) results in qualifying such images as suspected.

Details of the chi-squared test auxiliary values are given in Table 1.

	Blue color	Green color
Value of χ^2	312.548	365.336
Number of degrees of freedom	123	124

Table 1: Values of chi-squared test for accepting the hypothesis counted for a picture with a hidden message.

4. Summary

In this paper, our carrier was a 24-bit bitmap. It should be noted that high-quality scanners offer 30- or 36-bit sample depth. With 30-bit RGB color is derived after 10 bits per component, and a 36-bit, each component is a 12-bit. In these cases, 3 bits (for 10-bit component) can be placed in each component, and in the second case – 5 bits of concealed information.

Another interesting issue is the fact that different areas of the picture are more or less susceptible to steganoanalysis. In areas where every pixel has a color different from its neighbors, the changes caused by placing hidden bits in them are difficult to detect.

References

[1] D.H. Lehmer. Mathematical methods in large-scale computing units. *Proc. 2nd Symposium on Large-Scale Digital Calculating Machinery*, (Cambridge, MA, 1949), *Ann. Comp. Lab. Harvard University*, **26**, 141–146, 1951.

[2] S. Katzenbeisser, F.A.P. Petitcolas. *Information Hiding Techniques for Steganography and Digital Watermarking.* Artech House, Boston 2000.

[3] S.M. Kot, J. Jakubowski, A. Sokołowski. *Statystyka.* Wydawnictwo Difin, Warszawa 2007.

[4] A. Cheddad, J. Condell, K. Curran, P. Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal Processing*, **90**, 727–752, 2010.

[5] N. Provos, P. Honeyman. Detecting steganographic content on the Internet. *Proceedings of the Network and Distributed System Security Symposium*, NDSS 2002, San Diego, California, USA 2002.