BAB II

TELAAH LITERATUR

2.1 Rekayasa Perangkat Lunak

Fritz Bauer pada konferensi seminal mendefinisikan rekayasa piranti lunak sebagai "*the establishment and use of sound engineering principles in order to obtain economically software that is reliable dan works efficiently on real machines*".

Dalam buku klasik "*How to Solve It*" yang ditulis sebelum komputer modern ada, Goerge Polya menggambarkan inti dari *problem solving*, yang juga merupakan inti dari praktik rekayasa piranti lunak:

1. Mengerti masalahnya (komunikasi dan analisis).
2. Rancang solusi (memodelkan dan mendesain piranti lunak).
3. Jalankan rancangan yang telah dibuat (buat kode).
4. Teliti hasil untuk akurasi (uji coba dan *quality assurance*).

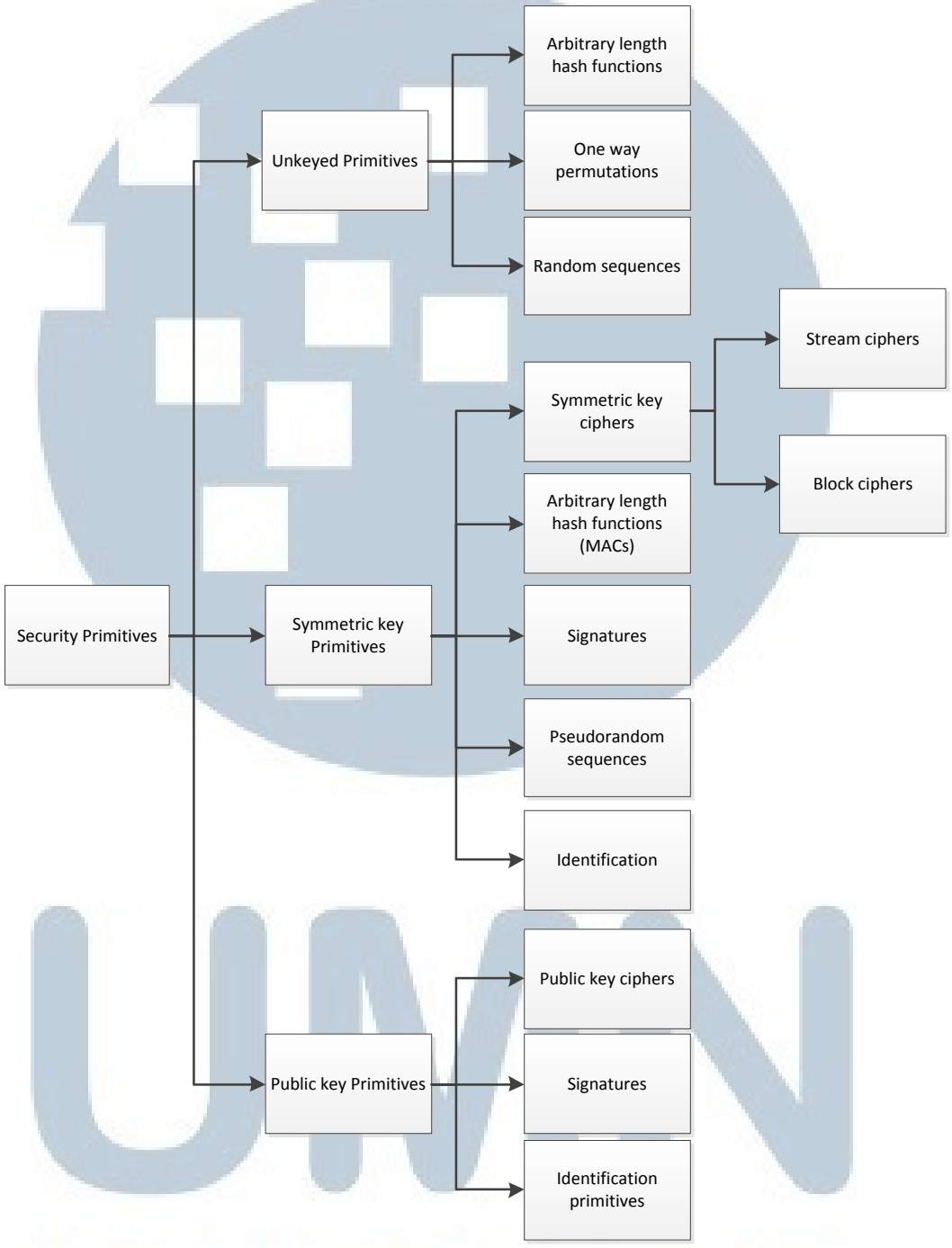
Pressman (1997, p.78) mengungkapkan untuk menyelesaikan suatu masalah pembangunan perangkat lunak, pelaku harus menggabungkan strategi pengembangan yang melingkupi lapisan proses, metode, dan alat-alat bantu. Startegi ini sering diacukan sebagai model proses atau paradigma rekayasa perangkat lunak. Model proses yang dipilih didasarkan pada sifat aplikasi dan proyeknya, metode dan alat-alat bantu yang dipakai, serta kontrol dan penyampaian yang dibutuhkan.

