

ANALISIS KUALITAS CITRA PADA PENYEMBUNYIAN PESAN PADA CITRA MENGGUNAKAN ALGORITMA GENETIKA

Syari Rukmana Wahab¹, Surlanti², Erpina D. C. Sihombing³

Staf Pengajar Pada AMIK UMEL Mandiri Jayapura

Abstraksi - Penelitian ini bertujuan untuk menerapkan Algoritma Genetika untuk mencari solusi menyisipkan pesan di dalam citra dan menganalisis kualitas citra setelah diisi oleh file. Penelitian ini dilaksanakan di Lab. STMIK Handayani Makassar. Metode yang digunakan pada penelitian ini adalah Enkripsi dan Dekripsi pesan menggunakan Algoritma AES, untuk pencarian penempatan pesan pada citra menggunakan Algoritma Genetika, untuk menyisipkan pesan pada citra menggunakan teknik LSB. Hasil penelitian yang diperoleh menunjukkan bahwa Algoritma Genetika dapat digunakan untuk pencarian solusi penempatan penyembunyian file pada citra, dimana kualitas citra yang dihasilkan pada penyisipan citra dalam citra dengan melihat nilai PSNR terendah berada pada citra cover 64 x 64 yang berada pada dimensi 30 x 30 dengan nilai PSNR sebesar 20,33 dB, dan nilai PSNR tertinggi tidak dapat didefinisikan, dan kualitas citra yang dihasilkan pada penyisipan pesan dalam citra dengan melihat nilai PSNR terendah berada pada citra cover 128 x 128 yang berada pada pesan 10 dengan nilai PSNR sebesar 33,98 dB, dan nilai PSNR tertinggi tidak dapat didefinisikan

Kata Kunci: Steganografi, Algoritma Genetika, kualitas citra, PSNR

1. PENDAHULUAN

Kerahasiaan pesan merupakan bagian dari ilmu kriptografi. Pada umumnya, pesan dijaga kerahasiaannya dengan cara dienkripsi, sehingga pesan menjadi tidak dapat dibaca dan digunakan. Akan tetapi, pesan yang tidak terbaca (dienkripsi) dapat menimbulkan kecurigaan bahwa data yang dikirim merupakan hal yang penting. Untuk menghindarinya, maka dapat digunakan ilmu steganografi. Steganografi merupakan salah satu ilmu dalam menangani masalah keamanan data. Berbeda dengan kriptografi, steganografi menyembunyikan data rahasia dalam keadaan tidak terlihat karena data disembunyikan di dalam suatu media, sehingga tidak menimbulkan kecurigaan. Media yang digunakan untuk penyembunyian data dapat berupa citra digital.

Berbagai teknik penempatan pesan dalam citra telah dilakukan antara lain Sekuensial (berurutan) yaitu teknik penyembunyian pesan secara sekuensial (berurutan) berarti pesan rahasia disisipkan secara berurutan dari data titik pertama yang ditemukan pada file gambar penyisipan dilakukan dari indeks (0,0), dari kiri ke kanan baris per baris sepanjang bit-bit pesan yang disembunyikan (Erwin yudi hidayat dkk, 2013). metode linear congruential Generator (LCG) yang artinya piksel yang telah dipilih secara semu acak menjadi tempat penyisipan pesan akan di cek nilai paritasnya dengan bit pesan (Prasetyo Andy wicaksono, 2013). dan juga sistem steganografi yang di implementasikan menggunakan Algoritma F5 yang didalamnya menerapkan matrix encoding untuk meningkatkan efisiensi pada saat penyisipan, yang bertujuan agar pergantian bit yang dilakukan menjadi lebih efektif (Yuda Trijaya, dkk, 2007). Algoritma Genetika merupakan salah satu metode yang dapat digunakan dalam steganografi. sebagai metode pencarian solusi optimal yang berfungsi menempatkan file-file yang akan disisipkan ke dalam citra.

2. TINJAUAN PUSTAKA

2.1. Steganografi

Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sedemikian rupa sehingga orang lain tidak menyadari ada pesan yang disembunyikan di dalam media tersebut. Kata steganografi (steganography) berasal dari bahasa Yunani steganos yang artinya "tersembunyi / terselubung", dan graphein yang artinya "menulis, sehingga steganografi artinya adalah "menulis (tulisan) terselubung".

Dalam bidang keamanan komputer, steganografi digunakan untuk menyembunyikan data rahasia, saat enkripsi tidak dapat dilakukan. Walaupun enkripsi berhasil dipecahkan (decipher), pesan atau data rahasia tetap tidak terlihat. Pada cryptography, pesan disembunyikan dengan "diacak" sehingga pada kasus-kasus tertentu dapat dengan mudah mengundang kecurigaan, sedangkan pada steganografi, pesan "disamarkan" dalam bentuk yang relatif aman sehingga tidak terjadi kecurigaan itu. Seperti yang terjadi pada peristiwa penyerangan gedung WTC pada 11 September 2001, disebutkan oleh "pejabat pemerintah dan para ahli dari pemerintahan AS", yang tidak disebutkan namanya, bahwa "para teroris menyembunyikan peta-peta dan foto target serta perintah untuk aktivitas teroris di ruang chat sport, bulletin boards porno, dan website lainnya". Isu lainnya menyebutkan bahwa teroris menyembunyikan pesan-pesannya dalam gambar-gambar porno di website tertentu.

Steganografi membutuhkan dua properti, yaitu wadah penampung dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, audio, teks dan video. Gambar 2.1 adalah ilustrasi untuk menggambarkan proses penyimpanan (penyisipan) pesan ke dalam media digital dengan teknik steganografi.



Gambar 2.1. Ilustrasi Steganografi

2.2. Perbedaan Steganografi dan Watermarking

Sebenarnya, watermarking adalah bagian dari steganografi. Watermarking merupakan aplikasi dari steganografi. Hanya saja yang perlu dipahami adalah pada steganografi, pesan rahasia disembunyikan di dalam media digital (pesan rahasia lebih penting daripada media digitalnya), sedangkan pada watermarking, media digital lebih penting daripada pesan (watermark) yang disisipkan ke media digital tersebut. Media digital ini akan dilindungi kepemilikannya dengan pemberian label hak cipta (watermark).

