High Complexity Bit-Plane Security Enchancement in BPCS Steganography

Andysah Putera Utama Siahaan
Faculty of Computer Science
Universitas Pembangunan Panca Budi
Jl. Jend. Gatot Subroto Km. 4,5 Sei Sikambing,
20122, Medan, Sumatera Utara, Indonesia

ABSTRACT
In BPCS Steganography, data hiding will be split into blocks that have a high complexity where the blocks are categorized into informative and noise-like regions. A noise-like region is a bit-plane that has the greatest probability as a data hiding since it has a high complexity. In this region, the data inserted is vulnerable to attack. Someone can easily take a series of characters that are stored on a noise-like region previously if the system is not modified. Improving the bit-plane composition is to increase data security. Bit-plane will be combined with a specified key. The key should be changed to bit-plane form as well. The key that has already been turned into the bit-plane will be mated with the original data. Using an exclusive-or of this part is the best way to produce the cipher bit-plane. Finally, the data residing on the cover image produced have a high-security level.

General Terms
Steganography, Cryptography, Security

Keywords
Bit-Plane, High Complexity, Segmentation

1. INTRODUCTION
Steganography is The technique to hide secret information in some other carrier without any apparent evidence of data exchange [5]. Steganography is also the art of data hiding. It covers messages in a vessel image [1][3]. Only the receiver knows what happens inside the picture. In steganography, sometimes the user do not think about the security level. When concealing the data, it is all the original data. It means those are hidden without encryption. Bit-Plane Complexity Segmentation, BPCS, is one of the steganography techniques that grouping the set of the original message to bit-planes [5]. A noise-like region consist complex pattern [11][12]. It happens since the bit-plane has the intensity variation. BPCS has an 8x8 matrix to represent the information. It serves to divide the plaintext to layers. There are totally eight layers to represent those bits in the plaintext. Complexity is the formula to measure whether the bit-plane is ready to be used. By attacking the bit-plane, someone might be retrieving the secret message. Modifying the structure of the noise-like region is the security technique to increase the security system [2]. There are numerous ways to manipulate the bit-plane. It can be combined with encryption to generate the new bit order. Creating the cipher block has numerous benefits to camouflage the original message. Once the cipher bits are created, the attacker must spare more time to resolve the message.

2. CRYPTOGRAPHY SCHEME
Steganography is the embedding messages art such that only the specific participants know the existence of the information [8] while cryptography is the way to change the character with the help of particular algorithms for turning the original data into an unintelligible form [7][16]. Steganography is hiding the message in a cover image so that it becomes invisible [9]. The extraction will involve the both method in retrieving the hidden information [10].

Table 1. Comparison of Steganography and Cryptography

Criteria	Steganography	Cryptography
Carrier	All digital media	Plaintext/ image/audio
Hiding Information	Yes	No
Additional Carrier	Required	Not required
Hidden Message	Imperceptible	Detection of message is possible
Key	Optional	Required
Visibility	Never visible	Always visible

Table 1 shows the comparison in steganography and cryptography. The criteria describe the parameter compared to both security techniques.

3. RELATED WORK
Eiji Kawaguchi and R. O. Eason discovered the new technique to hide the information. It is called Bit-Plane Complexity Segmentation (BPCS). The information hiding capacity is around 50% of the container size [4]. The image consists of three layers. Each layer component value ranges from zero 0 to 255, where zero represents the darkest color and 255 represent the brightest [13]. The image of the document is divided into several segments with a size of 8x8 pixels of each segment. In the 8-bit image of the document, each segment has eight bit-planes representing the pixels of each of these bits. The process of division of segment 8x8 pixels into eight bits is called the bit slicing plane [15]. The process of inserting a message is carried on the segments that have high complexity. This is called noise-like region. In these segments, the insertion is not only performed on the least significant bit, but on the whole noise-like bit planes.

In steganography, a message is hidden inside a container such as a picture, movie or audio [14]. Some of them use BPCS algorithm to cover up the message. But it is a problem when

the image does not fit to maintain. The data is vulnerable. There is no encryption method inserted in its algorithm.

4. PROPOSED WORK

The strength of BPCS depends on the sharpness of the eye can see behind the image. A subtle difference is that the color file has a slightly different file structure [6]. Usually, humankind is not able to extract the strange dot or secret information in noise-like regions. It is a complicated color intensity pattern. But, to improve the security, the noise-like pattern needs to be modified. Every message in bit-planes mates to the encryption key bit-plane. The modification is not affecting the image quality. Meanwhile, BPCS is robustness against third party attacks.

