Jurnal Rekayasa Elektrika

VOLUME 11 NOMOR 3 APRIL 2015

JRE	Vol. 11	No. 3	Hal 79–122	Banda Aceh, April 2015	ISSN. 1412-4785 e-ISSN. 2252-620X

Klasifikasi Bit-Plane Noise untuk Menyisipkan Pesan pada Teknik Steganografi BPCS Menggunakan Fuzzy Inference Sistem Mamdani

Rahmad Hidayat
Politeknik Negeri Lhokseumawe
Jl. Banda Aceh-Medan Km. 280, Buketrata 24301
e-mail: rahmad_anwar@yahoo.com

***Abstrak*—Teknik steganografi *Bit-Plane Complexity Segmentation* (BPCS) merupakan salah satu teknik steganografi yang cukup baru. Salah satu proses penting dalam Teknik Steganografi BPCS adalah proses penghitungan nilai kompleksitas suatu bit-plane. Nilai kompleksitas dihitung dengan melihat seberapa banyak pergantian bit yang terdapat dalam sebuah bit-plane. Jika bit-plane tersebut memiliki nilai kompleksitas yang tinggi, maka *bit-plane* tersebut dikategorikan sebagai *bit-plane noise* yang tidak mengandung informasi yang berharga pada gambar. Klasifikasi *bit-plane* yang menggunakan himpunan *cripst* (*noise* dan tidak) tersebut tidak adil, dimana sebuah perbedaaan nilai yang sedikit saja akan mengubah secara signifikan status dari *bit-plane* tersebut. Tujuan penelitian adalah untuk menerapkan prinsip himpunan *fuzzy* untuk mengklasifikasikan bit-plane menjadi kedalam tiga buah himpunan yaitu informatif, informatif sebagian, dan *noise region*. Klasifikasi *bit-plane* kedalam himpunan *fuzzy* terebut diharapkan dapat menggolongkan bit-plane secara lebih objektif dan pada akhirnya daya tampung gambar terhadap pesan dapat ditingkatkan dengan dengan menggunakan inferensi *fuzzy* Mamdani untuk mengambil keputusan bit-plane mana yang akan digantikan dengan pesan berdasarkan klasfikasi *bit-plane* yang tersedia dan ukuran pesan yang akan disisipkan. Penelitian ini menghasilkan teknik steganografi BPCS yang mampu menyisipkan pesan pada bit-pane dengan lebih tepat sehingga kualitas gambar kontainer lebih baik. Hal ini dapat dilihat dari nilai PNSR gambar asli dan *stego-image* tidak terlalu berbeda.**

Kata kunci: ***: steganografi, bit-plane, BPCS, batas ambang, stego-image, fuzzy***

***Abstract*—Bit-Plane Complexity Segmentation (BPCS) is a fairly new steganography technique. The most important process in BPCS is the calculation of complexity value of a bit-plane. The bit-plane complexity is calculated by looking at the amount of bit changes contained in a bit-plane. If a bit-plane has a high complexity, the bi-plane is categorized as a noise bit-plane that does not contain valuable information on the image. Classification of the bit-plane using the set cripst set (noise/not) is not fair, where a little difference of the value will significantly change the status of the bit-plane. The purpose of this study is to apply the principles of fuzzy sets to classify the bit-plane into three sets that are informative, partly informative, and the noise region. Classification of the bit-plane into a fuzzy set is expected to classify the bit-plane in a more objective approach and ultimately message capacity of the images can be improved by using the Mamdani fuzzy inference to take decisions which bit-plane will be replaced with a message based on the classification of bit-plane and the size of the message that will be inserted. This research is able to increase the capability of BPCS steganography techniques to insert a message in bit-pane with more precise so that the container image quality would be better. It can be seen that the PSNR value of original image and stego-image is only slightly different.**

Keywords: ***steganography, bit-plane, BPCS, threshold, stego-image, fuzzy***

I. PENDAHULUAN

Gambar merupakan salah satu media yang biasa digunakan sebagai media penyimpan pesan dalam teknik steganografi. Steganografi merupakan metode komunikasi untuk menyembunyikan pesan rahasia (teks atau gambar) di dalam berkas-berkas lain yang mengandung teks, *image*, bahkan audio tanpa menunjukkan ciri-ciri perubahan yang nyata atau terlihat dalam kualitas dan struktur dari berkas semula.

Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Dalam prakteknya, kebanyakan pesan disembunyikan dengan membuat perubahan tipis terhadap data digital lain yang isinya tidak akan menarik perhatian dari penyerang potensial, sebagai contoh sebuah gambar yang terlihat tidak berbahaya.

Salah satu proses terpenting dalam teknik steganografi adalah proses pendeteksian daerah yang yang akan didisipkan pesan dengan tepat. Pada teknik steganografi LSB bagian bagian yang akan disisipkan pesan hanya bit terakhir sehingga daya tampung pesan menjadi kecil [4]. Pada penelitian ini gambar dipecah menjadi *bit-plane*. *Bit-plane* tersebut kemudian dihitung nilai

ISSN. 1412-4785; e-ISSN. 2252-620X
DOI: 10.17529/jre.v11i3.2238

kompleksitasnya menggunakan variabel variabel β. Penghitungan kompleksitas dengan varibel β ditujukan untuk mengevaluasi ketidak seragaman pola dalam *bit-plane*. Varibel β juga dapat menghindari penyisipan pesan pada pola yang periodik sehingga bagian yang informatif dalam kontainer dapat dipertahankan.