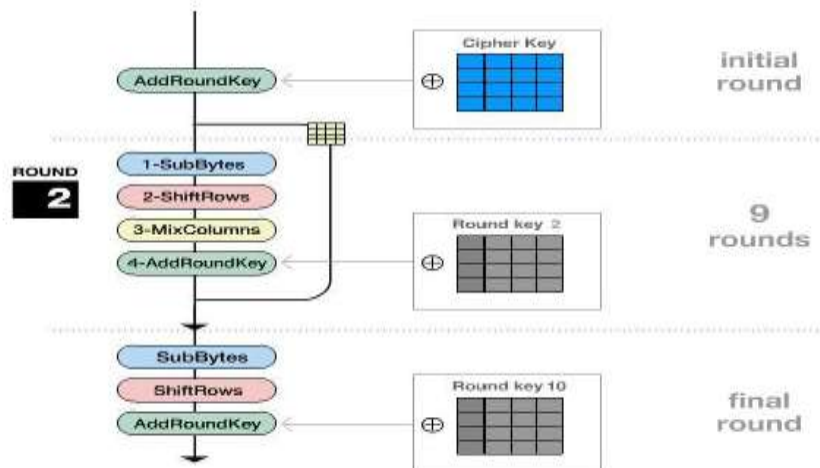
2.3. Advanced Encryption Standard (AES)

Advanced Encryption Standard (AES) merupakan algoritma cryptographic yang dapat digunakan untuk mengamankan data. Algoritma AES adalah blokchipertext simetrik yang dapat mengenkripsi (encipher) dan dekripsi (decipher) informasi. Enkripsi merubah data yang tidak dapat lagi dibaca disebut ciphertext; sebaliknya dekripsi adalah merubah ciphertext data menjadi bentuk semula yang kita kenal sebagai plaintext. Algoritma AES menggunakan kunci kriptografi 128, 192, dan 256 bits untuk mengenkrip dan dekrip data pada blok 128 bits.

AES (Advanced Encryption Standard) adalah lanjutan dari algoritma enkripsi standar DES (Data Encryption Standard) yang masa berlakunya dianggap telah usai karena faktor

keamanan.Keamanan merupakan faktor terpenting dalam evaluasi (minimal seaman triple DES), yang meliputi ketahanan terhadap semua analisis sandi yang telah diketahui dan diharapkan dapat menghadapi analisis sandi yang belum diketahui.AES juga harus efisien dan cepat (minimal secepat Triple DES) dijala.,mknk dalam berbagai mesin 8 bit hingga 64 bit, dan berbagai perangkat lunak. DES menggunakan stuktur Feistel yang memiliki kelebihan bahwa struktur enkripsi dan dekripsinya sama, meskipun menggunakan fungsi F yang tidak invertibel. Kelemahan Feistel yang utama adalah bahwa pada setiap ronde, hanya setengah data yang diolah.Sedangkan AES menggunakan struktur SPN (Substitution Permutation Network) yang memiliki derajat paralelisme yang lebih besar, sehingga diharapkan lebih cepat dari pada **Feistel**.

AES ini merupakan algoritma block cipher dengan menggunakan sistem permutasi dan substitusi (P-Box dan S-Box) bukan dengan jaringan **Feistel** sebagaimana block cipher pada umumnya. Jenis AES terbagi 3, yaitu :AES-128 , AES-192 , AES-256. Pengelompokkan jenis AES ini adalah berdasarkan panjang kunci yang digunakan.Angka-angka di belakang kata AES menggambarkan panjang kunci yang digunakan pada tipa-tiap AES.Selain itu, hal yang membedakan dari masing-masing AES ini adalah banyaknya round yang dipakai. AES-128 menggunakan 10 round, AES-192 sebanyak 12 round, dan AES-256 sebanyak 14 round. AES memiliki ukuran block yang tetap sepanjang 128 bit dan ukuran kunci sepanjang 128, 192, atau 256 bit. Tidak seperti Rijndael yang block dan kuncinya dapat berukuran kelipatan 32 bit dengan ukuran minimum 128 bit dan maksimum 256 bit. Berdasarkan ukuran block yang tetap, AES bekerja pada matriks berukuran 4x4 di mana tiap-tiap sel matriks terdiri atas 1 byte (8 bit). Diagram AES dapat dilihat pada Gambar di bawah.



Gambar 2.2 Diagram AES

2.4. Metode Least Significant Bit (LSB)

Metode LSB (Least significant Bit) merupakan metode steganografi yang paling Sederhana dan apa paling mudah diimplementasikan. Untuk menjelaskan metode ini kita menggunakan citra digital sebagai covertex.Setiap pixel di dalam citra berukuran 1 sampai 3 byte. Pada susunan bit di dalam byte (1 byte = 8 bit), ada bit yang paling berarti (most significant bit atau MSB) dan bit yang paling kurang berarti ((Least significant Bit atau LSB).misalnya pada byte 11010010,bit 1 yang pertama (digarisbawahi) adalah adalah bit MSB dan bit 0 yang terakhir (digarisbawahi) adalah bit LSB.bit yang cocok untuk diganti dengan bit pesan adalah bit LSB. sebab modifikasi hanya mengubah nilai byte tersebut satu lebih tinggi atau satu lebih rendah dari

nilai sebelumnya. Misalkan byte tersebut di dalam gambar memberikan persepsi warna merah tidak terlalu berarti. Mata manusia tidak dapat membedakan perubahan yang kecil ini.

2.5. Algoritma Genetika

Evolutionary Algorithm merupakan terminologi umum yang menjadi payung bagi empat istilah : Algoritma genetika (genetika Algorithm), pemrograman genetika (genetic Programming), strategi evolusi (evolution strategies), dan pemrograman evolusi (evolutionary Programming). Tetapi, jenis evolutionary algorithm yang paling populer dan banyak digunakan adalah algoritma genetika (genetic algorithm). Algoritma genetika merupakan evolusi / perkembangan dunia komputer dalam bidang kecerdasan buatan (artificial intelligence). Sebenarnya kemunculan algoritma genetika ini terinspirasi oleh teori evolusi Darwin (walaupun pada kenyataannya teori tersebut terbukti keliru) dan teori-teori dalam ilmu biologi, sehingga banyak istilah dan konsep biologi yang digunakan. Karena itu sesuai dengan namanya, proses-proses yang terjadi dalam algoritma genetika sama dengan apa yang terjadi pada evolusi biologi.