This research aims to combine the steganography and cryptography. One Time Pad is a symmetric key cryptography technique that has the same formula for encryption and decryption. The key used must be replicated as long as the bit-plane. The key can be generated from the random or regular key by typing them on a keyboard. If a random key stream is used, the ciphertext will be random as well. By combining One Time Pad and BPCS will make the concealment perfect. The attacker will be deceived. He gets a set of bits obtained at the time of interception, but unfortunately, the hard work required to break the message. It makes the combination of two methods work together. The following equation shows the One Time Pad formula.

$$CT = PT\ xor\ Key \qquad (1)$$

Where:

CT : Cipher Bit-Plane
PT : Plaint Bit-Plane
Key : Password Block

The method is delineated in several steps as follows.

- Convert the carrier image from PBC to CGC
- Make the bit-plane
- Bit-plane analysis
- Size-estimation
- Processing message
- Encrypting message
- Implant secret blocks into carrier image.
- Save file

5. TESTING AND EVALUATION

The original pixels are transformed into numbers from 0 to 255 which represent the color intensity. Those are turned into pure Binary Code (PBC) system. Each row and column represent the pixel position in the image. Table 2 is an example of image segmentation. Every segment consists of 64 pixels. It consists of three layers, red, green and blue.

Table 2. An 8x8 pixel of Image Light Intensity

PBC Pixel Segment 1								
	0	1	2	3	4	5	6	7
0	127	27	29	39	49	65	67	69
1	31	52	12	1	7	0	1	10
2	23	29	24	28	37	44	41	21
3	9	14	20	35	32	44	34	1
4	54	44	63	47	59	85	60	74
5	117	91	121	169	186	185	203	190
6	170	181	193	209	208	213	216	235
7	200	198	179	184	198	187	199	247

Every single pixel has to be converted to binary. It will split the order of the bits into the similar bit-plane. The PBC form must be turned into CGC to remap the bit planes to perform the message insertion. It stems from the fact that CGC is better than PBC in producing a better-looking stego image. Table 3 and Table 4 show the comparison of PBC and GCC. The insertion in PBC is irregularities.

Table 3. Pure Binary Code Segment

PBC (Pure Binary Code)								
	0	1	2	3	4	5	6	7
0	01111111	00011011	00011101	00100111	00110001	01000001	01000011	01000101
1	00011111	00110100	00001100	00000001	00000111	00000000	00000001	00001010
2	00010111	00011101	00011000	00011100	00100101	00101100	00101001	00010101
3	00001001	00001110	00010100	00100011	00100000	00101100	00100010	00000001
4	00110110	00101100	00111111	00101111	00111011	01010101	00111100	01001010
5	01110101	01011011	01111001	10101001	10111010	10111001	11001011	10111110
6	10101010	10110101	11000001	11010001	11010000	11010101	11011000	11101011
7	11001000	11000110	10110011	10111000	11000110	10111011	11000111	11110111

Let's take an example from row 0 and column 0. The binary digit is 01111111. The are eight digits. As seen in the formula, the first numbers are identical while the next come from the operation of the preceding digits.

PBC : $0_1 1_2 1_3 1_4 1_5 1_6 1_7 1_8$

CGC_1 : PBC_1
0

CGC_2 : $PBC_2 \oplus PBC_1$
$1 \oplus 0$
1

CGC_3 : $PBC_3 \oplus PBC_2$
$1 \oplus 1$
0

CGC_4 : $PBC_4 \oplus PBC_3$
$1 \oplus 1$
0

CGC_5 : $PBC_5 \oplus PBC_4$
$1 \oplus 1$
0

CGC_6 : $PBC_6 \oplus PBC_5$
$1 \oplus 1$
0

CGC_7 : $PBC_7 \oplus PBC_6$
$1 \oplus 1$
0

CGC_8 : $PBC_8 \oplus PBC_7$
$1 \oplus 1$
0

The CGC is 01000000. This calculation continues until 64 times for each segment. Table 4 is the result of the first segment conversion and it is repeated until all the segments are finished.

Table 4. Canonical Gray Code Segment

CGC (Canonical Gray Code)								
	0	1	2	3	4	5	6	7
0	01000000	00010110	00010011	00110100	00101001	01100001	01100010	01100111
1	00010000	00101110	00001010	00000001	00000100	00000000	00000001	00001111

2	00011100	00010011	00010100	00010010	00110111	00111010	00111101	00011111
3	00001101	00001001	00011110	00110010	00110000	00111010	00110011	00000001
4	00101101	00111010	00100000	00111000	00100110	01111111	00100010	01101111
5	01001111	01110110	01000101	11111101	11100111	11100101	10101110	11100001
6	11111111	11101111	10100001	10111001	10111000	10111111	10110100	10011110
7	10101100	10100101	11101010	11100100	10100101	11100110	10100100	10001100

Every bit order in Canonical Gray Code is split into bit-planes. For example, on the first bit-plane, it is resulted from drawing every first bit on CGC map while on the last one, it is from the last bit. Embedding a file-block in an n-th least PBC plane means modifying the colors of several pixels in that block with the value 2^{n-1}. In this case the "blocking effect" likely to appear on the stego image. While in the CGC embedding, the color change "differs pixel by pixel" in the block varying from 1 to 2^{n-1}. The average change in the block is 2^{n-1}. Table 5 is the result of splitting bit-planes.

