

## Implementasi Metode Bit Plane Complexity Segmentation pada Citra Digital dalam Penyembunyian Pesan Rahasia

Habibi Z

STMIK Budi Darma Medan, Jl. SM.Raja No.338 Sp.Limun Medan, Sumut, Indonesia  
E-Mail: h4bibizz@yahoo.com

### ABSTRACT

Tapping secret message information often occurs in communication media. Safeguarding information or data distributed is very important to maintain confidentiality, integrity and authenticity. In steganography, there are several digital media that can be used as covers to hide the existence of a message, such as: Image, Audio, Text, and Video. The cover media research used in digital imagery is the Bit-Plane Complexity Segmentation Method. Bit-plane complexity segmentation (BPCS) is a steganography technique that has a large capacity, because it can hold confidential data with a relatively large capacity when compared to other steganography methods. It is expected that this research can minimize the leakage of important information that is very confidential and detrimental if it is known by others.

Keywords: Steganography, Digital Image, Bit-Plane Complexity Segmentation Method.

### PENDAHULUAN

Untuk berbagai alasan, keamanan dan kerahasiaan sangat dibutuhkan dalam komunikasi data. Terdapat beberapa usaha untuk menangani masalah keamanan data rahasia yang dikirimkan melalui internet, di antaranya adalah menggunakan teknik kriptografi dan steganografi[1]. Steganografi lebih mengurangi kecurigaan karena pesan yang disamarkan disembunyikan ke dalam pesan lainnya. Steganografi dapat menyamarkan pesan ke dalam suatu media tanpa orang lain menyadari bahwa media tersebut telah disisipi suatu pesan, karena hasil keluaran steganografi adalah data yang memiliki bentuk persepsi yang sama dengan data aslinya apabila dilihat menggunakan indera manusia, sedangkan perubahan pesan dalam kriptografi dapat dilihat dan disadari langsung oleh indera manusia. Pada steganografi, data rahasia disisipkan pada data lain yang disebut *cover-object* dan menghasilkan *stego-object* (hasil steganografi). Media penampung yang umum digunakan pada teknik steganografi adalah gambar, suara, video, atau teks. Adapun data yang disimpan juga dapat berupa gambar, suara, video, teks, atau pesan lain[2]. Steganografi yang diterapkan adalah steganografi pada dokumen citra (gambar)[3]. Ada banyak metode yang digunakan untuk steganografi pada dokumen citra seperti metode *Least Significant Bit (LSB)*, *Spreed Spectrum Steganography* dan *Bit-Plane Complexity Segmentation (BPCS)*.

Metode steganografi yang digunakan adalah metode *Bit-Plane Complexity Segmentation (BPCS)* adalah salah satu metode dari *steganografi*. Teknik ini bekerja dengan memanfaatkan karakteristik pengelihat manusia yang tidak bisa mengerti bentuk informasi dalam suatu pola biner yang sangat rumit[4]. Dengan metode *Bit-Plane Complexity Segmentation (BPCS)*, setiap *byte* pada data rahasia dibagi menjadi blok blok, dimana data rahasia tersebut berada dalam *citra digital*, yang dapat dilakukan pergantian wilayah yang "*noise-like*" pada *cover-image* dengan data rahasia, tanpa merusak kualitas *cover-image*. Jadi, data rahasia yang berada dalam citra digital hanya si pengirim dan si penerima yang dapat melihatnya.

### LANDASAN TEORI

#### 2.1 Implementasi

Implementasi adalah suatu aktifitas, aksi, tindakan adanya suatu sistem. Implementasi bukan sekedar aktivitas tetapi suatu kegiatan yang terencana yang harus dicapai untuk memenuhi tujuan kegiatan[5]

Berdasarkan defenisi di atas, maka dapat disimpulkan bahwa implementasi bermuara pada mekanisme suatu sistem. Ungkapan mekanisme mengandung arti bahwa implementasi bukan sekedar aktivitas, tetapi suatu kegiatan yang terencana yang dilakukan secara sungguh-sungguh berdasarkan acuan norma tertentu untuk mencapai tujuan kegiatan. Implementasi dalam kenyataannya berupa gagasan dan

tolak ukur suatu tindakan individu yang diarahkan pada tujuan serta ditetapkan untuk mencapai kebijakan yang mampu memberi hasil yang bersifat praktis terhadap suatu sistem.

## 2.2 Bit-Plane Complexity Segmentation (BPCS)

*Bit-plane complexity segmentation (BPCS)* merupakan teknik steganografi yang diperkenalkan oleh Eiji Kawaguchi dan Richard O. Eason pada tahun 1998. Teknik ini merupakan teknik steganografi yang memiliki kapasitas besar, karena dapat menampung data rahasia dengan kapasitas yang relatif besar jika dibandingkan dengan metode steganografi lain seperti *LSB (Least Significant Bit)*[2]. Teknik *BPCS* ini adalah teknik steganografi yang tidak berdasarkan teknik pemrograman, tetapi teknik yang menggunakan sifat penglihatan manusia. Sifat penglihatan manusia yang dimanfaatkan yaitu ketidakmampuan manusia menginterpretasi pola biner yang sangat rumit.