Algoritma genetika merupakan teknik pencarian nilai optimum secara stochastic berdasarkan mekanisme seleksi alam. Algoritma genetika berbeda dengan teknik konvergensi konvensional yang lebih bersifat deterministik. Metodenya sangat berbeda dengan kebanyakan algoritma optimasi lainnya, yaitu mempunyai ciri-ciri sebagai berikut:

- a. Menggunakan hasil pengkodean dari parameter, bukan parameter itu sendiri.
- b. Bekerja pada populasi bukan pada sesuatu yang unik.
- c. Menggunakan nilai satu-satunya pada fungsi dalam prosesnya. Tidak menggunakan fungsi luar atau pengetahuan luar lainnya.
- d. Menggunakan fungsi transisi probabilitas, bukan sesuatu yang pasti.

2.6. Proses Algoritma Genetika

Algoritma genetika adalah algoritma pencarian yang berdasarkan pada berdasarkan mekanisme sistem natural yakni genetik dan seleksi alam. Dalam aplikasi algoritma genetik, variable solusi dikodekan ke dalam struktur string yang merepresentasikan barisan gen, yang merupakan karakteristik dari solusi problem. Berbeda dengan teknik pencarian konvensional, algoritma genetic berangkat dari himpunan solusi yang dihasilkan secara acak. Himpunan ini disebut populasi. Sedangkan setiap individu dalam populasi disebut kromosom yang merupakan representasi dari solusi. Kromosom-kromosom berevolusi dalam suatu proses iterasi yang berkelanjutan yang disebut generasi. Pada setiap generasi, kromosom dievaluasi berdasarkan suatu fungsi evaluasi. Setelah beberapa generasi maka algoritma genetik akan konvergen pada kromosom terbaik, yang diharapkan merupakan solusi optimal.

Tujuan dari algoritma genetika ini adalah menghasilkan populasi yang terbaik dari populasi awal. Sedangkan keuntungan dari algoritma genetika ini adalah sifat metode pencariannya yang lebih optimal, tanpa terlalu memperbesar ruang pencarian. Dalam menyusun suatu algoritma genetika menjadi program, maka diperlukan tahapan proses, yaitu proses pembuatan generasi awal, proses genetika, proses seleksi, dan pengulangan proses sebelumnya.

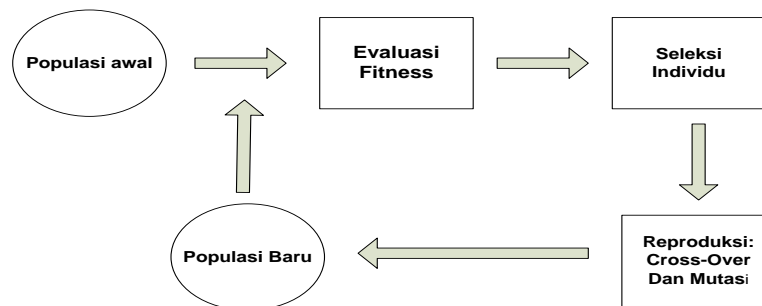
Pada algoritma genetika ini terdapat beberapa definisi penting yang harus dipahami sebelumnya, yaitu:

- a. Gen
Gen merupakan nilai yang menyatakan suatu dasar yang membentuk suatu arti tertentu dalam suatu kesatuan gen yang dinamakan kromosom.
- b. Kromosom / Individu
Kromosom merupakan gabungan dari gen-gen yang membentuk nilai tertentu dan menyatakan solusi yang mungkin dari suatu permasalahan. Ada beberapa macam bentuk kromosom, yaitu:
 - a. **Kromosom Biner**, adalah kromosom yang disusun dari gen-gen yang bernilai biner kromosom ini mempunyai tingkat keberhasilan yang tinggi. Jumlah gen pada kromosom biner menunjukkan tingkat ketelitian yang diharapkan.
 - b. **Kromosom float**, adalah kromosom yang disusun dari gen-gen yang bernilai pecahan, termasuk gen yang bernilai bulat. Kromosom ini merupakan model yang jumlah

- parameternya banyak. Tingkat keberhasilan dari kromosom ini rendah dalam kecepatan (jumlah generasi).
- c. **Kromosom string**, yaitu kromosom yang disusun dari gen-gen yang bernilai string.
 - d. **Kromosom kombinatorial**, yaitu kromosom yang disusun dari gen-gen yang dinilai berdasarkan urutannya.
- c. **Populasi**
Populasi merupakan sekumpulan individu yang akan diproses bersama dalam satu satuan siklus evolusi.
 - d. **Fitness**
Fitness menyatakan seberapa baik nilai dari suatu individu yang didapatkan
Nilai fitness ini yang dijadikan acuan dalam mencapai nilai optimal dalam algoritma genetika. algoritma genetika bertujuan mencari individu dengan nilai fitness yang paling tinggi.
 - e. **Seleksi**
Seleksi merupakan proses untuk mendapatkan calon induk yang baik.
 - f. **Crossover (pindah silang)**
Crossover merupakan proses pertukaran atau kawin silang gen-gen dari dua induk tertentu.
 - g. **Mutasi**
Mutasi merupakan proses pergantian salah satu gen yang terpilih dengan nilai tertentu.
 - h. **Generasi**
Generasi merupakan urutan iterasi dimana beberapa kromosom bergabung.
 - i. **Offspring**
Offspring merupakan kromosom baru yang dihasilkan.

2.7. Siklus Algoritma Genetika

Siklus Algoritma genetika pertama kali dikenalkan oleh David Goldberg, dimana siklus dimulai dari dari membuat populasi awal secara acak, kemudian setiap individu dihitung nilai fitness-nya. proses berikutnya menyeleksi individu terbaik, kemudian dilakukan *cross-over* dan dilanjutkan oleh proses mutasi sehingga terbentuk populasi baru. Selanjutnya populasi baru ini mengalami siklus yang sama dengan populasi sebelumnya. Proses ini berlangsung terus hingga generasi ke -n.