2.2 Kriptografi

Menurut Flourensia Sapty Rahayu (2005, p.4), Kriptografi merupakan seni dan ilmu menyembunyikan informasi dari penerima yang tidak berhak. Kata cryptography berasal dari kata Yunani *kryptos* (tersembunyi) dan *graphein* (menulis). Ada beberapa dasar tool kriptografi (primitif) yang digunakan untuk mendukung keamanan informasi. Gambar 2.1 menunjukkan daftar primitif yang dimaksud dan hubungan antar primitif.

2.2.1 Advanced Encryption Standard

Merupakan algoritma kriptografi yang dapat digunakan untuk mengamankan data. Algoritma AES adalah blok *ciphertext* simetrik yang dapat mengenkripsi (*enchiper*) dan dekripsi (*decipher*) informasi. Enkripsi merubah data menjadi tidak dapat lagi dibaca disebut *ciphertext;* sebaliknya dekripsi adalah merubah *ciphertext* data menjadi bentuk semula yang dikenal sebagai *plaintext.* Algoritma AES menggunakan kunci kriptografi 128, 192, atau 256 bits untuk mengenkripsi dan mendekripsi data pada blok 128 bits (Yoki Ariyana, http://www.p4tkipa.org/data/aes.pdf).



Gambar 2.1 Taksonomi Primitif Kriptografi

Sumber: Flourensia Sapty Rahayu, 2005

2.2.2 Nonlinear Chaotic Algorithm

Menurut Haojiang Gao et al. (2005), Nonlinear Chaotic Algorithm merupakan algoritma yang diajukan untuk mengatasi kelemahan dari *chaotic algorithm* yang menggunakan *logistic map*. *Nonlinear chaotic algorithm* (NCA) *map* menggunakan *power function* $(1 - x)^{\beta}$ dan fungsi tangent daripada menggunakan fungsi linear. NCA *map* didefinisikan sebagai

$$x_{n+1} = (1 - \beta^{-4}) \cdot \text{ctg}(\alpha / (1 + \beta)) \cdot (1 + (1/\beta))^{\beta} \cdot \text{tg}(\alpha x_n) \cdot (1 - x_n)^{\beta} \quad (2.1)$$

dimana xn ϵ (0,1), α ϵ (1 , 1,4], β ϵ [5 , 43] atau xn ϵ (0,1), α ϵ (1,4 , 1,5], β ϵ [9 , 38] atau xn ϵ (0,1), α ϵ (1,5 , 1,57], β ϵ [3 , 15].