Setelah nilai kompleksitas suatu *bit-plane* didapat maka nilai tersebut akan dipetakan ke variabel input *fuzzy*. Selain nilai kompleksitas *bit-plane*, variabel lain yang digunakan sebagai input antara lain berupa ratio pesan terhadap gambar. Dengan cara ini maka diharapkan nilai *threshold* yang dihasilkan dapat optimal dimana seluruh pesan dapat ditampung kedalam gambar tanpa membuat kualitas stego-image terlalu buruk [1].

II. Studi Pustaka

Kawaguchi dan Eason (1998) dalam penelitiannya memperkenalkan teknik steganografi *Bit-Plane Complexity Segmentation Steganography* (BPCS) yang merupakan teknik steganografi baru. Teknik ini mengganti *bit-planes (array bit* dengan ukuran 8x8) yang terlihat sebagai *noise* dalam kontainer sehingga data yang dapat disisipkan dapat meningkat sampai dengan 50% dari ukuran asli kontainernya [1]. Hirohisa (2002) melakukan penelitian dengan judul "A Data Embedding Method Using BPCS Principle With New Complexity Measures". Pada penelitian tersebut disimpulkan bahwa kelemahan dalam pengukuran kompleksitas *bit-plane* jika menggunakan variabel α seperti yang terdapat pada teknik BPCS standar adalah jika distribusi piksel hitam dan putih dalam sebuah blok memiliki urutan yang periodik maka blok tersebut tidak dapat digunakan untuk menyisipkan data . Oleh karena itu dalam penelitian tersebut diperkenalkan variabel β. Variabel β ini ditujukan untuk mengevaluasi distribusi piksel hitam dan putih yang tidak seragam yang terdapat dalam sebuah blok [6].

Murguia,et al. (2007) melakukan penelitian dengan judul "A Fuzzy Approach on Image Complexity Measure". Pada penelitian tersebut, disebutkan bahwa pengukuran kompleksitas *fuzzy* dapat digunakan sebagai kriteria untuk mendeteksi batas gambar. Kriteria *fuzzy* yang digunakan dalam penelitian tersebut antara lain *Little Complex* (LC), *More or Less Complex* (ML), dan *Very Complex* (VM). Kategori ini didapatkan dari menganalisa sekumpulan gambar *Fuzzy C-means* [9]. Noda, Hideki., et al melakukan penelitian kesesuain teknik steganografi BPCS dengan video yang dikompresi wavelet. Pada penelitian tersebut diketahui bahwa penggunaan kontainer berupa video yang dikompresi wavelet pada teknik steganografi BPCS mampu menghasilkan kapasitas penyimpanan yang besar [10].

Perbedaan penelitian ini dengan penelitian terdahulu yaitu pada penelitian ini *bit-plane* terlebih dahulu di klasifikasikan menggunakan inferensi *fuzzy* mamdani dan kemudian penentuan nilai *threshold*-nya menggunakan penghitungan kompleksitas *bit-plane* menggunakan variabel β.

III. Metode

Pada penelitian ini, langkah-langkah penyisipan pesan ke dalam gambar mengikuti langkah-langkah seperti yang terdapat dalam teknik steganografi BPCS. Hanya bedanya pada penelitian ini, penetuan nilai kompleksitas gambar biner menggunakan variable β. Selain itu juga penentuan nilai threshold yang di pakai menggunakan inferensi fuzzy mamdani. Berikut adalah langkah-langkah yang dilakukan pada penelitian ini:

1. Mengubah *cover image* dari sistem PBC menjadi sistem CGC. Sebelumnya, gambar tersebut di-*slice* terlebih dahulu menjadi *bit-plane* dengan ukuran 8x8 bit. Setiap *bit-plane* mewakili bit dari setiap piksel. Gambar 1 menunjukkan proses pemotongan gambar menjadi kumpulan *bit-plane*.
   Jika dilihat dari Gambar 1, maka *bit-plane* yang disusun oleh bit 7 merupakan *least significant bit*. Pada BPCS, semua *bit-plane* memiliki kemungkinan untuk diganti, tidak hanya *least significant bit*. *Bit-plane* yang berupa gambar biner tersebut dikonversi dari PBC menjadi CGC. Gambar 2 menunjukkan perbedaan pada PBC dan CGC. Berikut adalah rumus konversi PBC ke CGC dan sebaliknya.