Gambar 2.3. Siklus Algoritma Genetika oleh David Goldberg

(Sumber: entin.lecturer.pens.ac.id/.../Bab%207%20Algoritma%20Genetika.pdf diakses : Agustus 2015)

2.8. Peak Signal to Noise Ratio (PSNR)

Peak signal to Noise Ratio (PSNR) adalah perbandingan antara lain maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam setiap satuan decibel. PSNR digunakan untuk mengetahui perbandingan kualitas citra sebelum dan sesudah disisipkan pesan. Untuk PSNR, terlebih dahulu harus ditentukan nilai rata-rata kuadrat dari error (MSE – Mean Square Error). (Putri Alatas , 2009) Perhitungan MSE adalah sebagai berikut:

$$MSE = \frac{1}{mn} \sum_i^m \sum_j^n \|I(i,j) - K(i,j)\|^2$$

Dimana:

- MSE = Nilai Mean Square Error dari citra
- M = panjang citra (dalam piksel)
- N = lebar citra (dalam piksel)
- (i,j) = koordinat masing-masing piksel
- I = nilai Bit citra pada koordinat i,j
- K = nilai derajat keabuan citra pada koordinat i,j

Sedangkan nilai PSNR dihitung dari kuadrat nilai maksimum sinyal dibagi dengan MSE. Apabila diinginkan PSNR dalam decibel (dB), maka nilai PSNR akan menjadi sebagai berikut :

$$PSNR = 10 \cdot \log \left(\frac{MAX_i^2}{MSE} \right)$$

Dimana:

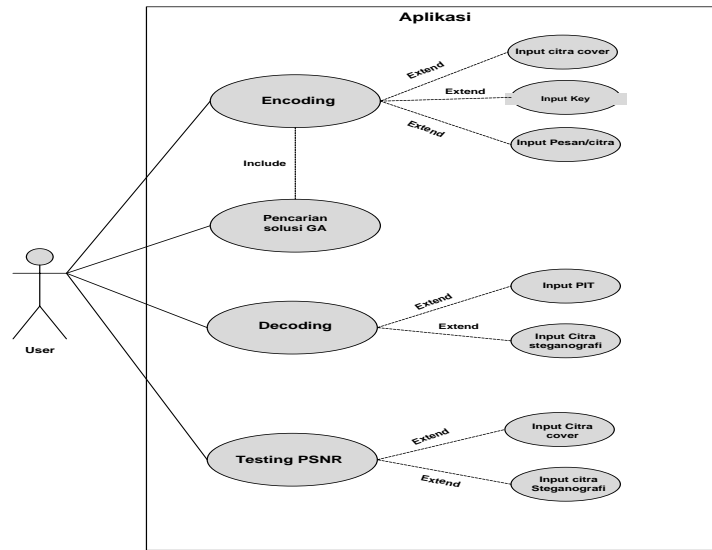
- PSNR = nilai PSNR citra (dalam dB)
- MAX_i = nilai maksimum piksel
- MSE = nilai MSE

Peak Signal to Noise Ration (PSNR) yang merupakan standar pengukuran error pada pengolahan citra .kualitas citra yang telah dimanipulasi pada umumnya mempunyai nilai PSNR yang berbeda pada rentang antara 20-40 dB. Jika PSNR yang tinggi maka kualitas citra akan lebih baik. (Hazuar Nurwanto, 2015).

3. METODOLOGI PENELITIAN

3.1. Use Case Diagram

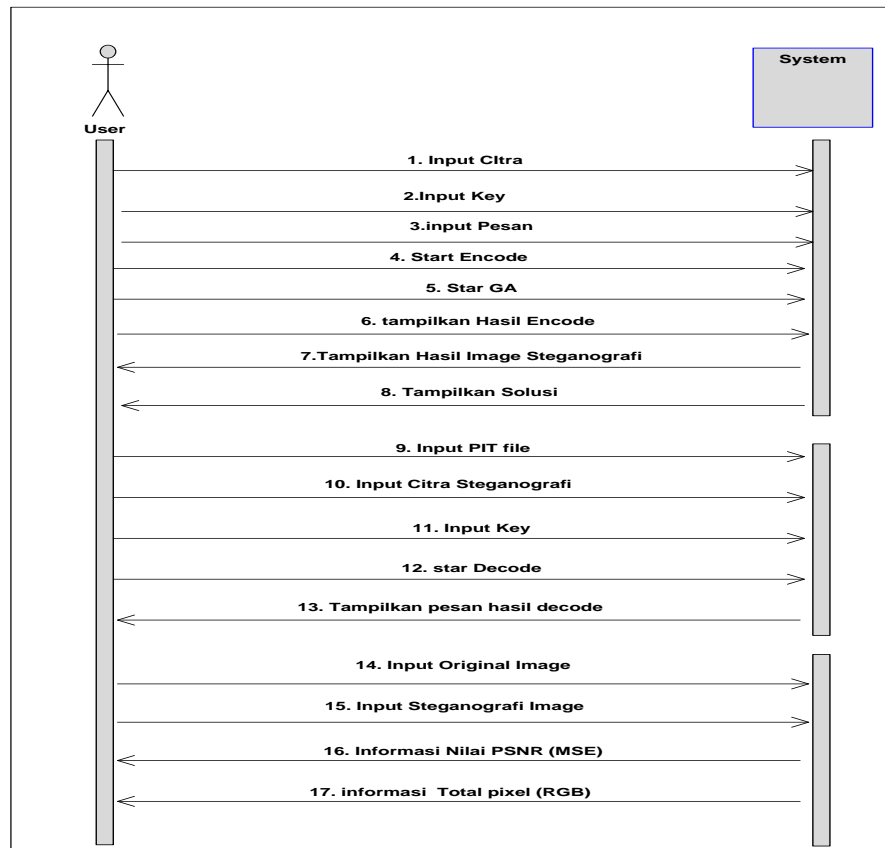
Use Case Diagram digunakan untuk menunjukkan hubungan antara *Use Case* dengan beberapa atau semua aktor serta interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun. *Use Case Diagram* menjelaskan manfaat suatu sistem jika dilihat menurut pandang orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau bagaimana sistem tersebut berinteraksi dengan dunia luar.



