International Journal of Security and Its Applications
Vol. 11, No. 9 (2017), pp.1-12
http://dx.doi.org/10.14257/ijsia.2017.11.9.01

Vernam Conjugated Manipulation of Bit-plane Complexity Segmentation

Andysah Putera Utama Siahaan

Faculty of Computer Science, Universitas Pembangunan Panca Budi, Medan, Indonesia
Ph.D. Student of School of Computer and Communication Engineering, Universiti Malaysia Perlis, Kangar, Malaysia
andiesiahaan@gmail.com

Abstract

Steganography has no protection to make the information hidden is safe from theft. This technique only tries to store information in the image purely. There are many ways to steal information from pixels stored in RGB colors. Bit-Plane Complexity Segmentation (BPCS) is one of the steganography technique is often performed to conceal data. But in BPCS method, a used pattern is not a classified anymore. It breaks down and changes the plain text structure into square information. BPCS has two types of area, informative and noise-like region. This division depends on the threshold value. The noise-like region is the only area that can store confidential information. It is an 8 x 8 matrix pattern. The plain text is turned into bits and finally kept in the matrix. Converting the bit-plane information is a technique for increasing the security of the vessel image. Vernam cipher can easily be occupied to modify the bit-plane structure with the predetermined blocks conjugation. The cipher block contains a new set of unbreakable characters. It increases the security level.

Keywords*: *BPCS, Steganography, Security, Cryptography, Vernam, Bit-plane

1. Introduction

A digital image is the result of the function of light intensity with the formula f (x, y) in the two-dimensional plane. X and Y are spatial coordinates. The function value at each coordinate point represents the brilliance level of the image at that point. This image can then be used as a place where information is inserted. Cryptography and Steganography are the science of data security that is still one clump of science. Steganography is occupied to hide information in the picture by changing the bit pattern while image encryption tries to rebuild the original image by replacing the bits by the encrypted information [7][11]. It makes difficult to understand. Steganography is a technique how to conceal the communication in a cover file [4]. It is always used to send a file without being detected, thereby reducing suspicion of transmitted data. It is covered by a vessel image [1]. Bit-plane Complexity Segmentation (BPCS) is a technique that is used to store information on a medium such as a picture, video, audio, etc. The stego-medium is called a vessel. This technique uses imagery division into segments divided into several bit-planes. However, the data can be possibility solved or attack so that data can be retrieved by the irresponsible. The pattern used in the BPCS is to store data by grouping bit corresponds to the index with the n x n patterns forming where each sequence of eight characters will be transformed into binary message blocks.

BPCS uses 8 x 8 matrix to implement the bit-plane formation. However, the weaknesses can be seen from the preparation of the information in the 8 x 8 pattern as

Received (January 18, 2017), Review Result (August 16, 2017), Accepted (August 24, 2017)

ISSN: 1738-9976 IJSIA

well. If complexity used is determined, it is possible that the information stored in the vessel image can be easily retrieved. There is no image complexity standard measure [14]. Typically the threshold revolves around the value of 0.3, and this value has become a commonly used method of this BPCS [2][5]. The binary value inserted into the vessel should be further enhanced image security to anticipate unwanted things. If the threshold value and the data are retrieved from the vessel image, the Vernam Cipher algorithm will take control in protecting the arrangement of bits of data on the bit-plane BPCS [3]. Data is retrieved, but the bit-plane is still fully protected.

Vernam Cipher has played a major role in cryptography because it is a perfect secrecy system [10]. It converts to the data bits before it is inserted into the vessel image. The original bits will be XOR by the key matrix as generated earlier. The information stored in the vessel image has been encrypted beforehand. This method is very lightweight and faster to use than other methods that have to use many math calculations.

2. Theories

2.1 Bit-plane Complexity Segmentation (BPCS)

Eiji Kawaguchi and R. O. Eason introduce the BPCS technique to be used in uncompressed color-images documents. He used an 8 x 8 matrix to cover the bit-planes data. Each matrix is called a segment. The original images consist of several segments representing the bit data. Slicing is the segment division into bits. Representation of bit-plane is Pure Binary Code (PBC) system. In BPCS, the insertion process is performed on a bit-plane system with CGC (Canonical Gray Code) because the process of slicing bits of CGC is better than in PBC. A bit plane with PBC representation is converted into bit-plane with CGC representation.Improved BPCS steganography takes different threshold values of image complexity to deal with bit-plane accordingly [15].