Menurut penelitian Haojiang Gao, dkk. algoritma NCA ini memiliki kelebihan *key space* yang besar dan keamanan yang tinggi, sementara mempertahankan efisiensi. Algoritma ini secara khusus cocok digunakan untuk enkripsi gambar di internet dan aplikasi transmisi.

Algoritma ini mampu bertahan dari serangan *brute-force.* Andaikan saja α, β, dan x0 masing-masing memiliki nilai berupa 15 *digit floating-point*, maka untuk melakukan *brute-force* akan memerlukan kombinasi sebanyak 10^{45}. Kombinasi tersebut telah melebihi 56-bit-DES yang sudah dianggap aman. Oleh karena itu, dapat dikatakan algoritma NCA juga merupakan algoritma yang aman dari serangan *brute-force* (Haojiang Gao et al., 2005).

Algoritma ini juga aman dari serangan *chosen/known-plain-text*. Interval antar *chaos sequence* menyebabkan hubungan x_n dan x_{n+1} menjadi sangat kompleks. Hal ini membuat serangan menggunakan *chosen/known-plain-text*

menjadi sangat tidak efisien dan percuma secara praktik (Haojiang Gao et al., 2005).

2.3 *Deflate Algorithm*

- Menurut dokumentasi *Library* MSDN, algoritma *deflate* merupakan standar algoritma industri untuk kompresi dan dekompresi *file lossless.* Algoritma ini menggunakan kombinasi algoritma Lempel-Ziv 1977 (LZ77) dan *Hufman coding* dan pertama kali didefinisikan oleh Phil Katz.

Algoritma LZ77 mencari *window* dengan kecocokan terpanjang dari bagian sekarang pada *stream input* ke bagian yang telah dilewati pada *stream input* (Arkadi Kagan).

Algoritma *Huffman coding* dapat dijelaskan sebagai berikut (nilkesh patra dan sila siba sankar, 2007).

1. Mulailah dengan hutan pohon, satu untuk setiap pesan. Setiap pohon berisi satu *vertex* dengan *weight* $W_I = P_I$
2. Ulangi sampai sampai hanya 1 pohon tersisa
   a. Pilih 2 pohon dengan *root* yang memiliki *weight* terendah (W_1 dan W_2)
   b. Gabungkan kedua pohon tersebut menjadi sebuah pohon dengan menambahkan *root* baru dengan *weight* $W_1 + W_2 = C$ dan buat kedua pohon tersebut sebagai anaknya.

2.4 Steganografi

Menurut Eiji Kawaguchi dan Richard O. Eason, Steganografi adalah teknik menyembunyikan bukti nyata hasil perubahan data.

Gary C. Kessler (2001) mengungkapkan steganografi sebagai ilmu menyembunyikan informasi. Jika tujuan dari kriptografi adalah membuat data tidak dapat dibaca oleh pihak ketiga, maka tujuan dari steganografi adalah menyembunyikan data dari pihak ketiga.