Gambar 2.4. Use case diagram

3.2. Sequence Diagram

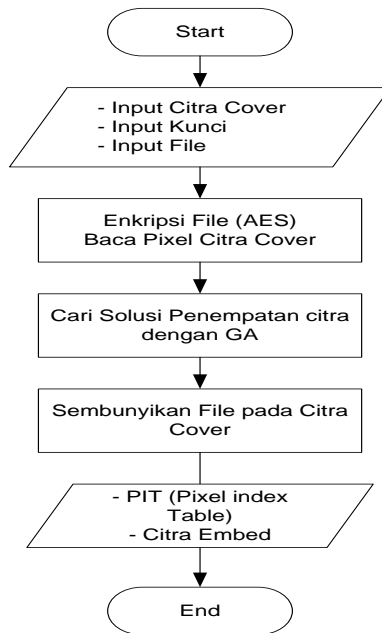
Sequence Diagram adalah diagram yang menggambarkan perilaku objek pada Use Case dan Message yang dikirimkan atau diterima antar objek.



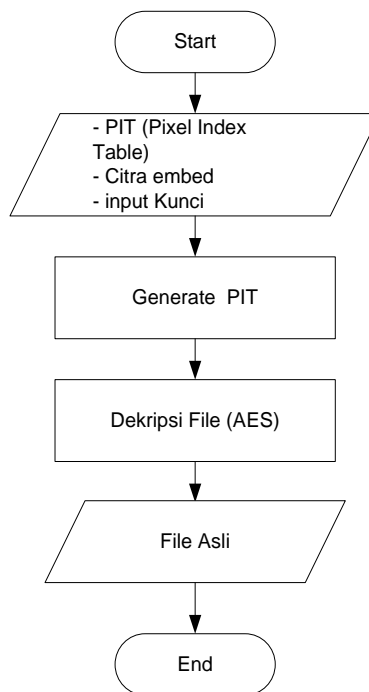
Gambar 2.5. Sequence Diagram

3.3. Alur Perancangan Sistem

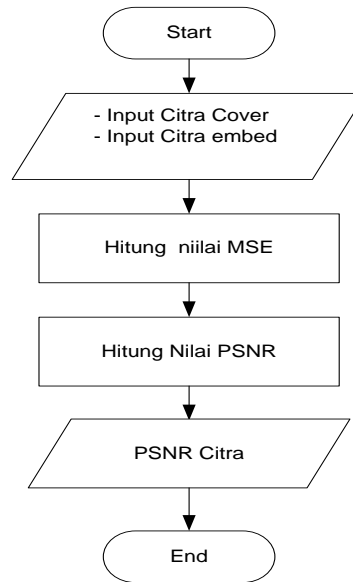
Rancangan alur sistem yang akan dikembangkan pada penelitian ini dibagi atas 3 bagian yaitu pertama proses encoding, kedua proses decoding dan yang ketiga pengujian kualitas citra dengan PSNR



Gambar 2.6. Flowchart Encoding



Gambar 2.7. Flowchart Decoding



Gambar 2.8. Flowchart PSNR

Adapun penjelasan dari Diagram Alur perancangan Sistem sebagai berikut:

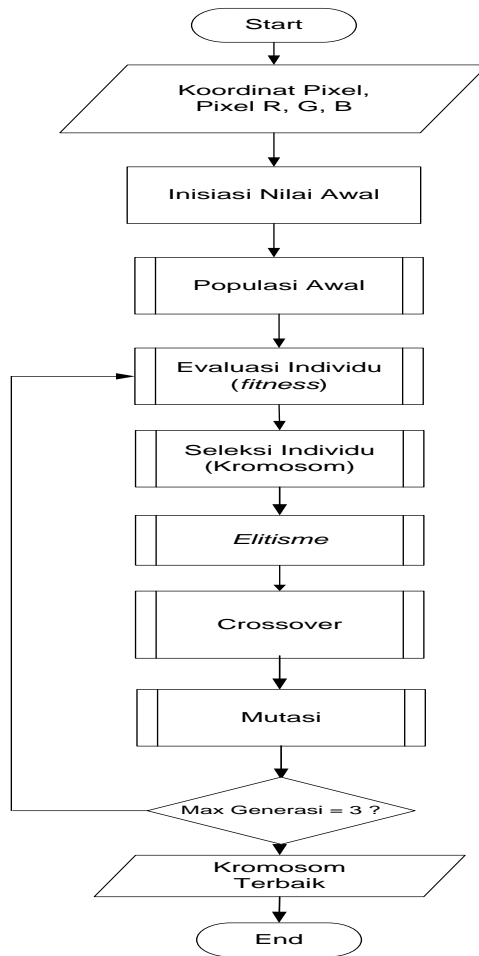
a. Proses Encoding

1. Proses Enkripsi Pesan

Citra yang digunakan sebagai citra cover merupakan citra berekstensi BMP 24 bit. Kunci Untuk proses enkripsi minimal 16 karakter yang dimana terdiri dari karakter yang dimana terdiri dari karakter (0,1,2,3,..9,A,B,C...F). File yang akan disembunyikan merupakan file berformat .txt. File yang akan disembunyikan kedalam citra terlebih dahulu akan di enkripsi untuk mengamankan pesan yang disembunyikan. Adapun sistem kerja AES adalah pertama semua karakter dalam file dan kunci akan dikonversi ke nilai byte. hasil dari dari enkripsi dari pesan berupa *chipertext* kemudian akan dikonversi ke *byte code*, apabila nilai byte berupa bilangan negative, maka akan dikonversi menjadi bilangan byte positif.

2. Penentuan Solusi Penempatan pesan dalam Citra dengan Algoritma Genetika

Dalam sistem penyembunyian pesan ini algoritma genetika digunakan sebagai pencarian penempatan pesan ke dalam citra cover. Secara umum proses pencarian solusi penempatan pesan dalam citra dengan algoritma genetika.



Gambar 2.9.Flowchart Proses pencarian Solusi dengan GA

Penentuan Populasi Awal merupakan langkah awal pada penentuan pesan menggunakan algoritma genetika adalah mengambil semua nilai pixel pada citra cover, setelah itu dilanjutkan dengan membaca tinggi dan lebar dari citra cover. Kemudian nilai-nilai dalam citra yaitu R, G, B serta koordinat X dan Y dari citra akan dikonversi ke dalam gen-gen yang akan membentuk kromosom. Tahapan pertama dalam algoritma genetika adalah inialisasi populasi yakni melakukan penentuan nilai awal.