The complexity of each subsection of a bit-plane is defined as the number of non-edge transitions from 1 to 0 and 0 to 1 [8]. BPCS is another substitution type method, but rather than replacing specific bits. It scans for complex areas of an image and replaces those with the converted plain text [12]. The process of inserting a message is carried on the segments that have high complexity. It is divided into “informative region” and “noise-like region” [13]. In these segments, the insertion is not only performed on the least significant bit, but on the whole noise-like bit-planes. Therefore, the BPCS technique, data capacity can optimally reach 50% of vessel image size [6].

Figure 1 describes the converting process from the original image to bit-plane slicing. The picture at the top-left is the original image which has converted to an 8-bits gray color. Each segment is divided into the 8 x 8 pixel, where each pixel is converted to the 8-bit binary system. Every index on the bits is concatenated into a new single piece bit-plane.

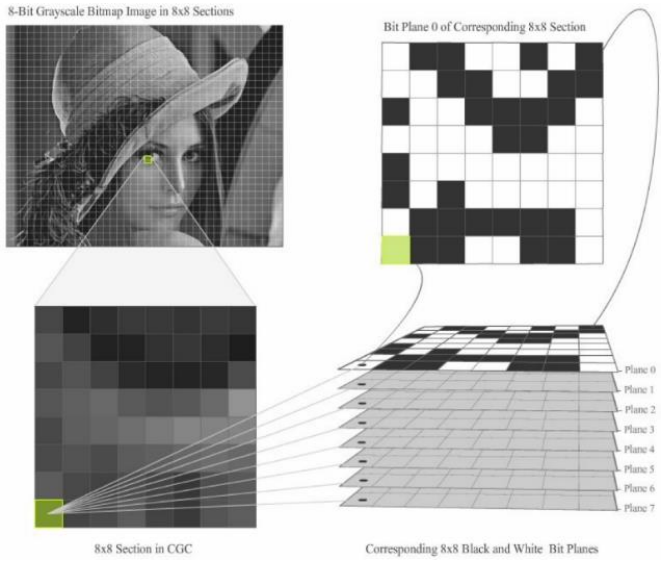


Figure 1. Bit-plane Construction

Conjugation is a technique to maintain bit-plane on BPCS to keep them in the noise-like region. Suppose a black and white image sized 8 x 8 pixel P has a white background color and foreground color black. W is a pattern with all pixels white. Suppose a black and white image sized 8 x 8 pixel P has a white background color and foreground color black. W is a pattern with all pixels white, and B is the pattern with all black pixels. Wc and Bc are a chessboard pattern, with a pixel in the top left of the white on black on the Wc and Bc. P * is the conjugate of the image P shown in Figure 2.

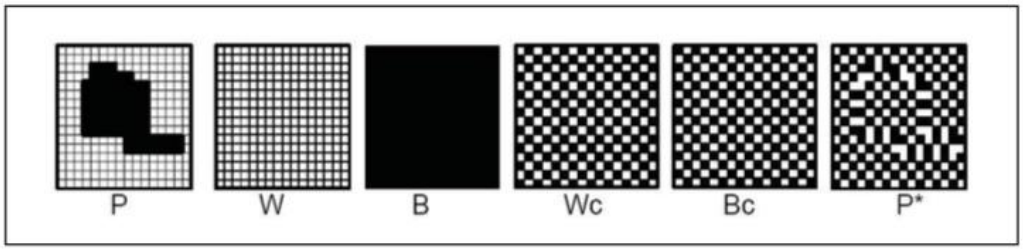


Figure 2. Conjugation Block

2.2 Vernam Cipher

The security or the data originality has become an important issue in data communication network [9]. Vernam Cipher is a crypto technique by combining each character of plain text with repeated key characters from a key stream. The key can be a random key or regular key by inserting decided characters. If a random key stream is used, the cipher text will be random too. By combining Vernam and BPCS will make the concealment perfectly. The attacker will be fooled by a set of bits obtained at the time of interception. It makes the combination of two methods work together. This following equation shows basic Vernam formula.