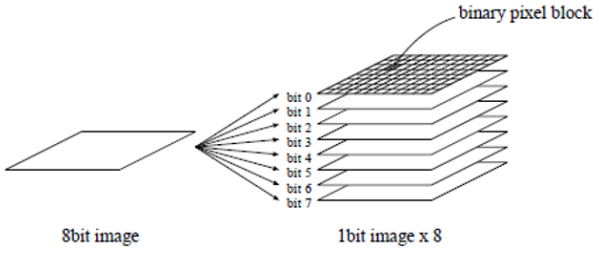
$$g_1 = b_1 \quad (1)$$

$$g_i = b_{i-1} \oplus b_i \quad (2)$$

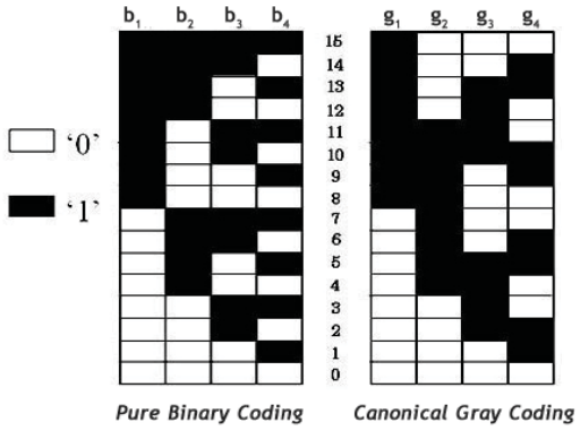
$$b_1 = g_1 \quad (3)$$

$$b_i = g_i - 1 \oplus b_{i-1} \quad (4)$$

   dengan, g1 adalah kolom pertama hasil konversi ke CGC; gi sebagai kolom ke-i hasil konversi ke CGC; b1 sama dengan kolom pertama hasil konversi ke PBC; dan bi adalah kolom ke-i hasil konversi ke PBC.
2. Segmentasi setiap *bit-plane* pada *cover image* menjadi *informative* dan *noise like region* dengan menggunakan nilai batas/*threshold* (α_0). Nilai umum dari α_0 adalah 0,3.
3. Bagi setiap *byte* pada data rahasia menjadi blok-blok.
4. Jika blok(*S*) tidak lebih kompleks dibandingkan dengan nilai *threshold*, 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* (atau gantikan semua bit pada *noise-like region*). Jika blok *S* dikonjugasi, maka simpan data pada "*conjugation map*".
6. Sisipkan juga pemetaan konjugasi yang telah dibuat



Gambar 1. Proses pemotongan gambar menjadi kumpulan *bit-plane*

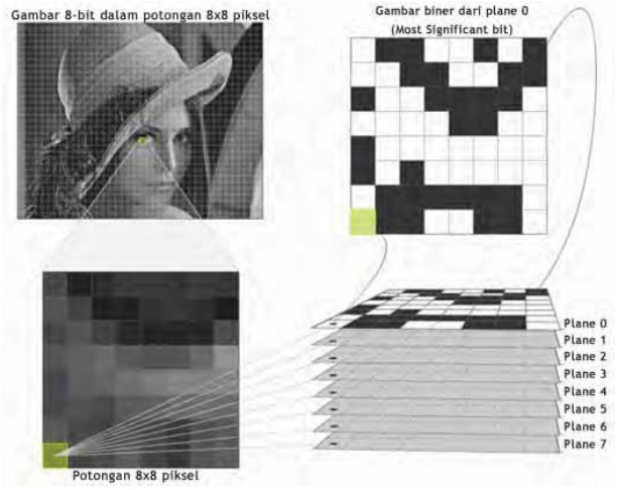


Gambar 2. Gambar Biner dengan PBC dan CGC

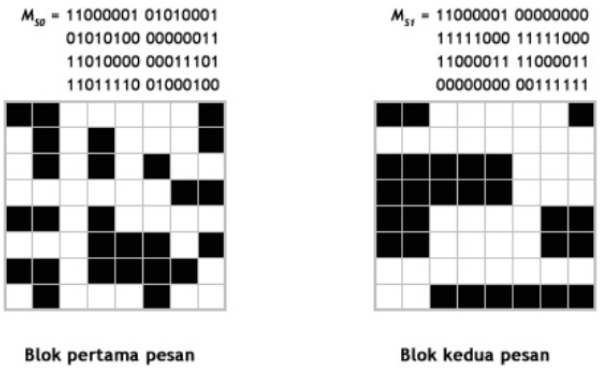
7. Ubah *stego-image* dari sistem CGC menjadi sistem PBC.

Proses ekstraksi data rahasia dapat dilakukan dengan menerapkan langkah-langkah penyisipan secara terbalik. Sebagai contoh, sebuah dokumen gambar akan disisipi sebuah pesan rahasia *Ms*. Pertama-tama piksel pada gambar tersebut (*cover image*) dibagi menjadi segmen-segmen gambar biner seperti ditunjukkan pada Gambar 3. Kemudian pesan rahasia dibagi menjadi blok yang masing-masing berukuran 64 bit, dan direpresentasikan pada matriks berukuran 8x8.

Penghitungan nilai kompleksitas sebuah *bit-plane* dilakukan dengan menghitung jumlah pergantian warna hitam-putih pada *bit-plane*. Jumlah maksimum perubahan warna pada gambar biner dengan ukuran 8x8 adalah 112 kali, sehingga nilai k=47 dan n = 112. Melalui persamaan 5 didapatkan nilai kompleksitas dari *bit-plane 0* tersebut, yaitu α = 0,42. Dengan menggunakan nilai *threshold* α_0 = 0,3 maka *bit-plane 0* dikategorikan sebagai *noise-like region* sehingga dapat dilakukan penyisipan didalamnya. Jika $\alpha < \alpha_0$, maka tidak dilakukan penyisipan karena segmen tersebut merupakan *informative region*. Selanjutnya bit pesan rahasia dibagi menjadi segmen-segmen yang masing-masing berukuran 64 bit. Jika bit pesan rahasia



Gambar 3. Proses pengubahan gambar menjadi segmen-segmen Bit-Plane



Gambar 4. Representasi blok pesan dalam gambar Biner

tersebut adalah M_S maka Blok pertama pesan rahasia adalah M_{S0} dan blok berikutnya adalah $M_{S1.}$

M_s= 10000010101000101010100000000011110100
00000111011101111001000101100000100000
00011111000111110001100001111000011000
00000 00111111

M_{s0}= 10000010101000101010100000000011110100
00000111011101111001000100

M_{s1}= 10000010000000001111100011111000110000 1
11100001100000000000 011 1111

Representasi blok pesan dalam gambar biner dapat dilihat pada Gambar 4. Blok pesan M_{S0} akan disisipkan pada blok gambar yaitu *bit-plane 0* (karena tergolong *noise-like region*), dan blok M_{S1} akan disisipkan pada *bit-plane* berikutnya yang tergolong *noise-like region* juga.