Bagian penentuan nilai awal ini merupakan parameter yang digunakan pada algoritma genetika meliputi:

1. Penentuan banyak generasi yang akan digunakan. Maksimal generasi yang digunakan pada sistem ini adalah 3 generasi.
2. Penentuan besar *crossover probability*, (peluang terjadinya pindah silang). 0,5
3. Penentuan besar *mutation rate* atau *probabilitas mutasi*. 0,05

Tahap kedua dari AG adalah evaluasi individu, dimana proses ini akan menghitung nilai *fitness* dari setiap kromosom yang telah dibangkitkan secara *random* pada tahap inialisasi populasi di atas. Untuk mencari nilai *fitness* setiap kromosom digunakan fungsi:

$$f(x) = \min_i (abs(x - r), abs(x - g), abs(x - b))$$

Dimana:

x = Solusi

r = Nilai Pixel R

g = Nilai Pixel G

b = Nilai Pixel B

Apabila nilai *fitness* masing-masing kromosom telah didapatkan, untuk menentukan kromosom mana yang terbaik maka dilakukan perbandingan terhadap nilai-nilai *fitness* kromosom. Hasilnya disimpan pada variabel *fitness* yang merupakan argumen output fungsi seperti halnya *populasi* di atas. nilai *fitness* ini merupakan *input* bagi proses berikutnya pada program utama. Pada program utama, tahap evaluasi individu ini dilakukan (*di looping*) sebanyak ukuran maksimal dari kromosom (*popsize*). Sehingga didapat nilai *fitness* dari semua kromosom dalam satu populasi. Nilai *fitness* suatu kromosom ini kemudian akan dibandingkan dengan *fitness-fitness* kromosom yang lainnya yang ada pada semua generasi.

Tahapan selanjutnya proses seleksi kromosom yang dijadikan induk (*parent*). Proses seleksi adalah proses pemilihan kromosom. Pada sistem ini digunakan sistem seleksi menggunakan sistem perbandingan nilai *fitness* kromosom. Kromosom akan diurutkan mulai dari yang terbesar hingga yang terkecil. Setelah kromosom itu terpilih maka kromosom itu langsung akan menjadi kromosom awal pada generasi yang baru. Dengan kata lain setelah melewati proses seleksi itu maka akan berganti ke generasi baru.

Setelah proses seleksi individu dilakukan, proses selanjutnya adalah proses elitisme. Proses seleksi dilakukan dengan meranking nilai *fitness* kromosom akan tetapi mungkin saja kromosom tersebut akan rusak (nilai *fitness*nya menurun) karena proses pindah silang. Oleh karena itu, untuk menjaga agar individu bernilai *fitness* tertinggi tersebut tidak hilang selama evolusi, maka perlu dibuat satu atau beberapa kopian dari kromosom-kromosom yang mempunyai nilai *fitness* terbaik.

Tahapan selanjutnya adalah proses pindah silang digunakan untuk mendapatkan nilai optimal dari sebuah proses genetika. Salah satu syarat dari pindah silang (*crossover*) adalah pindah silang hanya bisa dilakukan dengan jika sebuah bilangan r dalam interval $0 - 1$ yang dibangkitkan nilainya lebih kecil dari probabilitas tertentu dengan kata lain $r < \text{prob}$. Nilai probabilitas yang digunakan pada sistem ini adalah 0.5. Ketika nilai probabilitas atau laju mutasi lebih kecil daripada bilangan r maka proses *crossover* tidak dilaksanakan. Hasil akhir dari prosedur pindah silang ini adalah dua buah kromosom anak hasil persilangan atau perkawinan dua induk (*parent*). Anak (*children*) ini merupakan input untuk proses berikutnya yakni mutasi.

Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh parameter *mutation rate* atau probabilitas mutasi (pm) yang telah ditetapkan yaitu 0,05. Proses mutasi dilakukan dengan metode *random gen* yaitu dengan cara mengganti satu gen yang terpilih secara acak dengan suatu nilai baru yang didapat secara acak.

Langkah mutasi hampir sama dengan *crossover*, yakni tiap-tiap individu dibangkitkan nilai *random r* $[0..1]$, nilai *random* ini akan melekat pada tiap individu. Misal pada perhitungan ditentukan *probabilitas* mutasi (pm) adalah sebesar 0.05, maka diharapkan chromosome yang memiliki nilai r dibawah nilai probabilitas mutasi dari tiap satu populasi mengalami proses mutasi. Maka kromosom ke k akan dipilih untuk dikenai mutasi jika $r(k)$ lebih kecil dari probabilitas mutasi.

Setelah individu yang mengalami mutasi terpilih, selanjutnya menentukan posisi gen pada tiap-tiap individu yang akan mengalami mutasi. Penentuan posisi gen dilakukan secara random agar didapat hasil individu baru yang bervariasi. Maka nilai gen pada posisi tersebut diganti dengan bilangan acak yang telah ditentukan. Proses eksekusi mutasi akan dilaksanakan sebanyak jumlah populasi yang ada. Untuk pergantian populasi dalam suatu generasi digunakan *general replacement* yaitu pergantian populasi secara keseluruhan. Populasi pada generasi sebelumnya yang merupakan induk (*parent*) diganti seluruhnya dengan populasi baru yang merupakan anak (*children*) atau turunannya (*offspring*). Populasi pada generasi berikutnya adalah kromosom bentuk baru hasil pindah silang dan mutasi.

Prosedur yang sama akan berlaku untuk populasi baru, yakni akan mengalami tahapan yang sama dengan populasi sebelumnya. Apabila perhitungan dilanjutkan sampai maksimal generasi yang telah ditentukan maka akan didapatkan nilai *fitness* tertinggi dari seluruh generasi yang menunjukkan kromosom terbaik yang akan diambil sebagai solusi.