$$CT = PT \oplus Key \quad (1)$$

Where:

CT	:	Cipher Text
PT	:	Plain Text
Key	:	Password

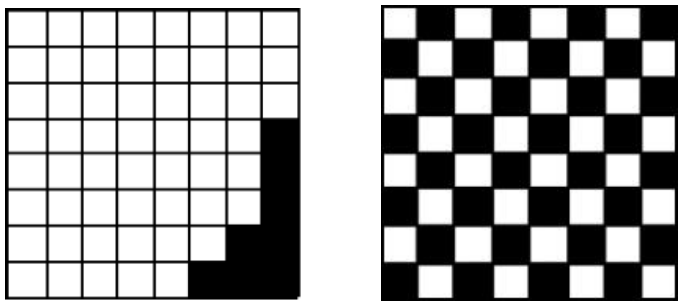
3. Proposed Work

This research proposes the conjugation block replacement with the encryption method. The BPCS always checks the message bit-plane whether it can be inserted. If only there is an informative region (a) in them, it will be replaced by the conjugation block (b) and put it back into the message. This following formula produces the complexity (α) of the blocks.

$$\alpha = \frac{k}{2 * 2^n * (2^n - 1)} \tag{2}$$

Where:

α : Complexity
k : Bit change
n : Block length



(a) (b)

Figure 3. Simple Block (a) and Complex Block (b)

Figure 3 explains the differences in complexity values in simple blocks and complex blocks. The complex block (b) has maximum border k = 112. The $\alpha = \frac{112}{2*2^3*(2^3-1)} = \frac{112}{112} = 1$. The value of 1 states that the block is fully complex. The Simple block (a) has k = 8, so the $\alpha = \frac{8}{2*2^3*(2^3-1)} = \frac{8}{112} = 0.0714285714285714$. Since block (a) does not meet the threshold requirement, it must be combined with another block to produce high complexity block. The following explanation illustrates what the author has to offer. A plain text of 8 characters will be converted to a bits-plane consisting of one segment. The key will be converted into a bit-plane as well which will then be encrypted using Vernam algorithm. The plain text (PT) is "**MUHAMMAD**".

$$PT = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

The number of bit change (PT) in the matrix is 46. The complexity value is $\alpha = \frac{48}{112} = 0.4285714285714286$. Assume the key (K) is "**SDM21HAL**". The following matrix show the bit values.

$$K = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

The number of bit change (K) in the matrix is 57. The complexity value is $\alpha = \frac{57}{112} = 0.5089285714285714$.

$$CT = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The number of bit change (CT) in the cipher matrix is 29. The complexity value is $\alpha = \frac{29}{112} = 0.2589285714285714$. The value obtained is 0.2589285714285714. This value is below the minimum threshold value (0.3). If imposed, this will have a negative impact on the image. Vessel image will have spots so suspicious people who see it. As a result, the information contained in the image can be picked up by hackers. Plain text may not be replaced with other information. The solution is to change the key that is used to produce a threshold value > 0.3.

4. Result and Discussion

This test will take 64 pixels of a picture as a sample which is spanned in N x N square matrix. M is the matrix container. The following matrix shows the 8 x 8 pixel of color intensity which converted to pure decimal binary code. It is taken from one of the RGB color content. The sample is only taken from one of the RGB colors. For example, Red color.

$$M = \begin{bmatrix} 236 & 131 & 11 & 10 & 98 & 201 & 104 & 41 \\ 122 & 220 & 48 & 94 & 180 & 132 & 202 & 215 \\ 204 & 171 & 24 & 116 & 92 & 92 & 254 & 19 \\ 47 & 3 & 171 & 15 & 27 & 239 & 6 & 125 \\ 156 & 209 & 63 & 2 & 130 & 235 & 249 & 53 \\ 74 & 153 & 77 & 221 & 119 & 77 & 7 & 172 \\ 187 & 152 & 112 & 89 & 220 & 221 & 127 & 224 \\ 151 & 157 & 64 & 128 & 111 & 84 & 21 & 151 \end{bmatrix}$$

Every segment consists of 64 pixels. The color intensity is measured from 0 to 255 as a byte value. The PBC form must be converted to CGC to remap the bit planes to perform the message insertion. Table 2 is the pure binary code, and Table 3 shows the canonical gray code.