Table 5. Bit-Planes Slicing

Bit Plane 1					9	0,080357143		
	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	1	1	1	1	1
6	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1

Bit Plane 2					33	0,294642857		
	0	1	2	3	4	5	6	7
0	1	0	0	0	0	1	1	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	1	0	1
5	1	1	1	1	1	1	0	1
6	1	1	0	0	0	0	0	0
7	0	0	1	1	0	1	0	0

Bit Plane 3					34	0,303571429		
	0	1	2	3	4	5	6	7
0	0	0	0	1	1	1	1	1
1	0	1	0	0	0	0	0	0
2	0	0	0	0	1	1	1	0
3	0	0	0	1	1	1	1	0
4	1	1	1	1	1	1	1	1
5	0	1	0	1	1	1	1	1
6	1	1	1	1	1	1	1	0
7	1	1	1	1	1	1	1	0

Bit Plane 4					48	0,428571429		
	0	1	2	3	4	5	6	7
0	0	1	1	1	0	0	0	0
1	1	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1
3	0	0	1	1	1	1	1	0
4	0	1	0	1	0	1	0	0
5	0	1	0	1	0	0	0	0
6	1	0	0	1	1	1	1	1
7	0	0	0	0	0	0	0	0

Bit Plane 5					61	0,544642857		
	0	1	2	3	4	5	6	7
0	0	0	0	0	1	0	0	0
1	0	1	1	0	0	0	0	1
2	1	0	0	0	0	1	1	1
3	1	1	1	0	0	1	0	0
4	1	1	0	1	0	1	0	1
5	1	0	0	1	0	0	1	0
6	1	1	0	1	1	1	0	1
7	1	0	1	0	0	0	0	1

Bit Plane 6					52	0,464285714		
	0	1	2	3	4	5	6	7
0	0	1	0	1	0	0	0	1
1	0	1	0	0	1	0	0	1
2	1	0	1	0	1	0	1	1
3	1	0	1	0	0	0	0	0
4	1	0	0	0	1	1	0	1
5	1	1	1	1	1	1	1	0
6	1	1	0	0	0	1	1	1
7	1	1	0	1	1	1	1	1

Bit Plane 7					57	0,508928571		
	0	1	2	3	4	5	6	7
0	0	1	1	0	0	0	1	1
1	0	1	1	0	0	0	0	1
2	0	1	0	1	1	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	1	1	1	1
5	1	1	0	0	1	0	1	0
6	1	1	0	0	0	1	0	1
7	0	0	1	0	0	1	0	0

Bit Plane 8					56	0,5		
	0	1	2	3	4	5	6	7
0	0	0	1	0	1	1	0	1
1	0	0	0	1	0	0	1	1
2	0	1	0	0	1	0	1	1
3	1	1	0	0	0	0	1	1
4	1	0	0	0	0	1	0	1
5	1	0	1	1	1	1	0	1
6	1	1	1	1	0	1	0	0
7	0	1	0	0	1	0	0	0

Every bit-plane has the complexity value. The threshold is the limit to determine whether the bit-plane can be inserted by the message. The complexity is measured by how many times the bit change from 0 to 1 and from 1 to 0. The maximal bit change value of 8x8 block is 112. The formula to calculate the complexity is given in equation 2.

$$\alpha = \frac{k}{n} \qquad (2)$$

Where:

α : Bit-Plane Complexity

k : Total Bit change

n : Maximum Change of 8x8 bit-plane

Table 6 shows the complexity of all bit-planes. The next procedure is to set the limitation between informative and noise-like region. The limit is called threshold. Generally, the standard value of the threshold is 0.3, but it can be modified as needed. The modification is often performed to avoid the informative area or to add the noise-like area. The n value is set to the maximum that is 112.