Sebelum melakukan penyisipan, gambar biner yang merupakan representasi blok pesan tersebut dihitung nilai kompleksitasnya terlebih dahulu. Pada blok pesan pertama (M_{S0}), jumlah perubahan warna adalah 54 kali, sehingga dengan persamaan 5 diperoleh α_{MS0} = 0,48. Karena blok pesan ini memiliki kompleksitas $\alpha_{MS0} > \alpha_0$, maka blok *bit-plane* pada gambar diganti oleh 64 bit pesan ini. Pada blok kedua pesan rahasia, jumlah perubahan warna adalah 32, sehingga didapatkan nilai α_{MS1} = 0,29. Nilai kompleksitas $\alpha_{MS1} < \alpha_0$ menunjukkan bahwa blok kedua pesan tidak cukup kompleks untuk disisipkan, karena itu blok pesan tersebut harus dikonjugasi terlebih dahulu. Hasil konjugasi, yaitu α_{MS1*} akan memiliki kompleksitas 0,71 menurut persamaan 5 Hasil konjugasi inilah yang kemudian disisipkan pada *noise-like region* pada gambar digital. Saat proses ekstrasi pesan, yang perlu dilakukan hanyalah mengambil segmen bit yang memiliki kompleksitas diatas *threshold*. Jika nilai kompleksitas segmen tersebut lebih besar dari *threshold*, maka segmen tersebut merupakan bagian dari pesan rahasia. Tabel konjugasi yang disisipkan juga dibaca untuk melihat proses konjugasi yang perlu dilakukan pada tiap blok pesan.

A. Kompleksitas Gambar Biner dengan Variabel β

Pengukuran kompleksitas dengan menggunakan ***α*** mudah dimengerti dan biasanya dapat berkerja dengan

baik dalam mengklasifikasikan blok yang komplek dan blok yang sederhana. Bagaimanapun juga pengukuran ini tidak selalu dapat digunakan. Ketika sebuah blok memiliki banyak piksel hitam dan putih, maka tidak secara otomatis blok tersebut komplek.

Sebagai contoh motif papan catur yang terdapat pada Gambar 5 (a) memiliki nilai α =1, yang mana merupakan nilai α maksimum. Blok tersebut memiliki motif yang periodik dan tidak dapat di klasifikasikan sebagai blok yang komplek. Jika blok seperti tersebut diganti dengan blok pesan maka pergantian tersebut akan mengganggu motif dan akan membuat perubahan yang cukup signifikan pada gambar.

Jika distribusi piksel hitam dan putih dalam sebuah blok memiliki periodik tertentu, maka blok tersebut tidak dapat digunakan untuk penyisipan. *Run-length irregularity* merupakan pengukuran kompleksitas baru yang diperkenalkan untuk mengevaluasi distribusi piksel hitam dan putih yang tidak seragam. Jika nilai *run-length irregularity* sebuah blok besar, maka blok tersebut tidak memiliki motif piksel hitam dan putih yang bergantian secara periodik, dan juga blok tersebut tidak mungkin memiliki piksel hitam dan putih secara keseluruhan. *Run-length irregularity* didefenisikan berdasarkan histogram *run-length* dari piksel hitam dan putih pada baris dan kolom sebuah blok. Sebagai contoh blok yang terdapat pada Gambar 6 (a), blok tersebut terdiri dari tiga piksel putih yang berutan yaitu :

1. Sebuah piksel hitam
2. Dua piksel putih yang berurutan
3. Dua piksel hitam yang berurutan

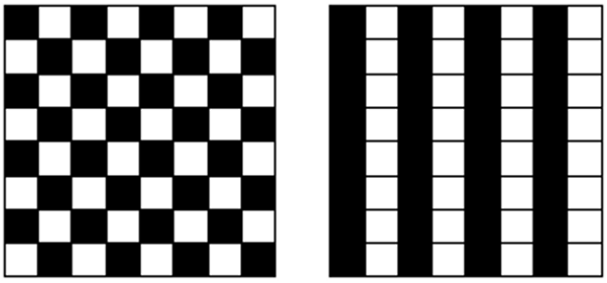
Dari fakta tersebut didapatkan h[1]=1,h[2]=2, dam h[3]=1, Dimana h[i] merupakan frekuensi urutan dari piksel i dalam hitam ataupun putih.

Persamaan h_s berikut ini digunakan untuk menghitung irregularity dari urutan piksel biner:

$$h_s = -\sum_{i=1}^{n} h[\mathrm{i}]\log_2 pi \qquad (6)$$

$$pi = \frac{h[i]}{\sum_{j=1}^{n} h[j]}$$

dimana, h_s adalah nilai *run-length* untuk kolom/baris; h_i sebagai *run-length* dengan jumlah ke-i; dan p_i merupakan nilai perbandingan *run-length* ke-i dengan total *run-length* dalam sebuah baris/kolom.