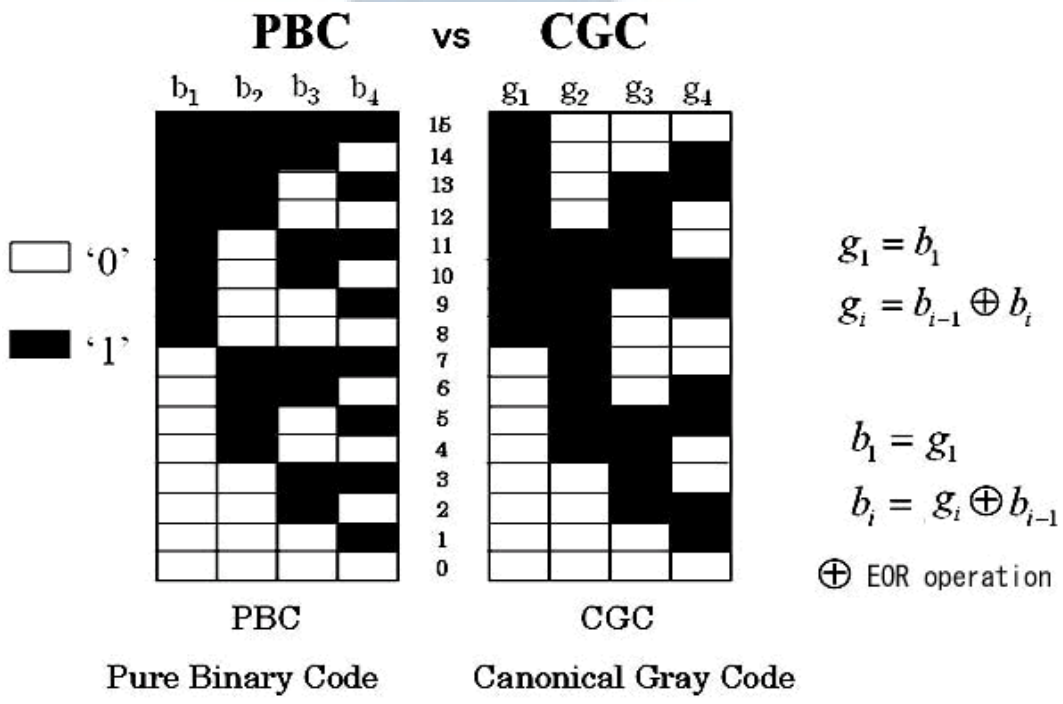
Ada banyak metode steganografi yang dikenal, mulai dari tinta tembus pandang hingga *spread spectrum* dalam komunikasi radio. Dengan komputer dan jaringan, ada banyak cara lain untuk menyembunyikan data seperti menyimpan pesan rahasia dalam halaman web, menyembunyikan data dalam "*plain sight*" (menyembunyikan file pada direktori c:/winnt/system32), serta *Null ciphers* (misalkan menggunakan huruf pertama dari setiap kata untuk membentuk pesan rahasia dalam pesan lainnya) (Gary C. Kessler, 2001).

Steganografi dewasa ini jauh lebih maju dari contoh yang telah disebutkan di atas, memberikan kemampuan pada pengguna untuk menyembunyikan informasi yang besar dalam gambar atau pun suara. Bentuk steganografi seperti ini sering kali digunakan bersamaan dengan kriptografi sehingga informasi dapat dilindungi secara ganda. Informasi pertama dienkrip baru disembunyikan sehingga penyerang harus menemukan dahulu informasi tersebut (tugas yang tidak mudah untuk dilakukan), baru kemudian mendekripsi informasi tersebut (Gary C. Kessler, 2001).

2.4.1 *Bit-Plane Complexity Segmentation* (BPCS)

Menurut Eiji Kawaguchi dan Richard O. Eason, BPCS merupakan metode steganografi untuk menyembunyikan informasi rahasia dalam gambar berwarna. Ini bukan berdasarkan teknik pemrograman, tetapi berdasarkan karakter sistem penglihatan manusia. Kapasitas penyembunyian bisa sebesar 50% dari gambar asli.

Pada metode ini, gambar digital dikategorikan sebagai gambar *binary* atau *multi-valued* tanpa memerdulikan warna aslinya. n-bit gambar dapat didekomposisikan menjadi sebuah set gambar binary dengan operasi *bit-slicing*. Karena itu, analisis gambar biner sangat penting pada setiap pemrosesan gambar digital. *Bit-slicing* bukan selalu merupakan yang terbaik dalam sistem *Pure-Binary Coding* (PBC) tetapi dalam beberapa kasus, sistem *Canonical Gray Coding* (CGC) jauh lebih baik. *Pure-Binary Coding* merupakan metode *encoding* yang merepresentasikan angka dalam *binary sequence*-nya masing-masing. Dalam gambar tiap nilai warna direpresentasikan dalam 1 byte. Dimana tiap bit memiliki nilai 2x dari bit di sebelah kanannya. (misal: 1 -> 00000001, 129 -> 10000001, 255 -> 11111111). *Canonical Gray Coding* merupakan sistem biner numeral dimana 2 nilai yang berdampingan hanya memiliki perbedaan 1 bit pada representasi binernya (Eiji Kawaguchi dan Richard O. Eason, 1998).