Table 6. The Complexity of Bit-Planes

Bit-Plane	Bit Change	Complexity
1	9	0,080357142857
2	33	0,294642857143
3	34	0,303571428571
4	48	0,428571428571
5	61	0,544642857143
6	52	0,464285714286
7	57	0,508928571429
8	56	0,500000000000

Assume the threshold limit is 0.3. In Table 6, there are two numbers are below the threshold. They are bit-plane number 1 and 2. The numbers which are in that range are called the informative regions. The message cannot insert these regions. Putting the message block in these ranges will make the stego image easily distinguished. The process of embedding is skipped to the next bit-plane.

In this case, the word to embed is "ANDYSAH!". The word consists of eight characters. It has one message block. Table 6 and Table 8 show the plain message and the bit-planes.

Table 7. The Original Message

Char	Dec.	Biner
A	65	1000001
N	78	1001110
D	68	1000100
Y	89	1011001
S	83	1010011
A	65	1000001
H	72	1001000
!	33	100001

Table 8. The Bit-Plane of Message

Bit Plane Message								
	0	1	2	3	4	5	6	7
0	0	1	0	0	0	0	0	1
1	0	1	0	0	1	1	1	0
2	0	1	0	0	0	1	0	0
3	0	1	0	1	1	0	0	1
4	0	1	0	1	0	0	1	1
5	0	1	0	0	0	0	0	1
6	0	1	0	0	1	0	0	0
7	0	0	1	0	0	0	0	1

The bit-plane must be transformed into encrypted block before it is completely embedded in the noise-like regions. This step is the main part of the study. The encryption method takes apart in this step. The One Time Pad is the very common algorithm to be applied in BPCS. The OTP needs key as the password to produce the cipher bit-planes. For example, the key is "SDM21". The word is repeated until the length of bit-plane is covered.

Table 9. Repeated Key

Char	Dec.	Biner
S	83	01010011
D	68	01000100
M	77	01001101
2	50	00110010
1	49	00110001
S	83	01010011
D	68	01000100
M	77	01001101

The key itself is converted to bit-plane model as shown in Table 10. The key can replace the conjugation block. BPCS uses conjugation block to converted informative region to noise-like region. It is needed when the informative region is found. In conventional BPCS, not all the bit-plane needs to be conjugated. But in combining with the encryption, mating the bit-plane with the conjugation block is a must.

Table 10. Bit-Plane of the Key

Bit Plane Message								
	0	1	2	3	4	5	6	7
0	0	1	0	1	0	0	1	1
1	0	1	0	0	0	1	0	0
2	0	1	0	0	1	1	0	1
3	0	0	1	1	0	0	1	0
4	0	0	1	1	0	0	0	1
5	0	1	0	1	0	0	1	1
6	0	1	0	0	0	1	0	0
7	0	1	0	0	1	1	0	1

The bit-plane of message and key must be transformed by performing the exclusive-or ($\oplus$). The result of the bit-planes is the cipher bit-planes. After all the bit-planes are encrypted, the bit-planes are returned to its original position and reconverted to pure binary code before they are finally rewritten to the new stego image.

Table 11. Bit-Plane of Cipher

Cipher Bit-Plane								
	0	1	2	3	4	5	6	7
0	0	0	0	1	0	0	1	0
1	0	0	0	0	1	0	1	0
2	0	0	0	0	1	0	0	1
3	0	1	1	0	1	0	1	1
4	0	1	1	0	0	0	1	0
5	0	0	0	1	0	0	1	0
6	0	0	0	0	1	1	0	0
7	0	1	1	0	1	1	0	0

Table 11 shows the bit-plane after encryption and Table 12 shows the bit-plane after conversion. The previous process turns the message into the encrypted message.

Table 12. Cipher Message

Char	Dec.	Biner
	18	00010010
	10	00001010
	9	00001001
k	107	01101011
b	98	01100010
	18	00010010
	12	00001100
l	108	01101100

As discussed earlier, the bit-plane 1 and 2 in Table 5 are the informative regions. The embedding starts from the third bit-plane which the complexity value is 0,303571428571. This value is greater than the threshold, so it is safe to hide. After the calculation is finished, the third bit-plane is now replaced by the message block. This process is to make the intruder takes more time to break the message. Once the intruders solve the bit-plane, they will get the key to reconstructing the bit-planes into the plain text back.

6. ANALYSIS

The different before and after hiding can be calculated by Peak Signal to Noise Ratio. It shows how close the vessel image compared to the original image. The closer the distance, the better the result. The following formula shows how it works.

$$PSNR = 20\log\left(\frac{255}{\sqrt{\frac{1}{n}*\sum_{i=0}^{n-1}(D-S)^2}}\right) \quad (3)$$

7. CONCLUSION

BPCS Steganography uses the noise-like region to store the information which can be associated with encryption. The noise-like pattern can be changed without changing the original information. One Time Pad is the best way of applying the security technique in this method since it does not use the complicated calculation. It is only to reconstruct the message block with the exclusive-or operation by providing the key as the password. The strength key ensures the security level increase.