(a) (b)

Gambar 5. Blok yang tidak komplek

Dimana *n* adalah *run-length* terpanjang yang mungkin. Persamaan h_s ini menghitung ketidaksamaan distribusi *run-length* dalam urutan biner. Jika sebuah urutan memiliki panjang yang sama, maka akan dapat diketahui adanya motif yang berulang dan periodik. Pada kasus tersebut maka hs akan bernilai 0. sebagai contoh, hs bernilai 0 untuk urutan yang ditunjukkan oleh Gambar 6 (a), 6 (b), 6 (c), dan 6 (d). Pada sisi lain, jika jika sebuah urutan piksel memiliki panjang yang bervariasi, maka urutan tersebut memiliki nilai hs yang besar. Nilai h_s dapat dinormalisasi dalam [0,1], h_s yang telah ternormalisasi dinotasikan $\hat{h}_s$. Misalkan sebuah blok dengan ukuran *n x n*. r_i dan c_j merupakan baris ke i dan kolom ke j. Secara berurutan. Maka *run-length* β dari sebuah blok didefenisikan sebagai berikut,

$$\beta = \min\left\{\overline{\hat{H}_s(r)}, \overline{\hat{H}_s(c)}\right\} \qquad (7)$$

dengan, H_s(r) adalah rata-rata nilai *run-length* untuk baris, dan H_s(c) sebgai rata-rata nilai *run-length* untuk kolom.

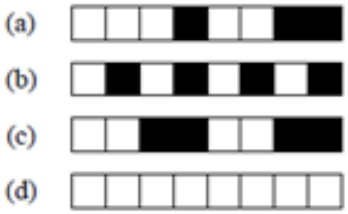
Dari persamaan tersebut dapat dilihat, fungsi yang sama di aplikasikan secara bersama-sama untuk kolom dan baris sebuah blok. Nilai hs dari sebuah blok dihitung untuk setiap kolom dan baris, kemudian diacari nilai rata-ratanya ($\hat{H}_s(r)$ and $\hat{H}_s(c)$) untuk kolom dan baris. Kemudian nilai terkecil dari kedua rata-rata tersebut kemudian akan dipakai sebagai nilai *run-length irregularity* [6]

B. Konjugasi dari Gambar Biner

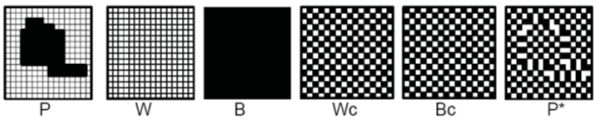
Konjugasi dari suatu gambar biner *P* adalah sebuah gambar biner lainnya yang memiliki nilai kompleksitas sebesar satu dikurangi nilai kompleksitas *P*. Misalkan sebuah gambar hitam-putih P berukuran 8x8 piksel memiliki warna *background* putih dan warna *foreground* hitam. *W* adalah *pattern* dengan semua piksel berwarna putih dan *B* adalah *pattern* dengan semua piksel berwarna hitam. *Wc* dan *Bc* adalah pola papan catur, dengan warna piksel atas-kiri berwarna putih pada *Wc* dan hitam pada *Bc*. *P** adalah konjugasi dari gambar *P* yang ditunjukan pada Gambar 7.

Dari Gambar 7 dapat dilihat bahwa *P* adalah gambar yang memiliki piksel *background* dengan pola *W* dan piksel *foreground* dengan pola *B*. *P** yang merupakan konjugasi dari *P* memiliki spesifikasi sebagai berikut [1]:

1. Memiliki bentuk area *foreground* sama dengan *P*.



Gambar 6. Blok yang periodik



Gambar 7. Contoh Konjugasi dan *binary pattern*

2. Memiliki pola area *foreground* sama dengan pola *Bc*.
3. Memiliki pola area *background* sama dengan pola *Wc*.

Untuk membangun sebuah konjugasi *P** dari sebuah gambar *P*, dapat dilakukan dengan rumus berikut, dimana "⊕"menandakan operasi *exclusive* OR (XOR).

$$P^* = P \oplus W_C \tag{8}$$

Keterangan :
P : *bit-plane* asli
*P**: *bit-plane* hasil konjugasi
W_C: matriks untuk mengkonjugasi/mendekonjugasi *bit-plane,*

Jika **α**(*P*) adalah kompleksitas dari *P*, maka:

$$\alpha(P^*) = 1 - \alpha(P) \tag{9}$$

Keterangan :
α(*P**) : nilai kompleksitas *bit-plane* hasil konjugasi
α(*P*) : nilai kompleksitas *bit-plane* asli

C. Uji Kualitas Stego-Image

Pengujian kualitas *stego-image* juga merupakan pengujian performansi perangkat lunak. Pengujian ini dilakukan untuk mengetahui kualitas gambar keluaran hasil proses penyisipan, apakah memiliki tingkat kerusakan gambar yang rendah atau tinggi. Kualitas ini diukur dengan membandingkan gambar masukan dan keluaran. Pembandingan dilakukan dengan dua cara, yaitu dengan penampilan visual dan menggunakan PSNR *(Peak Signal-to-Noise Ratio)*. PSNR yang digunakan merupakan ukuran untuk menentukan rasio perbedaan piksel diantara dua buah gambar. Persamaan 10 menyatakan perhitungan PSNR yang digunakan dalam pengujian perangkat lunak ini.

$$PSNR = 20.\log_{10}\left(\frac{MAX_1}{\sqrt{MSE}}\right) \tag{10}$$

Keterangan :
MAX : nilai maksimum dari sebuah piksel (255)

Nilai MSE (*mean squared error*) dapat dihitung menggunakan persamaan 11.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\|I(i,j) - K(i,j)\right\|^2 \tag{11}$$

Keterangan :
m : tinggi gambar
n : lebar gambar
I(*i*,*j*): nilai piksel pada gambar asli
K(*i*,*j*): nilai piksel pada *stego-image*

Persamaan *MSE* di atas hanya berlaku untuk gambar dengan pewarnaan *grayscale*. Untuk gambar RGB, maka setiap nilai *channel red, green,* dan *blue* digunakan dalam persamaan 11, kemudian nilai MSE dibagi dengan tiga. Satuan yang biasanya digunakan untuk pengukuran menggunakan PSNR adalah satuan desibel (dB). Nilai PSNR yang wajar pada pembandingan dua dokumen gambar berkisar pada 30-50 dB (Thu dan Ganbari, 2008).