Table 2. Pure Binary Code Segmentation

Pure Binary Code							
11101100	10000011	00001011	00001010	01100010	11001001	01101000	00101001
01111010	11011100	00110000	01011110	10110100	10000100	11001010	11010111
11001100	10101011	00011000	01110100	01011100	01011100	11111110	00010011
00101111	00000011	10101011	00001111	00011011	11101111	00000110	01111101
10011100	11010001	00111111	00000010	10000010	11101011	11111001	00110101
01001010	10011001	01001101	11011101	01110111	01001101	00000111	10101100
10111011	10011000	01110000	01011001	11011100	11011101	01111111	11100000
10010111	10011101	01000000	10000000	01101111	01010100	00010101	10010111

Table 3. Canonical Gray Code Segmentation

Canonical Gray Code							
10011010	11000010	00001110	00001111	01010011	10101101	01011100	00111101
01000111	10110010	00101000	01110001	11101110	11000110	10101111	10111100
10101010	11111110	00010100	01001110	01110010	01110010	10000001	00011010
00111000	00000010	11111110	00001000	00010110	10011000	00000101	01000011
11010010	10111001	00100000	00000011	11000011	10011110	10000101	00101111
01101111	11010101	01101011	10110011	01001100	01101011	00000100	11111010
11100110	11010100	01001000	01110101	10110010	10110011	01000000	10010000
11011100	11010011	01100000	11000000	01011000	01111110	00011111	11011100

The structure of both PBC and CGC are completely different. It has a technique to map the bits respectively. Figure 4 shows the difference of PBC and CGC image. The CGC turns the pixel by seeing the next bit of color intensity. After the CGC values are obtained, the segment is divided into eight bit-planes and categorize the same index of bit position into the same bit-plane. The following matrices show the complexity values.

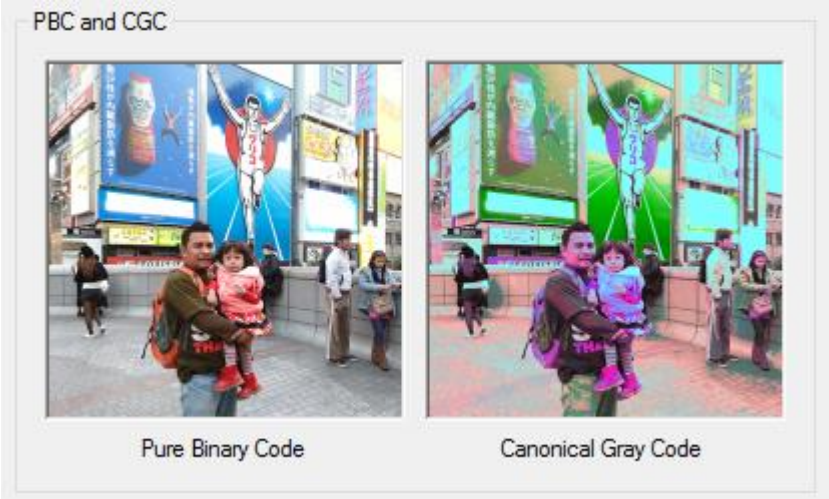


Figure 4. Pure Binary Code and Canonical Gray Code

$$BP1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} k &= 58 \\ \alpha &= \frac{58}{112} \\ &= 0.517857143 \end{aligned}$$

$$BP2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} k &= 57 \\ \alpha &= \frac{57}{112} \\ &= 0.508928571 \end{aligned}$$

$$BP3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\begin{aligned} k &= 54 \\ \alpha &= \frac{54}{112} \\ &= 0.482142857 \end{aligned}$$

$$BP4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{aligned} k &= 57 \\ \alpha &= \frac{57}{112} \\ &= 0.508928571 \end{aligned}$$

$$BP5 = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{aligned} k &= 62 \\ \alpha &= \frac{62}{112} \\ &= 0.553571429 \end{aligned}$$

$$BP6 = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{aligned} k &= 55 \\ \alpha &= \frac{55}{112} \\ &= 0.491071429 \end{aligned}$$

$$BP7 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\begin{aligned} k &= 52 \\ \alpha &= \frac{52}{112} \\ &= 0.464285714 \end{aligned}$$

$$BP8 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\begin{aligned} k &= 42 \\ \alpha &= \frac{42}{112} \\ &= 0.375 \end{aligned}$$