Gambar 2.2 Perbandingan PBC dan CGC
Sumber: http://www.datahide.com/BPCSe/pbc-vs-cgc-e.html

Metode BPCS menggunakan area yang kompleks dalam gambar untuk menyembunyikan data. Tidak ada definisi standar mengenai ukuran kompleksitas gambar. Salah satu pengukuran kompleksitas gambar adalah "*black-and-white border image complexity*". Pada pengukuran ini kompleksitas diukur pada banyaknya perubahan warna hitam dan putih pada gambar. Nilai kompleksitas pada suatu gambar biner dapat didefinisikan sebagai berikut.

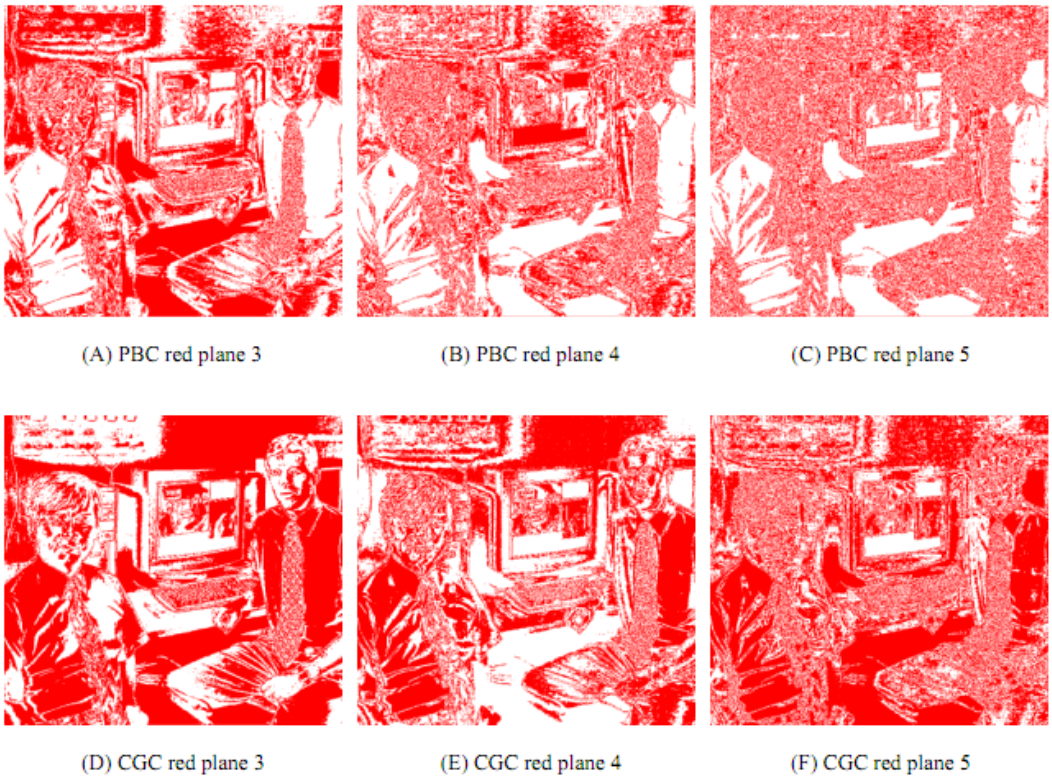
$$\alpha = \frac{k}{\text{max perubahan warna}} \qquad (2.2)$$

Dari definisi di atas dapat disimpulkan bahwa nilai kompleksitas (α) berjarak dari 0 sampai 1 (Eiji Kawaguchi dan Richard O. Eason, 1998).