Dokumen citra tersebut dibagi menjadi beberapa segmen dengan ukuran 8x8 piksel setiap segmennya (Kawaguchi dan Eason, 1998). Pada dokumen citra 8-bit, setiap satu segmen akan memiliki 8 buah *bit plane* yang merepresentasikan piksel-piksel dari setiap bit tersebut. Proses pembagian segmen 8x8 piksel menjadi 8 buah *bit plane* disebut proses *bit slicing*. Representasi kedelapan *bit plane* ini merupakan *PBC system (Pure Binary Code)*. Pada *BPCS*, proses penyisipan dilakukan pada *bit plane* dengan sistem *CGC (Canonical Gray Code)* karena proses *bit slicing* pada *CGC* cenderung lebih baik dibandingkan pada *PBC* (Kawaguchi dan Eason, 1998). Sehingga pada proses penyisipan, *bit plane* dengan representasi *PBC* diubah menjadi *bit plane* dengan representasi *CGC*.

Proses penyisipan pesan dilakukan pada segmen yang memiliki kompleksitas yang tinggi. Segmen yang memiliki kompleksitas tinggi ini disebut *noise-like regions*. Pada segmen-segmen ini penyisipan dilakukan tidak hanya pada *least significant bit*, tapi pada seluruh *bit plane* yang termasuk *noise-like regions*. Oleh sebab itu, pada teknik *BPCS*, kapasitas data yang disisipkan dapat mencapai 50% dari ukuran *coverimagenya* [1][2].

## 2.3 Algoritma BPCS (Bit Plane Complexity Segmentation)

Langkah-langkah yang dilakukan pada algoritma *BPCS* pada saat menyisipkan data adalah sebagai berikut[2][5]:

1. *Cover image* dengan sistem *PBC* diubah menjadi sistem *CGC*, kemudian gambar tersebut di-*slice* menjadi *bit-plane* dalam bentuk gambar biner. Setiap *bit-plane* mewakili bit dari setiap piksel pada gambar.
2. Segmentasi setiap *bit-plane* pada *cover image* menjadi *informative* dan *noise like region* dengan menggunakan nilai batas/*threshold* ( $\alpha$ ). Nilai umum dari  $\alpha = 0,3$ .
3. Kelompokkan byte-byte pesan rahasia menjadi rangkaian blok pesan rahasia.
4. Jika blok(S) kurang kompleks dibandingkan dengan nilai batas, maka lakukan konjugasi terhadap S untuk mendapatkan S\* yang lebih kompleks. Blok konjugasi(S\*) pasti lebih kompleks dibandingkan dengan nilai batas.
5. Sisipkan setiap blok pesan rahasia ke *bit-plane* yang merupakan *noise-like region* (atau gantikan semua bit pada *noise-like region*). Jika blok S dikonjugasi, maka simpan data pada "conjugation map".
6. Sisipkan juga *conjugation map* seperti yang dilakukan pada blok pesan rahasia.
7. Ubah *stego-image* dari sistem *CGC* menjadi sistem *PBC*.

Proses ekstraksi pesan rahasia dapat dilakukan dengan menerapkan langkah-langkah penyisipan secara terbalik. Saat proses ekstraksi pesan, yang perlu dilakukan hanyalah mengambil segmen bit yang memiliki kompleksitas diatas *threshold*. Jika nilai kompleksitas segmen tersebut lebih besar dari *threshold* [6], maka segmen tersebut merupakan bagian dari pesan rahasia.

## PEMBAHASAN

### 3.1. Analisa Penerapan Metode BPCS Dalam Penyisipan Pesan

Sebagai contoh permasalahan yaitu gambar yang digunakan sebagai *cover image* adalah gambar dengan format *BMP* yang menggunakan jenis pewarnaan *RGB* dengan kedalaman 24 bit dan pesan yang disisipkan merupakan pesan teks yaitu "kill the king tonight". Langkah-langkah dalam penyisipan pesan adalah sebagai berikut :

1. Pembagian gambar menjadi segmen-segmen berukuran 8x8 piksel. Nilai intensitas dari segmen pertama adalah sebagai berikut :