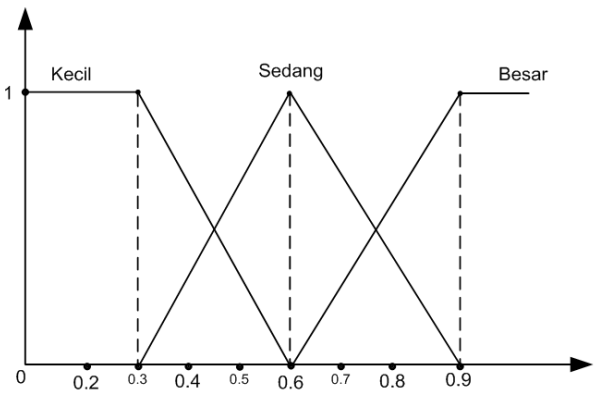
D. Rancangan Knowledge Base, Inferensi dan Defuzzifikasi

Fuzzy logic control memiliki empat bagian utama dalam pembuatan struktur dasar sistem kendali *fuzzy*, yaitu: Fuzzifikasi, Knowledge Base, Inferensi dan Defuzzifikasi [8]. Berikut adalah proses fuzzifikasi terdapat variabel input dan output yang digunakan dalam system ini.

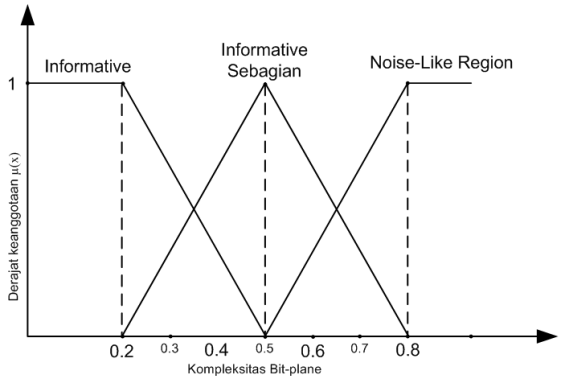
Fungsi keanggotaan ratio perbandingan ukuran pesan (yang ingin disisipkan) dan ukuran gambar memiliki tiga buah variabel linguistik yaitu kecil, sedang, dan besar. Varibel linguistik kecil menggunakan kurva bahu, sedangkan varibel lingusitik sedang menggunakan kurva segitiga dan varibel linguistik besar menggunakan kurva bahu. Untuk lebih jelasnya dapat dilihat pada Gambar 8.

Fungsi keanggotaan kompleksitas *bit-plane* memiliki tiga buah variabel linguistik yaitu *informative*, *informative* sebagian, dan *noise-like-regions.* Varibel linguistik *informative* menggunakan kurva bahu, sedangkan varibel lingusitik *informative* sebagian menggunakan kurva segitiga dan varibel linguistik *noise-like-regions* menggunakan kurva bahu. Untuk lebih jelasnya dapat dilihat pada Gambar 9.

Basis rule berisi aturan kendali *fuzzy* yang dijalankan untuk mencapai tujuan pengendalian. Aturan-aturan IF – THEN yang ada dikelompokkan dan disusun kedalam bentuk *Fuzzy Associative Memory* (FAM). FAM ini berupa suatu matriks yang menyatakan input-output sesuai



Gambar 8. kurva ratio perbandingan ukuran file



Gambar 9. Kurva kompleksitas *bit-plane*

dengan aturan IF – THEN pada basis aturan yang ada. Aturan yang telah dibuat harus dapat mengatasi semua kombinasi-kombinasi input yang mungkin terjadi, dan harus dapat menghasilkan sinyal kendali yang sesuai agar tujuan pengendalian tercapai. Untuk rule yang telah di bentuk dapat dilihat pada Tabel 1, sedangkan untuk fungsi keanggotaan output (*threshold*) dapat ditunjukkan pada Gambar 10.

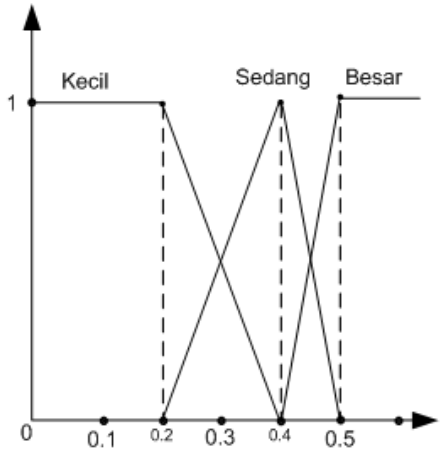
Defuzzyfikasi pada komposisi aturan mamdani dengan menggunakan metode *centroid*. Dimana pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah *fuzzy*.