Penggunaan area kompleks sebagai area penyembunyian pesan rahasia memperlihatkan pentingnya penggunaan sistem CGC pada metode ini. Gambar 2.3 memperlihatkan perbedaan *bit-plane* pada sistem CGC dan PBC. Membandingkan kedua sistem tersebut, terlihat bahwa sebenarnya sistem PBC memberikan area yang lebih banyak untuk menyembunyikan informasi. Hal ini sesuai dengan tujuan metode BPCS yang ingin mencapai kapasitas penyembunyian sebesar-besarnya dari suatu gambar. Namun, ternyata pada sistem PBC, *bit-plane* pada tingkat tinggi yang tampaknya dapat disubtitusi dengan pesan ternyata memiliki warna yang cukup merata (seperti tembok di *background* gambar) (Eiji Kawaguchi dan Richard O. Eason, 1998).

Kejadian di atas terjadi karena efek "*Hamming Cliffs*" yang muncul pada sistem PBC karena sedikit perubahan pada warna akan memberikan perubahan yang signifikan pada nilai bit. Apabila sampai terjadi subtitusi pada area seperti ini, maka akan terjadi perubahan warna yang cukup jelas Khaire, (Shrikant S. dan Sanjay L. Nalbalwar, 2010).

Sistem CGC tidak mengalami hal yang sama kerena setiap karakteristiknya yang hanya berbeda 1 bit tiap perubahan nilai warna. Walaupun sistem ini membatasi area yang dapat digunakan utuk penyimpanan pesan, area yang tidak seharusnya disubtitusi tidak akan disubtitusi dengan pesan rahasia (Shrikant S. dan Sanjay L. Nalbalwar, 2010).

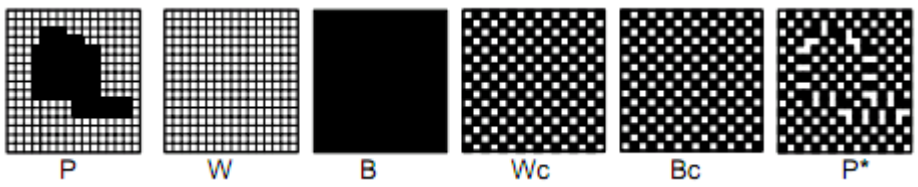


Gambar 2.3 Perbandingan *plane* pada Sistem PBC dan CGC
Sumber: Eiji Kawaguchi dan Richard O. Eason, 1998

Gambar informatif itu sederhana sedangkan gambar yang *noise-like* itu kompleks. Hal ini hanya berlaku pada gambar biner yang merupakan bagian dari gambar natural. Suatu gambar biner yang tidak cukup kompleks dapat dibuat menjadi kompleks dengan mengkonjugasikannya (Arya Widyanarko).

Apabila terdapat gambar biner P yang memiliki besar 2^N x 2^N. Anggap P merupakan gambar yang terdiri dari warna hitam dan putih dimana hitam merupakan warna *foreground* dan putih adalah *background.* W dan B secara berurutan merupakan gambar yang terdiri dari warna putih secara keseluruhan dan hitam secara keseluruhan. Terdapat juga Wc dan Bc, dimana Wc memiliki warna putih pada posisi paling atas kiri dan BC adalah komplemen dari Wc. Pixel hitam

dan putih dianggap secara berurutan sebagai nilai “1” dan “0” (Eiji Kawaguchi dan Richard O. Eason, 1998).



Gambar 2.4 Ilustrasi Gambar-Gambar Biner (N=4)
Sumber: Eiji Kawaguchi dan Richard O. Eason, 1998

P dapat dijelaskan sebagai berikut, pixel di area *foreground* memiliki pola B sedangkan pada area *background* memiliki pola W. P* didefinisikan sebagai hasil konjugasi P sebagai berikut (Eiji Kawaguchi dan Richard O. Eason, 1998)..