R=226 G=137 B=125	R=226 G=137 B=125	R=223 G=137 B=133	R=223 G=136 B=128	R=226 G=138 B=120	R=226 G=129 B=116	R=228 G=138 B=123	R=227 G=134 B=124
R=226 G=137 B=125	R=226 G=137 B=125	R=223 G=137 B=133	R=223 G=136 B=128	R=226 G=138 B=120	R=226 G=129 B=116	R=228 G=138 B=123	R=227 G=134 B=124
R=226 G=137 B=125	R=226 G=137 B=125	R=223 G=137 B=133	R=223 G=136 B=128	R=226 G=138 B=120	R=226 G=129 B=116	R=228 G=138 B=123	R=227 G=134 B=124
R=226 G=137 B=125	R=226 G=137 B=125	R=223 G=137 B=133	R=223 G=136 B=128	R=226 G=138 B=120	R=226 G=129 B=116	R=228 G=138 B=123	R=227 G=134 B=124
R=227 G=140 B=123	R=227 G=140 B=123	R=227 G=131 B=113	R=222 G=130 B=111	R=226 G=130 B=120	R=228 G=133 B=115	R=226 G=132 B=120	R=230 G=133 B=113
R=228 G=134	R=228 G=134	R=225 G=141	R=224 G=133	R=225 G=134	R=229 G=137	R=229 G=132	R=226 G=128

Gambar 1 Segmen Citra

Nilai-nilai desimal pixel diatas dikonversikan ke biner

- Nilai intensitas dari masing-masing komponen warna direpresentasikan dengannilai binernya dan nilai biner tersebut diubah dari sistem PBC menjadi CGC. Kemudian dilakukan proses bit-plane slicing untuk membagi gambar menjadi bitplane dan hitung nilai kompleksitas ( $\alpha$ ) dari masing-masing bit-plane.

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna merah (Red) :

11100010	11100010	11011111	11011111	11100010	11100010	11100100	11100011
11100010	11100010	11011111	11011111	11100010	11100010	11100100	11100011
11100010	11100010	11011111	11011111	11100010	11100010	11100100	11100011
11100010	11100010	11011111	11011111	11100010	11100010	11100100	11100011
11100010	11100010	11011111	11011111	11100010	11100010	11100100	11100011
11100011	11100011	11100011	11011110	11100010	11100010	11100010	11100110
11100100	11100100	11100001	11100000	11100101	11100101	11100101	11100101
11011111	11011111	11100010	11011101	11100011	11100010	11100010	11100100

Representasi biner dengan sistem CGC dari nilai intensitas setiap piksel pada warna merah (Red) :

10010011	10010011	10110000	10110000	10010011	10010011	10010110	10010010
10010011	10010011	10110000	10110000	10010011	10010011	10010110	10010010
10010011	10010011	10110000	10110000	10010011	10010011	10010110	10010010
10010011	10010011	10110000	10110000	10010011	10010011	10010110	10010010
10010011	10010011	10110000	10110000	10010011	10010011	10010110	10010010
10010010	10010010	10010001	10110001	10010011	10010110	10010011	10010101
10010110	10010110	10010001	10010000	10010001	10010111	10010111	10010111
10110000	10110000	10010011	10110011	10010010	10010001	10010011	10010110

Hasil bit-plane slicing dari nilai intensitas piksel pada warna merah (Red) adalah sebagai berikut :

- Bit-plane 1 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 2 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 3 Nilai kompleksitas ( $\alpha$ ) =  $20/112=0.18$
- Bit-plane 4 Nilai kompleksitas ( $\alpha$ ) = 0

- Bit-plane 5 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 6 Nilai kompleksitas ( $\alpha$ ) =  $26/112=0,23$
- Bit-plane 7 Nilai kompleksitas ( $\alpha$ ) =  $29/112=0,26$
- Bit-plane 8 Nilai kompleksitas ( $\alpha$ ) =  $37/111$

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna hijau (Green) :

10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001001	10001001	10001001	10001000	10001010	10000001	10001010	10000110
10001100	10001100	10000011	10000010	10001000	10000101	10000100	10000101
10000110	10000110	10001101	10000101	10000110	10001001	10000100	10000000
10000101	10000101	10000001	10000100	10000011	10000101	10000001	10000011

Representasi biner dengan sistem CGC dari nilai intensitas setiap piksel pada warna hijau (Green) :

11001101	11001101	11001101	11001100	11001111	11000001	11001111	11000101
11001101	11001101	11001101	11001100	11001111	11000001	11001111	11000101
11001101	11001101	11001101	11001100	11001111	11000001	11001111	11000101
11001101	11001101	11001101	11001100	11001111	11000001	11001111	11000101
11001101	11001101	11001101	11001100	11001111	11000001	11001111	11000101
11001010	11001010	11000010	11000011	11001100	11000111	11000110	11000111
11000111	11000101	11001011	11000111	11000101	11001101	11000110	11000000
11000111	11000111	11000001	11000110	11000010	11000111	11000001	11000010

Hasil bit-plane slicing dari nilai intensitas piksel pada warna hijau (Green) adalah sebagai berikut :