IV. Hasil dan Pembahasan

A. Pengujian

Terdapat beberapa hal yang merupakan tujuan dari pengujian perangkat lunak yang dikembangkan dalam thesis ini, yaitu:

1. Memeriksa kesesuaian hasil implementasi perangkat lunak dengan spesifikasi kebutuhan yang ada.
2. Mengukur pengaruh penentuan *threshold* secara manual dengan variabel α, dan penentuan *threshold* menggunakan logika *fuzzy* dengan menggunakan variabel β terhadap kualitas *stego-image* yang dinilai secara visual.



Gambar 10. Representasi kurva output

Table 1. Rule untuk output *fuzzy*

Ukuran gambar dan pesan / kompleksitas	KCL	SDG	BSR
IF	Sedang	Sedang	Kecil
IFS	Sedang	Sedang	Kecil
NLR	Besar	Besar	Sedang

3. Mengukur kualitas *stego-image* berdasarkan nilai PNSR.

Untuk data yang digunakan pada pengujian perangkat lunak yang sudah dibangun dapat dilihat pada Tabel 2.

Gambar Abraham Lincoln memiliki ukuran 100x200, sedangkan gambar baboon memiliki ukuran 800x600 piksel, dan gambar lena memiliki ukuran 300x250 pixel.

B. Pelaksanaan dan Hasil Pengujian

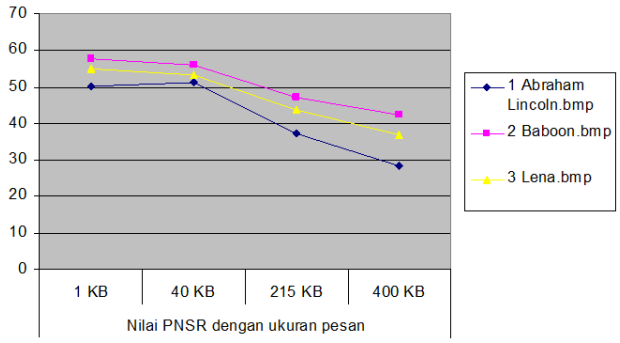
Pada bagian ini dijelaskan mengenai pelaksanaan pengujian dan hasil pengujian setiap kasus uji yang telah didefinisikan. Pada pengujian fungsionalitas perangkat lunak, dilakukan pengujian terhadap proses penyisipan dan ekstraksi. Pengujian dilakukan dengan melakukan penyisipan pada lima citra yang memiliki pewarnaan berbeda, kemudian melakukan ekstraksi pada *stego-image*.

Pengujian kualitas *stego-image* dilakukan dengan cara menyisipkan empat pesan pada masing-masing gambar pengujian. Pesan yang disisipkan terdiri dari tiga dokumen gambar dengan ukuran yang berbeda-beda. Hasil pengujian kualitas PNSR *stego-image* dapat dilihat pada Tabel 3..

PSNR yang digunakan merupakan ukuran untuk menentukan rasio perbedaan piksel diantara dua buah gambar. Disini gambar original dan gambar hasil

Tabel 2. Data Pengujian

No	Nama File	Gambar
1	abraham_lincoln.bmp	
2	baboon.bmp	
3	lena.bmp	



Gambar 11. Nilai PNSR

Tabel 3. *Summary* pengujian kualitas *stego-image*

No	Gambar	Nilai PNSR dengan ukuran pesan		
		40 KB	215 KB	400 KB
1	Abraham Lincoln.bmp	51.39	37.19	28.19
2	Baboon.bmp	55.94	47.21	42.31
3	Lena.bmp	53.27	43.67	36.95

penyisipan dibandingkan nilai pikselnya, hasil secara grafis dapat dilihat pada Gambar 11.

Dari grafik tersebut terlihat bahwa hasil penyisipan mengeluarkan *stego-image* yang memiliki kualitas baik hingga ukuran tertentu. Pada ukuran pesan yang sudah terlalu besar, kualitas gambar akan semakin menurun. Hal ini telah diperkirakan sebelumnya, karena untuk bisa menyisipkan pesan yang besarnya hingga 70%-80% dari *vessel image* pasti diimbangi dengan adanya kerusakan pada gambar keluaran. Pada grafik, hal ini terjadi pada saat dilakukan penyisipan dokumen sebesar 400 KB pada masing-masing gambar.

Dari pengujian yang telah dilakukan, dapat dilihat pula bahwa ukuran *stego-image* akan mengalami perubahan dibandingkan dengan ukuran asli. Ukuran tersebut dapat mengecil maupun membesar yang dipengaruhi oleh ukuran pesan yang disisipkan. Menurut pengujian yang telah dilakukan, semakin besar pesan yang disisipkan, maka semakin besar ukuran *stego-image*.

Penyisipan pesan yang dilakukan terhadap gambar Abraham Lincoln, maka dapat dilihat secara visual beberapa perbedaan seperti yang terlihat pada gambar 12.

Dari hasil pengamatan secara visual dari ketiga gambar tersebut perbedaan yang paling mencolok terjadi pada Gambar 12 (c) dimana terlihat *noise* yang cukup

(a). Gambar asli (b). *Threshold* 0.3 (c). *Threshold* 0.367

Gambar 12. Nilai PNSR Abrahman Lincoln

(a). Gambar asli (b). *Threshold* 0.3 (b). *Threshold* 0.367

Gambar 13. Nilai PNSR Lena

signifikan. Sementara pada Gambar 12 (b) juga terlihat *noise* tetapi tidak terlalu signifikan.