8. FUTURE SCOPE

The threshold needs to limit the hiding into the bit-planes. It is still static. The future research is expected that the vessel image is smoother than the earlier. The hiding can be placed based on a specific threshold value. The value itself is adjustable. The threshold is automatically determined by image analysis module. They will be different in the different case. The program will find the highest possibility complexity. Hence, it needs an appropriate algorithm to distinguish the threshold value. Afterward, the container will be undetected by human eyes. The vessel image after embedding is a lossy steganography. The method can be further developed to return the original bits when extraction.

9. REFERENCES

[1] N. Johnson dan S. Jajodia, "Exploring Steganography: Seeing The Unseen," IEEE Computer, pp. 26-34, 1998.

[2] Y. K. S. P. S. M. M. B. Prashant Lahane, "Visual Cryptography and BPCS Steganography for Data Shielding," International Journal Of Engineering And Computer Science, vol. 4, no. 5, pp. 11997-11999, 2015.

[3] P. R. Rudramath dan M. R. Madki, "High Capacity Data Embedding Technique Using Improved BPCS Steganography," International Journal of Scientific and Research Publications, vol. 2, no. 7, pp. 1-4, 2012.

[4] E. Kawaguchi dan R. O. Eason, "Principle and applications of BPCS-Steganography," SPIE.

[5] S. Mehta, K. Dighe, M. Jagtap dan A. Ekre, "Web Based BPCS Steganography," International Journal of Computer Technology and Electronics Engineering, vol. 2, no. 2, pp. 126-130, 2015.

[6] P. P. Khairnar dan P. V. S. Ubale, "Steganography Using BPCS technology," International Journal of Engineering and Science, vol. 3, no. 2, pp. 8-16, 2013.

[7] C. Jain, VivekParate, A. Dhamanikar dan R. Badgujar, "Review on Steganography and BPCS Technology in Steganography for Increasing Data Embedding Capacity," International Journal of Innovative Research in Computer and Communication Engineering, vol. 3, no. 1, pp. 60-65, 2015.

[8] N. F. Johnson dan S. Jajodia, "Steganography: Seeing the Unseen," IEEE Computer, pp. 26-34, 1998.

[9] B. A. Forouzan, Cryptography & Network Security, New Delhi: McGraw Hill Publication, 2008.

[10] A.-M. A., Steganography-Based Secret And Reliable Communications Improving Steganographic Capacity And Imperceptibility, School of Information Systems, Computing and Mathematics, 2010.

[11] V. J. Patel dan N. RipalSoni, "Uncompressed Image Steganography using BPCS: Survey and Analysis," OSR Journal of Computer Engineering, vol. 15, no. 4, pp. 57-64, 2013.

[12] H. Noda, M. Niimi dan EijiKawguchi, "A Steganography Based on Region Segmentation by Using Complexity Measure," Trans. of IEICE, vol. 81, no. 2, pp. 1132-1140, 1998.

[13] P. R. Rudramath dan M. R. Madki, "Improved BPCS Steganography Based Novel Approach for Data Embedding," International Journal of Engineering and Innovative Technology, vol. 1, no. 3, pp. 156-159, 2012.

[14] V. J. Deshmukh dan D. A. S. Alvi, "BPCS Steganography and Visual Cryptography: An Advance Technique for Online Payment Security in E-Commerce for Developing Countries," International Journal of Science and Research, vol. 4, no. 3, pp. 294-297, 2013.

[15] S. S. Solanke dan P. D. C. Dhanwani, "Survey on Modified BPCS Steganography Based on Sequence of Cipher Bits," International Journal of Engineering Development and Research, vol. 3, no. 2, pp. 1036-1040, 2015.

[16] A.P. U. Siahaan, "RC4 Technique in Visual Cryptography RGB Image Encryption," SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE), vol. 3, no. 7, pp. 1-6, 2016.

10. AUTHOR PROFILE

Andysah Putera Utama Siahaan was born in Medan, Indonesia, in 1980. He received the S.Kom. degree in computer science from Universitas Pembangunan Panca Budi, Medan, Indonesia, in 2010, and the M.Kom. in computer science as well from the University of Sumatera Utara, Medan, Indonesia, in 2012. In 2010, he joined the Department of Engineering, Universitas Pembangunan Panca Budi, as a Lecturer, and in 2012 became a junior researcher. He is applying for his Ph. D. degree in 2016. He has written in several international journals He is now active in writing papers and joining conferences.

.