- Bit-plane 1 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 2 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 3 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 4 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 5 Nilai kompleksitas ( $\alpha$ ) =  $32/112=0.29$
- Bit-plane 6 Nilai kompleksitas ( $\alpha$ ) =  $30/112=0.27$
- Bit-plane 7 Nilai kompleksitas ( $\alpha$ ) =  $49/112=0.44$
- Bit-plane 8 Nilai kompleksitas ( $\alpha$ ) =  $33/112=0.2$

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna biru (Blue) :

01111101	01111101	10000101	10000000	01111100	01110100	01111011	01111100
01111101	01111101	10000101	10000000	01111100	01110100	01111011	01111100
01111101	01111101	10000101	10000000	01111100	01110100	01111011	01111100
01111101	01111101	10000101	10000000	01111100	01110100	01111011	01111100
01111101	01111101	10000101	10000000	01111100	01110100	01111011	01111100
01111011	01111011	01110001	01101111	01111000	01110011	01111000	01110001
01110111	01110111	01110100	01110011	01111101	01110000	01110100	01101001
01111001	01111001	01101010	01110010	01111000	01110100	01110000	01101010

Representasi biner dengan sistem CGC dari nilai intensitas setiap piksel pada warna biru (Blue) :

01000011	01000011	11000111	11000000	01000100	01001110	01000110	01000010
01000011	01000011	11000111	11000000	01000100	01001110	01000110	01000010
01000011	01000011	11000111	11000000	01000100	01001110	01000110	01000010
01000011	01000011	11000111	11000000	01000100	01001110	01000110	01000010
01000011	01000011	11000111	11000000	01000100	01001110	01000110	01000010
01000110	01000110	01001001	01001001	01000100	01001010	01000100	01001001
01001100	01001100	01001110	01001010	01000011	01001000	01001110	01011101
01000101	01000101	01011111	01001011	01000100	01001110	01001000	01011111

Hasil *bit-plane slicing* dari nilai intensitas piksel pada warna biru (Blue) adalah sebagai berikut :

- Bit-plane 1 Nilai kompleksitas ( $\alpha$ ) =  $12/112 = 0,11$
- Bit-plane 2 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 3 Nilai kompleksitas ( $\alpha$ ) = 0
- Bit-plane 4 Nilai kompleksitas ( $\alpha$ ) =  $6/112 = 0,05$
- Bit-plane 5 Nilai kompleksitas ( $\alpha$ ) =  $28/112 = 0,25$
- Bit-plane 6 Nilai kompleksitas ( $\alpha$ ) =  $42/112 = 0,38$
- Bit-plane 7 Nilai kompleksitas ( $\alpha$ ) =  $36/112 = 0,32$
- Bit-plane 8 Nilai kompleksitas ( $\alpha$ ) =  $25/112 = 0,2$

3. Tentukan *bit-plane* yang *noise like* dan *informative* dengan *threshold* ( $\alpha_0$ ) = 0,3. Jadi, *bit-plane* yang *noise like* ( $\alpha > 0,3$ ) dan siap untuk disisipi pesan adalah *bit-plane 8 Red*, *bit-plane 7 Green*, *bit-plane 6 Blue*, dan *bit-plane 7 Blue*.

4. Baca pesan rahasia sebagai string pada karakter ASCII dan representasikan dengan nilai binernya, kemudian bentuk pesan rahasia menjadi blok 8x8 dan hitung nilai kompleksitas ( $\alpha$ ) dari setiap blok.

Pesan rahasia: kill the king tonight  
 Nilai ASCII : 75 105 108 108 32 116 104 101 32 107 105 110 103 32 116 111 110 105 103 104 116

Representasi biner pesan rahasia :

01001011 01101001 01101100  
 01101100 00100000 01110100  
 01101000 01100101 00100000  
 01101011 01101001 01101110  
 01100111 00100000 01110100  
 01101111 01101110 01101001  
 01100111 01101000 01110100

Blok-blok pesan rahasia:

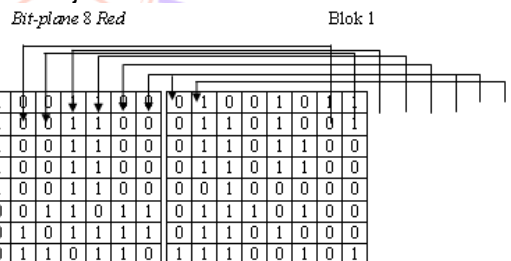
- Blok 1 Nilai kompleksitas ( $\alpha$ ) =  $49/112 = 0,44$
- Blok 2 Nilai kompleksitas ( $\alpha$ ) =  $49/112 = 0,44$
- Blok 3 Nilai kompleksitas ( $\alpha$ ) =  $37/112 = 0,33$