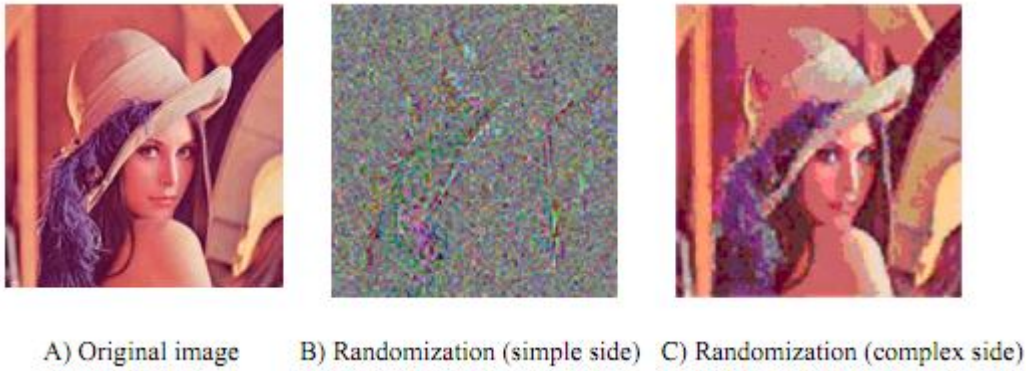
1. Memiliki bentuk area *foreground* yang sama dengan P.
2. Memiliki area *foreground* dengan pola Bc.
3. Memiliki area *background* dengan pola Wc.

Properti di bawah ini bernilai benar bagi pada operasi konjugasi.

1. $P^* = P \wedge Wc$ (2,3)
2. $(P^*)^* = P$ (2,4)
3. $P^* \neq P$ (2,5)
4. $\alpha(P^*) = 1 - \alpha(P)$ (2,6)

Untuk mengkategorikan gambar biner yang informatif dan *noise-like*, digunakan ukuran kompleksitas α. Menurut hasil penelitian yang dilakukan oleh

Eiji Kawaguchi dan Richard O. Eason, nilai α yang menjadi pembatas antara gambar yang informatif dan *noise-like* adalah 0,5 – 8σ, dimana σ adalah standar deviasi dari percobaan yang juga dilakukan oleh mereka untuk mendapatkan nilai rata-rata dari α pada pola biner 2^{64}. Percobaan yang dilakukan untuk mendapatkan nilai tersebut adalah dengan mensubtitusi semua area *noise-like* yang dikategorikan dengan nilai tersebut dengan pola biner acak.



A) Original image B) Randomization (simple side) C) Randomization (complex side)

Gambar 2.5 Hasil Subtitusi Area *Noise-Like* dengan Pola Acak (α = 0,5 – 8σ)
Sumber: Eiji Kawaguchi dan Richard O. Eason, 1998

Secara umum steganografi dengan metode BPCS secara umum memiliki langkah sebagai berikut (Arya Widyanarko).

1. Mengubah cover image dari sistem PBC menjadi sistem CGC. Sebelumnya, gambar tersebut dipotong terlebih dahulu menjadi bit-plane. Setiap bitplane mewakili bit dari setiap piksel.
2. Segmentasi setiap bit-plane pada cover image menjadi informatif dan *noise like* region dengan menggunakan nilai batas/threshold (α0).
3. Bagi setiap byte pada data rahasia menjadi blok-blok.

4. Jika blok (S) tidak lebih kompleks dibandingkan dengan nilai batas, maka lakukan konjugasi terhadap S untuk mendapatkan S* yang lebih kompleks.
5. Sisipkan setiap blok data rahasia ke bit-plane yang merupakan noise-like region. Kemudian simpan data konjugasi pada “conjugation map”.
6. Sisipkan juga pemetaan konjugasi yang telah dibuat.
7. Ubah stego-image dari sistem CGC menjadi sistem PBC.