Each bit-plane is calculated to obtain the value of complexity. The complexity is counted by how many times the bit change from 0 to 1 and from 1 to 0. The maximal border change value of 8 x 8 block is 112. The formula to calculate the complexity is shown in Equation 2. Table 5 illustrates the complexity of all bit-planes. The threshold value must be set to limit between informative and noise-like region. The limit function is called threshold. The standard value of the threshold is 0.3, but it is adjustable. The modification is often performed to avoid the informative area or to add the noise-like area. The maximum changes are 112. However, after calculation, not all the bit-plane generate high complexity area. Some of them are in the informative regions.

Table 4. Bit-planes Complexity Value

Bit-plane	Bit Change	Complexity
BP1	58	0. 517857143
BP2	57	0. 508928571
BP3	54	0. 482142857
BP4	57	0. 508928571
BP5	66	0. 553571429
BP6	55	0.491071429
BP7	52	0.464285714
BP8	42	0.375

All bit-planes in Table 4 are noise-like regions. The insertion can be performed in any bit-plane from BP1 to BP7. The value of complexity is bigger than the threshold requirement. Assume the plain text is “ANDYSAH!”. The word consists of eight characters. It has eight bit-planes. Table 5 shows the original message and the bit-plane.

Table 5. Char, ASCII Code and Binary of Original Message

Char	ASCII Code	Biner
J	74	01001010
O	79	01001111
U	85	01010101
R	82	01010010
N	78	01001110
A	65	01000001
L	76	01001100
S	83	01010011

$$PT = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The Vernam algorithm needs the repeated key to making the cipher text. Then the key is "**THOMSONS**". The word is repeated until the length of bit-plane is covered. The following table describes the repeated key.

Table 7. Repeated Key ASCII

Char	ASCII Code	Binary
T	84	01010100
H	72	01001000
O	79	01001111
M	77	01001101
S	83	01010011
O	79	01001111
N	78	01001110
S	83	01010011

The key itself is converted to bit-plane model as showed in the following matrix. The bit-plane of the key is called key block. BPCS uses this key to convert informative region to noise-like region and encrypt the plain text.

$$K = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The bit-plane of message and key must be transformed by performing the exclusive-or (XOR). The result from both bit-planes will be the cipher bit-plane. After all the bit-planes is encrypted, the bit-planes are restored to its original position and re-converted to pure binary code before finally rewritten to the new image.

$$CT = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The above matrix shows the bit-plane after encrypted and Table 8 shows the bit-plane after conversion to ASCII code. The message earlier is turned into the encrypted message.

Table 8. Cipher Text

ASCII Code	Binary
30	00011110
7	00000111
26	00011010
31	00011111
29	00011101
14	00001110
2	00000010
0	00000000

The modification of the conjugation block to encrypted block can increase the security level. Each bit-plane is encrypted with the key desired. There are many bit-planes created in making the encryption. The key bit-planes can be specified from a single word to repetitive word.

5. Conclusion

BPCS is good at hiding information. It consists of many bit-planes where the plain text is securely embedded. BPCS technique can save plain text up to 70% of its capacity. It has conjugation block that can be encrypted using cryptography method. It is Vernam Cipher. It is more suitable to reconstruct the bit-planes because this encryption method is very light but powerful. The calculation does not have to use difficult a mathematical operation. The way is just only to remap the bit-plane after calculation the complexity. Applying the Vernam method in the bit-planes will strengthen the security level. The important thing to keep the bit-planes secure is to find the proper key to produce the noise-like region. The stego-image produced has already been an encryption vessel image. It is the way to make the concealment perfect.