5. Bentuk peta konjugasi blok pesan rahasia. Karena blok pesan rahasia hanya 3 maka lakukan *padding* (menambahkan nilai '0' pada bit-bit terakhir) agar peta konjugasi dapat dibentuk menjadi blok 8x8. Peta konjugasi dari blok pesan rahasia: Nilai kompleksitas ( $\alpha$ ) = 0

Peta konjugasi tidak kompleks ( $\alpha < 0,3$ ) sehingga harus dikonjugasi dengan cara meng-XOR-kannya dengan blok  $W_c$ . Setelah dikonjugasi, peta konjugasi menjadi:

1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1

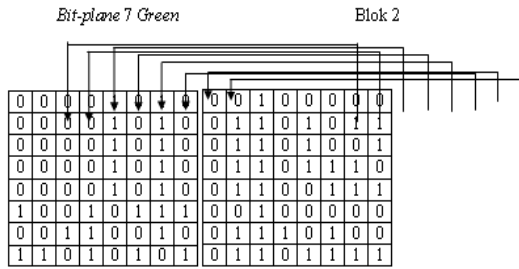
Penyisipan pesan dilakukan dengan cara mengganti *bit-plane* yang *noise like* pada gambar dengan blok-blok pesan rahasia. *Bit-plane 8 Red* diganti dengan blok 1 pesan rahasia, seperti yang ditunjukkan berikut:



Sehingga isi *Bit-plane 8 Red* menjadi :

0	1	0	0	1	0	1	1
0	1	1	0	1	0	0	1
0	1	1	0	1	1	0	0
0	1	1	0	1	1	0	0
0	0	1	0	0	0	0	0
0	1	1	1	0	1	0	0
0	1	1	0	1	0	0	0
0	1	1	0	0	1	0	1

*Bit-plane 7 Green* diganti dengan blok 2 pesan rahasia, seperti yang ditunjukkan berikut:



1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1

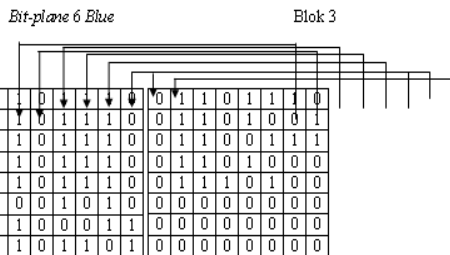
Karena nilai *bit-plane* 8 pada warna merah (*Red*) diganti dengan blok 1 pesan rahasia, maka nilai bit kedelapan yang merepresentasikan nilai intensitas setiap piksel pada warna merah (*Red*) berubah, dalam sistem CGC menjadi:

Sehingga isi *Bit-plane* 7 Green menjadi :

0	0	1	0	0	0	0	0
0	1	1	0	1	0	1	1
0	1	1	0	1	0	0	1
0	1	1	0	1	1	1	0
0	1	1	0	0	1	1	1
0	0	1	0	0	0	0	0
0	1	1	1	0	1	0	0
0	1	1	0	1	1	1	1

10010010	10010011	10110000	10110000	10010011	10010010	10010111	10010011
10010010	10010011	10110001	10110000	10010011	10010010	10010110	10010011
10010010	10010011	10110001	10110000	10010011	10010011	10010110	10010010
10010010	10010011	10110001	10110000	10010011	10010011	10010110	10010010
10010010	10010011	10110001	10110000	10010010	10010010	10010110	10010010
10010010	10010011	10010011	10110001	10010010	10010111	10010010	10010100
10010110	10010111	10010001	10010000	10010001	10010110	10010110	10010110
10110000	10110001	10010011	10110010	10010010	10010001	10010010	10010111

*Bit-plane* 6 Blue diganti dengan blok 3 pesan rahasia, seperti yang ditunjukkan berikut:



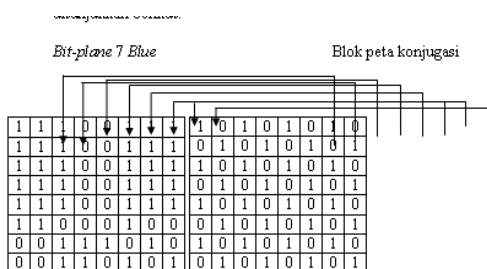
Karena nilai *bit-plane* 7 pada warna hijau (*Green*) diganti dengan blok 2 pesan rahasia, maka nilai bit ketujuh yang merepresentasikan nilai intensitas setiap piksel pada warna hijau (*Green*) berubah, dalam sistem CGC menjadi:

Sehingga isi *Bit-plane* 6 Blue menjadi :

0	1	1	0	1	1	1	0
0	1	1	0	1	0	0	1
0	1	1	0	0	1	1	1
0	1	1	0	1	0	0	0
0	1	1	1	0	1	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