Proses Penyembunian Citra dengan metode Least Signifikan Bit

Setelah didapat solusi penempatan penyisipan dengan algoritma genetika maka tahapan selanjutnya adalah menyisipkan file ke dalam citra cover. Teknik yang digunakan disini adalah teknik Least Signifikan Bit (LSB). Semua nilai pixel diset mula-mula dengan nilai -1 kemudian dilakukan pencarian nilai pixel sesuai dengan solusi yang didapatkan oleh algoritma genetika.

3. Proses Decode

Proses dekripsi merupakan proses mengembalikan pesan yang telah disembunyikan dalam citra menjadi pesan *original*. Proses decode merupakan kebalikan dari proses encode. Pada proses decode ada data menjadi inputan yaitu file *Pixel Index Table* (PIT) dan citra embed. File PIT merupakan parameter yang merupakan hasil dari pencarian solusi dengan algoritma genetika dimana PIT berisi nilai koordinat X dan Y dari penyembunian pesan dan serta nilai bit yang LSB yang digantikan. Sebelum dilakukan proses decode terlebih dahulu dilakukan pengecekan terhadap citra, apabila ukuran citra sesuai maka akan dilakukan proses decode. nilai yang ada pada tabel PIT akan *digenerate* untuk dilakukan pembacaan koordinat piksel yang digantikan serta bit-bit yang dimodifikasi. Kemudian dilakukan pembalikan kembali nilai yang didapatkan seperti pada proses encode. proses selanjutnya adalah membaca key atau password yang digunakan pada saat encode. apabila password yang digunakan tidak sesuai maka proses akan berhenti. Apabila semua parameter yang digunakan sebagai inputan maka proses decode akan dilakukan.

B. Pengujian PSNR

Untuk menguji kualitas citra yang telah disisipi oleh pesan, maka akan dilihat dari PSNR dalam sistem sebagai ada. Sebelum mengitung nilai PSNR, nilai yang pertama harus diketahui adalah nilai *mean square error* (MSE). Nilai MSE didapatkan dari mengitung jumlah pixel pada masing-masing lapisan yang ada pada citra cover dan citra embed, kemudian nilai masing-masing lapisan dikurangi setelah itu dibagi dari hasil kali Panjang dan lebar citra. Setelah itu akan dihitung nilai MSEnya dengan mengitung $\sqrt{R^2 + G^2 + B^2}$. Apabila ukuran citra yang dibandingkan tidak sama maka perhitungan nilai PSNR tidak dilakukan. Untuk mengitung PSNR citra maka dihitung dengan menggunakan rumus:

$$PSNR = 10 \cdot \log \left(\frac{MAX \frac{I}{I}}{MSE} \right)$$

Dimana nilai MAX merupakan nilai intensitas piksel maksimum di masing-masing lapisan citra.

4. HASIL DAN PEMBAHASAN

Setelah sistem selesai dibuat maka dilakukan pengujian terhadap 4 (empat) sampel citra yang digunakan sebagai citra cover dengan ukuran masing-masing citra adalah 64 x 64, 128 x 128, 256 x 256 dan 512 x 512. Pengujian dilakukan menggunakan 2 (dua) skenario, skenario pertama pengujian dilakukan dengan menggunakan 4 (empat) citra yang akan disisipkan dengan dimensi ukuran yang berbeda yaitu 10x10, 20x20, 30x30, 40x40, dan 50x50, 60x60, 70x70, 80x80, 90x90, 100x100, 200x200, 300x300, dan 400x400. Skenario kedua pengujian dilakukan dengan menyisipkan 10 (sepuluh) sampel pesan dimana setiap pesan memiliki panjang karakter yang berbeda-beda.

Tabel 1. Tabel Hasil pengujian kualitas citra skenario 1

Ukuran Citra	Dimensi Max yang bisa disisip	PSNR	Kriteria kualitas citra
13 KB (64 x 64)	10 x 10	48,13	Good
	20 x 20	37,33	Reasonable
	30 x 30	20,33	Usunable
49 KB (128 x 128)	10 x 10	Tidak terdefinisi	Good
	20 x 20	48,13	Good
	30 x 30	43,35	Reasonable
	40 x 40	39,09	Reasonable
	50 x 50	33,98	Poor picture
	60 x 60	20,56	Usunable
193 KB (256 x 256)	10 x 10	Tidak terdefinisi	Good
	20 x 20	Tidak terdefinisi	Good
	30 x 30	Tidak terdefinisi	Good
	40 x 40	48,13	Good
	50 x 50	45,12	Reasonable
	60 x 60	43,35	Reasonable
	70 x 70	41,14	Reasonable
	80 x 80	39,09	Reasonable
	90 x 90	36,99	Poor picture
	100 x 100	34,15	Poor picture
769 KB (512 x 512)	10 x 10	Tidak terdefinisi	Good
	20 x 20	Tidak terdefinisi	Good
	30 x 30	Tidak terdefinisi	Good
	40 x 40	Tidak terdefinisi	Good
	50 x 50	Tidak terdefinisi	Good
	60 x 60	Tidak terdefinisi	Good
	70 x 70	Tidak terdefinisi	Good
	80 x 80	Tidak terdefinisi	Good
	90 x 90	Tidak terdefinisi	Good
	100 x 100	Tidak terdefinisi	Good
	200 x 200	34,70	Poor picture
	300 x 300	23,13	Usunable
	400 x 400	20,50	Usunable