6. Future Scope

BPCS has a formation of 8 x 8 matrices. In the future, this can be modified to better or with different combinations of f (x, y) so that information is safer from the theft of irresponsible parties. The combination of bit-plane that has various sizes can trick hackers in the process of unloading messages on the vessel image. Bit-plane modifications and encryption combinations with public key methods can be applied in the future to make the information stored in the vessel image better. Customization on bit-plane complexity also affects stored information. The better way to customize bit-plane, the harder the vessel image to solve.

References

[1] A. P. U. Siahaan, "High Complexity Bit-plane Security Enchancement in BPCS Steganography, International Journal of Computer Applications, vol. 148, no. 3, **(2016)**, pp. 17-22.
[2] P. Lahane, Y. Kumbhar, S. Patil, S. More and M. Barse, "Data Security Using Visual Cryptography and Bit Plane Complexity Segmentation," International Journal of Emerging Engineering Research and Technology, vol. 2, no. 8, **(2014)**, pp. 40-44.
[3] M. Niimi, H. Noda and E. Kawaguchi, "A Steganography Based on Region Segmentation by Using Complexity Measure", IEICE, vol. 81, no. 6, **(1998)**, pp. 1132-1140.
[4] S. Singh and T. J. Siddiqui, "A Security Enhanced Robust Steganography Algorithm for Data Hiding", International Journal of Computer Science, vol. 9, no. 3, **(2012)**, pp. 131-139.

[5] S. Toosizadeh and S. M. R. Farshchi, "A Hybrid Steganography Algorithm Based on Chaos & BPCS", Proceedings of the World Congress in Computer Science, Computer Engineering and Applied Computing, **(2012)**; Mashhad, Iran.
[6] E. Kawaguchi and R. O. Eason, "Principle and Applications of BPCS-Steganography", International Symposium on Voice, **(1998)**.
[7] A. P. U. Siahaan, "Dynamic Key Matrix of Hill Cipher Using Genetic Algorithm", International Journal of Security and Its Applications, vol. 10, no. 8, **(2016)**, pp. 173-180.
[8] J. Spaulding, H. Noda, M. N.Shirazi and E. Kawaguchi, "BPCS Steganography Using EZW Lossy Compressed Images" , Pattern Recognition Letters, vol. 3, no. 12, **(2002)**, pp. 1579–1587.
[9] P. Banerjee and A. Nath, "Bit and Byte Level Generalized Modified Vernam Cipher Method with Feedback", International Journal of Computer Applications, vol. 64, no. 2, **(2013)**, pp. 9-15,.
[10] B. Ryabko, "The Vernam Cipher is Robust to Small Deviations From Randomness", Problems of Information Transmission, vol. 51, no. 1, **(2015)**, pp. 82-86.
[11] M. Verma, "Modern Image Security Mechanism using Hill and Vernam Cipher", International Journal of Engineering Research & Technology, vol. 3, no. 2, **(2014)**, pp. 1251-1254.
[12] R. Rusia, M. K. Mishra and R. K. Tiwari, "More Advanced Steganography Using BPCS", International Journal of Computer Engineering and Applications, vol. 8, no. 2, **(2014)**, pp. 264-282.
[13] C. Jain, V. Parate, A. Dhamanikar and 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, **(2015)**, pp. 60-65.
[14] K. Ramani, E. V. Prasad and S. Varadarajan, "Steganography Using Bpcs to The Integer Wavelet Transformed Image", International Journal of Computer Science and Network Security, vol. 7, no. 7, **(2007)**, pp. 293-302.
[15] V. J. Patel and N. R. Soni, "Image Steganography System Using Modified BPCS Steganography Method", International Journal of Engineering Research & Technology, vol. 3, no. 6, **(2014)**, pp. 728-730.

Author

Andysah Putera Utama Siahaan was born in 1980, Medan, Indonesia. He received the bachelor degree in computer science from Universitas Pembangunan Panca Budi, Medan, Indonesia, in 2010, and in 2012, he obtained his master degree from the University of North Sumatera, Medan, Indonesia. In 2010, he joined as a lecturer in Faculty of Computer Science, Universitas Pembangunan Panca Budi. He has been a researcher since 2012. He has been studying his Ph.D. degree for two years. He is now active in writing international journals and conferences.