11001101	11001101	11001111	11001100	11001101	11000001	11001101	11000101
11001101	11001111	11001111	11001100	11001111	11000001	11001111	11000111
11001101	11001111	11001111	11001100	11001111	11000001	11001101	11000111
11001101	11001111	11001111	11001100	11001111	11000011	11001111	11000101
11001101	11001111	11001111	11001100	11001101	11000011	11001111	11000111
11001000	11001000	11000010	11000001	11001100	11000101	11000100	11000101
11000101	11000111	11001011	11000111	11000101	11001111	11000100	11000000
11000101	11000111	11000011	11000100	11000010	11000111	11000011	11000010

*Bit-plane* 7 Blue diganti dengan blok peta konjugasi pesan rahasia, seperti yang ditunjukkan berikut:



Karena nilai *bit-plane* 6 dan 7 pada warna biru (*Blue*) diganti dengan blok 3 pesan rahasia dan peta konjugasi pesan rahasia, maka nilai bit keenam dan ketujuh yang merepresentasikan nilai intensitas setiap piksel pada warna biru (*Blue*) berubah, dalam sistem CGC menjadi:

Sehingga isi *Bit-plane* 7 Blue menjadi :

1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1

01000011	01000101	11000111	11000000	01000110	01001100	01000110	01000000
01000001	01000111	11000101	11000010	01000100	01001010	01000000	01000110
01000011	01000101	11000111	11000000	01000010	01001100	01000110	01000100
01000001	01000111	11000101	11000010	01000100	01001010	01000000	01000010
01000011	01000101	11000111	11000100	01000010	01001100	01000010	01000000
01000000	01000010	01001001	01001011	01000000	01001010	01000000	01001011
01001010	01001000	01001010	01001000	01000011	01001000	01001010	01011001
01000001	01000011	01010010	01001011	01000000	01001010	01001000	01011011

Nilai intensitas setiap piksel pada masing-masing warna yang direpresentasikan dengan sistem CGC diubah kembali menjadi sistem PBC.

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna gambar merah (*Red*) setelah disisipi pesan :

11100011	11100010	11011111	11011111	11100010	11100011	11100101	11100010
11100011	11100010	11011110	11011111	11100010	11100011	11100100	11100010
11100011	11100010	11011110	11011111	11100010	11100010	11100100	11100011
11100011	11100010	11011110	11011111	11100010	11100010	11100100	11100011
11100011	11100011	11011110	11011111	11100011	11100011	11100100	11100011
11100011	11100010	11100010	11011110	11100011	11100101	11100011	11100111
11100100	11100101	11100001	11100000	11100001	11100100	11100100	11100100
11011111	11011110	11100010	11011100	11100011	11100001	11100011	11100101

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna gambar hijau (Green) setelah disisipi pesan :

10001001	10001001	10001010	10001000	10001001	10000001	10001001	10000110
10001001	10001010	10001010	10001000	10001010	10000001	10001010	10000101
10001001	10001010	10001010	10001000	10001010	10000001	10001001	10000101
10001001	10001010	10001010	10001000	10001010	10000010	10001010	10000110
10001001	10001010	10001010	10001001	10001001	10000010	10001010	10000101
10001111	10001111	10000011	10000001	10001000	10000110	10000111	10000110
10000110	10000101	10000101	10000101	10000110	10000110	10000111	10000000
10000110	10000101	10000010	10000111	10000011	10000101	10000010	10000011

Representasi biner dengan sistem PBC dari nilai intensitas setiap piksel pada warna gambar biru (Blue) setelah disisipi pesan :

01111101	01111001	10000101	10000000	01110111	01110111	01111011	01111111
01111110	01111010	10000110	10000011	01111000	01110010	01111111	01111011
01111110	01111001	10000101	10000000	01111100	01110111	01111011	01111011
01111110	01111010	10000110	10000011	01111000	01110011	01111111	01111100
01111101	01111001	10000101	10000111	01111100	01110111	01111000	01111111
01111111	01111100	01110001	01101101	01111111	01110011	01111111	01110010
01110011	01110000	01110011	01110000	01111101	01110000	01110011	01101110
01111110	01111101	01101110	01110010	01111111	01110011	01110000	01101101

Nilai intensitas segmen pertama pada gambar setelah disisipi pesan menjadi:

R=227 G=137 B=125	R=226 G=137 B=121	R=223 G=138 B=133	R=223 G=136 B=128	R=226 G=137 B=123	R=227 G=129 B=123	R=229 G=137 B=123	R=226 G=134 B=127
R=227 G=137 B=126	R=226 G=138 B=122	R=222 G=138 B=134	R=223 G=136 B=131	R=226 G=138 B=120	R=227 G=129 B=114	R=228 G=138 B=127	R=226 G=133 B=123
R=227 G=137 B=126	R=226 G=138 B=121	R=222 G=138 B=133	R=223 G=136 B=128	R=226 G=138 B=124	R=226 G=129 B=123	R=228 G=137 B=123	R=227 G=133 B=120
R=227 G=137 B=126	R=226 G=138 B=122	R=222 G=138 B=134	R=223 G=136 B=131	R=226 G=138 B=120	R=227 G=130 B=115	R=228 G=138 B=127	R=227 G=134 B=124
R=227 G=137 B=125	R=227 G=138 B=121	R=222 G=138 B=133	R=223 G=137 B=135	R=227 G=137 B=124	R=227 G=130 B=123	R=228 G=138 B=124	R=227 G=133 B=127
R=227 G=143 B=127	R=226 G=143 B=124	R=226 G=131 B=113	R=222 G=129 B=113	R=227 G=136 B=127	R=229 G=134 B=115	R=227 G=135 B=127	R=231 G=134 B=114
R=228 G=134 B=115	R=229 G=133 B=112	R=225 G=141 B=115	R=224 G=133 B=112	R=225 G=134 B=125	R=228 G=138 B=112	R=228 G=135 B=115	R=228 G=128 B=110
R=223 G=134 B=126	R=222 G=133 B=125	R=226 G=130 B=110	R=220 G=135 B=114	R=227 G=131 B=127	R=225 G=133 B=115	R=227 G=130 B=112	R=229 G=131 B=109

**3.2. Analisa Ekstraksi Pesan**

Proses ekstraksi pesan merupakan kebalikan dari proses penyisipan. Langkah awal dalam proses ekstraksi ini yaitu membaca header dokumen untuk mengetahui jenis warna dan kedalaman bit yang digunakan oleh dokumen BMP agar tidak terjadi kesalahan dalam proses pembentukan bit-plane. Selanjutnya dilakukan dekompresi terhadap dokumen citra BMP untuk mendapatkan nilai bit yang merepresentasikan nilai intensitas yang sebenarnya pada setiap piksel. Nilai intensitas ini akan dikonversi terlebih dahulu

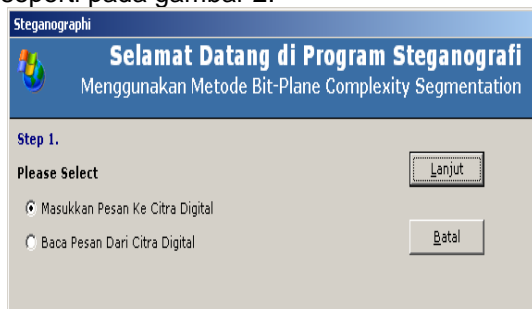
dari sistem PBC menjadi sistem CGC. Kemudian gambar (cover image) dibagi menjadi beberapa segmen yang berukuran 8x8 piksel setiap segmennya. Setiap segmen akan di-slicing atau diuraikan menjadi beberapa bit-plane seperti yang dilakukan pada proses penyisipan pesan. Setiap bit-plane tersebut dibentuk menjadi gambar biner 8x8 dan kemudian dihitung nilai kompleksitas dari masing-masing gambar biner tersebut. Gambar biner yang memiliki kompleksitas tinggi ( $\alpha > \alpha_0$ ) kemungkinan memiliki pesan yang akan diekstraksi [7].

Ambil tiga gambar biner pertama yang kompleks pada gambar. Gambar biner pertama dan kedua berisi header pesan dan gambar biner ketiga berisi peta konjugasinya. Jika bit pertama pada gambar biner yang berisi peta konjugasi bernilai '1', maka gambar biner tersebut telah dikonjugasi yang berarti harus dikonjugasi kembali untuk mendapatkan informasi yang sebenarnya. Ubah gambar biner 8x8 ini ke dalam rangkaian 63 bit (tidak termasuk bit pertama) dan gunakan 2 bit pertama dalam rangkaian ini untuk melihat apakah 2 gambar biner pertama telah dikonjugasi atau tidak. Jika nilai bit bernilai '1' maka gambar biner yang bersesuaian telah dikonjugasi dan perlu dikonjugasi kembali untuk mendapatkan informasi yang sebenarnya. Sebaliknya jika nilai bit bernilai '0' maka gambar biner yang bersesuaian tidak perlu dikonjugasi karena sudah berisi informasi yang sebenarnya. Setelah informasi yang sebenarnya dari kedua gambar biner pertama diperoleh, maka ukuran dan nama pesan dapat diketahui. Misalkan jumlah gambar biner 8x8 yang telah disisipi pesan sebanyak N dimana nilai N merupakan hasil dari ukuran file (file size) dibagi 8 byte. Jumlah peta konjugasi pesan adalah N/63. Apabila hasil bagi bukan merupakan bilangan bulat maka dilakukan pembulatan ke atas. Jadi, gambar biner keempat sampai gambar biner ke-N + 3 yang kompleks pada gambar berisi N blok pesan rahasia tersebut dan gambar biner ke-N + 4 yang kompleks pada gambar berisi peta konjugasi pesan. Sebelum pembentukan kembali pesan rahasia dari gambar biner terlebih dahulu dilakukan pengecekan pada nilai bit pertama dari masing-masing peta konjugasi. Apabila nilai bit pertama tersebut bernilai '1' berarti peta konjugasi tersebut telah dikonjugasi dan perlu dikonjugasi kembali untuk mendapatkan informasi yang sebenarnya. Sebaliknya, apabila nilai bit tersebut bernilai '0' berarti peta konjugasi telah berisi informasi sebenarnya. Bentuk kembali N gambar biner tersebut menjadi