Berdasarkan hasil pengamatan Gambar 13 secara visual dari ketiga gambar tersebut terlihat perbedaan yang cukup mencolok antara gambar yang menggunakan nilai *threshold* 0.3 dan *threshold* 0.367. Terdapat *noise* yang lebih siginifikan pada gambar dengan *threshold* 0.3 dibanding *noise* yang terdapat pada gambar dengan *threshold* 0.367. hal ini dapat dilihat pada bagian wajah kedua gambar tersebut.

V. KESIMPULAN

Berdasarkan pembahasan pada bab sebelumnya, maka diambil beberapa kesimpulan sebagai berikut: pengukuran kompleksitas dengan menggunakan varibel β menghasilkan nilai PNSR yang lebih baik dari pada pengukuran kompleksitas dengan variabel α.

Pengujian nilai PSNR menunjukkan bahwa semakin besar pesan yang disisipkan, maka nilai *treshold* yang dihasilkan akan semakin kecil, dan kualitas *stego-image* yang dihasilkan semakin buruk.Pemilihan variabel input *fuzzy* berupa ukuran pesan dan kompleksitas *bit-plane* dapat mempermudah pengguna dalam menentukan nilai *threshold* yang akan digunakan dalam proses penyisipan. Gambar dengan format bitmap cukup cocok untuk penerapan steganografi dengan metode BPCS. Hal ini dikarenakan format bitmap menggunakan pewarnaan RGB. Selain itu gambar dengan format bitmap tidak dikompresi sehingga kerusakan pesan dapat dihindari. Melakukan modifikasi terhadap *stego-image* dapat menimbulkan pesan tidak bisa diekstraksi.

Untuk pengembangan lebih lanjut, saran-saran yang dapat diberikan adalah sebagai berikut: perlu adanya analisis lebih lanjut dan implementasi metode BPCS pada gambar dengan format lainnya seperti JPEG, GIF. Analisis yang diperlukan antara lain adalah penanganan kerusakan gambar pada dokumen GIF dan penanganan DCT pada kompresi JPEG untuk menangani kerusakan pesan. Perlu adanya analisis penerapan penggunaan variabel input dan output *fuzzy* yang berbeda sehingga hasilnya dapat dibandingkan dengan penelitian ini. Perlu adanya penelitian yang menggunakan ukuran *bit-plane* yang berbeda untuk melihat pengaruhnya terhadap kualitas dan daya tampung yang dihasilkan oleh sebuah gambar. Untuk meningkatkan keamanan pesan, maka metode-metode kriptografi dapat digunakan untuk mengenkripsi pesan sebelum disisipkan kedalam gambar.

REFERENSI

[1] Kawaguchi, E. dan Eason, R. O. 1998. Principle and Application of BPCS Steganography. Proceedings of SPIE: Multimedia Systems and Applications, vol.3528, hal. 464–472.

[2] Image Steganography System using Modified BPCS Steganography Method International Journal of Engineering Research & Technology (IJERT) IJERT ISSN: 2278-0181 Vol. 3 Issue 6, June – 2014

[3] N. Johnson and S. Jajodia, (Feb 1998): Exploring steganography: seeing the unseen, IEEE Computer, pp.26-34

[4] R.J. Anderson, F.A.P. Peticolas, (May 1998): On the Limits of Steganography, IEEE Journal of Selected Areas in communication

[5] Babu, K. S., Raja, K. B., Kiran, K. K., Manjula Devi, T. H., Venugopal, K. R., & Patnaik, L. M. (2008, 19- 21 Nov. 2008). Authentication of secret information in image Steganography. Paper presented at the TENCON 2008 - 2008 IEEE Region 10 Conference.

[6] Hirohisa, H. 2002. A Data Embedding Method Using BPCS Principle With New Complexity Measures, Proceedings Pacific Rim Workshop on Digital Steganography 2002, vol. 3423, hal.30-47

[7] Hedieh, S., & Jamzad, M. (2008, 8-11 July 2008). Cover Selection Steganography Method Based on Similarity of Image Blocks. Paper presented at the Computer and Information Technology Workshops, 2008. CIT Workshops 2008. IEEE 8th International Conference on.

[8] Klir, G.J., dan Yuan, B., 1995, Fuzzy Sets and Fuzzy Logic : Theory and Applications. Prentice Hall International Inc., Upper Saddle River, NJ 07458.

[9] Murguia, C., Saenz, dan A., Rodriguez., 2007, *"Fuzzy Approach on Image Complexity Measure"*, Journal of Computation System, vol. 10, hal. 268-248.

[10] Noda, Hideki., Furuta, Tomonori., Niimi, Michiharu., Kawaguchi., Eiji., 2004,*" Application of BPCS Steganography to Wavelet Compressed Video",* Kyushu Institute of Technology, Kyushu, Japan

[11] Shi, P. dan Li, Z. 2010. An improved BPCS Steganography based on Dynamic Threshold. 2010 International Conference on Multimedia Information Networking and Security. vol.32, hal. 231-259.

[12] Srinivasan, Y. 2003. "High Capacity Data Hiding System Using BPCS Steganography". Thesis. Texaz Tech University. Texas. Hal 20-25.

Penerbit:
Jurusan Teknik Elektro, Fakultas Teknik, Universitas Syiah Kuala
Jl. Tgk. Syech Abdurrauf No. 7, Banda Aceh 23111
website: http://jurnal.unsyiah.ac.id/JRE
email: rekayasa.elektrika@unsyiah.net
Telp/Fax: (0651) 7554336