pesan rahasia dengan memperhatikan nilai bit pada peta konjugasi yang bersesuaian untuk masing-masing gambar biner. Jika nilai bit pada peta konjugasi bernilai '1' berarti gambar biner yang bersesuaian telah dikonjugasi.

### 3.3. Implementasi

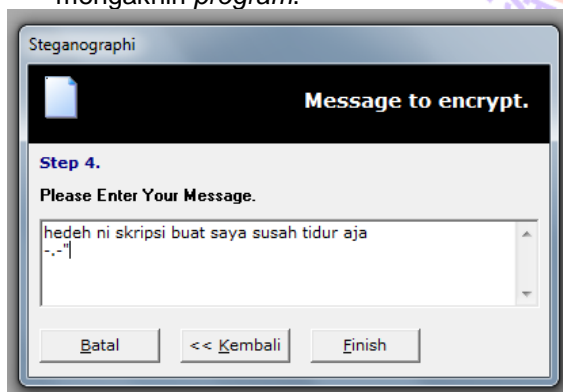
Pada Tampilan *Menu Encrypt/ Decrypt* berfungsi untuk pemilihan penyisipan pesan teks pada *Citra* atau mengekstrakan *Citra* yang telah disisipi teks. Untuk langkah awal pilih *option create (encrypt) a message*, seperti pada gambar 2.



Gambar 1 Menu Penyisipan Pesan dan Ekstrak Pesan

Pada *menu* inilah kita mengisi teks yang akan disisipi pada *file audio*. Pada *menu* ini ada 4 *option* tombol yaitu :

1. *Browser* : Berfungsi untuk memilih *file* yang berekstensi *txt*.
2. *Cancel* : Berfungsi untuk menunda pengisian *plaintext*.
3. *Back* : Berfungsi untuk kembali ke *menu* sebelumnya.
4. *Finish* : Berfungsi untuk mengakhiri *program*.



Gambar 2. Menu Pengisian Plaintext

### KESIMPULAN

Berdasarkan hasil perancangan dari perangkat lunak *steganografi citra digital* dengan *metode BPCS* ini, penulis dapat menarik kesimpulan sebagai berikut :

1. Proses perbaikan citra yang disisipi dengan gambar awal sebelum disisipi

pesan teks tidak mengalami perubahan bentuk, sehingga secara kasat mata tidak dapat diketahui apakah ada pesan di dalam gambar tersebut.

2. Proses perancangan sebuah aplikasi yang dapat menyisipi sebuah pesan terhadap *Citra digital* dengan cara menggunakan *Tehnik Steganography* yaitu, dengan metode *Bit-Plane Complexity Segmentation (BPCS)*.

### DAFTAR PUSTAKA

- [1] D. Ariyus, "Kriptografi keamanan data dan komunikasi," *Yogyakarta Graha Ilmu*, 2006.
- [2] C. Paper, A. Solichin, and U. Budi, "Implementasi Steganografi Dengan Metode Bit Plane Complexity Segmentation Untuk Menyembunyikan," no. March, pp. 2–3, 2016.
- [3] P. B. N. Simangunsong, "Peningkatan Kualitas Citra Pada Studio Photography Dengan Menggunakan Metode Gaussian Filter," *J. Tek. Inform. UNIKA St. Thomas*, vol. 3, no. 1, pp. 59–63, 2018.
- [4] T. Limbong and P. D. P. Silitonga, "Testing the Classic Caesar Cipher Cryptography using of Matlab," *Int. J. Eng. Res. Technol.*, vol. 6, no. 2, pp. 175–178, 2017.
- [5] R. Munir, "Kriptografi," *Inform. Bandung*, 2006.
- [6] T. Sutojo, E. Mulyanto, V. Suhartono, and O. K. I. D. W. I. NURHAYATI, "Teori Pengolahan Citra Digital."
- [7] T. Limbong *et al.*, "The implementation of computer based instruction model on Gost Algorithm Cryptography Learning," in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 420, no. 1, p. 12094.