Tabel 2. Tabel Hasil pengujian kualitas citra skenario 2

Dimensi citra cover	Pesan	PSNR	Kriteria kualitas citra
13 KB (64 x 64)	Pesan 1	Tidak terdefinisi	Good
	Pesan 2	42,11	Reasonable
	Pesan 3	35,82	Poor picture
	Pesan 4	35,34	Poor picture
49 KB (128 x 128)	pesan 1	Tidak terdefinisi	Good
	pesan 2	Tidak terdefinisi	Good
	Pesan 3	Tidak terdefinisi	good
	Pesan 4	Tidak terdefinisi	Good
	Pesan 5	Tidak terdefinisi	Good
	pesan 6	39,09	Reasonable
	pesan 7	38,13	Reasonable
	Pesan 8	36,99	Poor picture
	pesan 9	35,34	Poor picture
	Pesan 10	33,98	Poor picture
193 KB (256 x 256)	pesan 1	Tidak terdefinisi	Good
	pesan 2	Tidak terdefinisi	Good
	Pesan 3	Tidak terdefinisi	Good
	Pesan 4	Tidak terdefinisi	Good
	Pesan 5	Tidak terdefinisi	Good
	pesan 6	48,13	Good
	pesan 7	48,13	Good
	Pesan 8	45,12	Reasonable
	pesan 9	45,12	Reasonable
	pesan 10	45,12	Reasonable
769 KB (512 x 512)	pesan 1	Tidak terdefinisi	Good
	pesan 2	Tidak terdefinisi	Good
	Pesan 3	Tidak terdefinisi	Good
	Pesan 4	Tidak terdefinisi	Good
	Pesan 5	Tidak terdefinisi	Good
	pesan 6	Tidak terdefinisi	Good
	pesan 7	Tidak terdefinisi	Good
	Pesan 8	Tidak terdefinisi	Good
	pesan 9	Tidak terdefinisi	Good
	Pesan 10	Tidak terdefinisi	Good

Dari hasil pengujian pada skenario 1 terlihat bahwa solusi yang dihasilkan oleh algoritma genetika tidak dipengaruhi oleh besarnya citra cover yang digunakan untuk menyisipkan file, akan tetapi lebih dipengaruhi oleh dimensi citra yang akan disisipkan. Pada pengujian skenario 1 pada penyisipan citra 64 x 64 dengan dimensi 40 x 40 dan 50 x 50 berhenti pada solusi 4094, pada citra 128 x 128 dengan dimensi 60 x 60 berhenti pada solusi 14463. Pada citra 256 x 256 dengan dimensi 200 x 200 sampai dengan 400 x 400 berhenti pada solusi 65533. hal ini disebabkan oleh nilai fitness yang dihasilkan oleh proses mutasi hingga generasi yang sudah ditetapkan tidak lagi mencapai nilai probabilitas mutasi yang diinginkan yaitu 0.05. akan tetapi secara keseluruhan hasil pengujian memperlihatkan bahwa solusi penempatan penyisipan file dengan algoritma genetika menghasilkan nilai PSNR yang baik dimana nilai PSNR yang terkecil didapatkan pada skenario 1 adalah 20,33

sampai dengan nilai PSNR yang tidak bisa didefinisikan sehingga nilai PSNR ini termasuk dalam kategori PSNR baik.

Pada pengujian skenario 2 pada penyisipan pesan ke-5 berhenti pada solusi 2512, hal ini disebabkan oleh nilai fitness yang dihasilkan oleh proses mutasi hingga generasi yang sudah ditetapkan tidak lagi mencapai nilai probabilitas mutasi yang diinginkan yaitu 0.05. akan tetapi secara keseluruhan hasil pengujian memperlihatkan bahwa solusi penempatan penyisipan file dengan algoritma genetika menghasilkan nilai PSNR yang baik dimana nilai PSNR yang terkecil didapatkan pada skenario 2 adalah 35,34 sampai dengan nilai PSNR yang tidak bisa didefinisikan sehingga nilai PSNR ini termasuk dalam kategori PSNR baik.

5. KESIMPULAN

- a. Algoritma Genetika dapat digunakan untuk pencarian solusi penempatan penyembunyian pesan pada citra.
- b. Hasil pengujian yang dilakukan pada penyisipan file pesan yaitu pesan dengan karakter pesan 1 (100 karakter), pesan 2 (1000 karakter), pesan 3 (2500 karakter), pesan 4 (3500 karakter), pesan 5 (4500 karakter), pesan 6 (5500 karakter), pesan 7 (6500 karakter), pesan 8 (7500 karakter), pesan 9 (8500 karakter), pesan 10 (9500 karakter) terlihat bahwa nilai PSNR terkecil yang dihasilkan adalah 35,15 dimana nilai ini masih dikategorikan sebagai nilai PSNR yang baik.
- c. Hasil pengujian yang dilakukan pada penyisipan file citra dengan dimensi 10 x 10, 20 x 20, 30 x 30, 40 x 40, 50 x 50, 60 x 60, 70 x 70, 80 x 80, 90 x 90, 100 x 100, 200 x 200, 300 x 300 serta 400 x 400 terlihat bahwa nilai PSNR terkecil yang dihasilkan adalah 20,33 dB dimana nilai ini masih dikategorikan sebagai nilai PSNR yang baik.
- d. Hasil pengujian kualitas citra pada skenario 1 dengan dimensi 64 x 64 (13 KB) maksimal citra yang disisipkan untuk menghasilkan kualitas citra yang baik berada pada dimensi 30 x 30, dimensi 128 x 128 (49 KB) maksimal citra yang disisipkan untuk menghasilkan kualitas citra yang baik berada pada dimensi 40 x 40, dimensi 256 x 256 (193 KB) maksimal citra yang disisipkan untuk menghasilkan kualitas citra yang baik berada pada dimensi 80 x 80, dan dimensi 512 x 512 maksimal citra yang disisipkan untuk menghasilkan kualitas citra yang baik berada pada dimensi 100 x 100.
- e. Hasil pengujian kualitas citra pada skenario 2 dengan dimensi 64 x 64 (13 KB) maksimal pesan yang disisipkan untuk menghasilkan kualitas citra yang baik berada pada pesan 4, dimensi 128 x 128 sampai dengan dimensi 512 x 512 secara keseluruhan masih menghasilkan kualitas citra yang baik

6. DAFTAR PUSTAKA

- Amriane Hidayat, dkk, 2012. *Desain dan Analisis Steganografi Plus Minus 1 Menggunakan Algoritma Genetika untuk Citra Digital*. Universitas Telkom.
- Aries Pratiarso, Mike Yuliana, dkk, 2012. *Analisa PSNR pada Steganografi menggunakan Spread Spectrum*. Politeknik Elektronika Negeri Surabaya.
- Hidayat Arif, 2009, *Implementasi Algoritma Genetika untuk Steganografi pada Citra JPEG*, ITS.
- Hazuar Nurwanto, 2015, *Implementasi Penyandian Dan Penyembunyian Pesan pada Citra menggunakan Algoritma RSA dan Modified LSB*. Amikom Yogyakarta
- Lifang yu, yao zhao, rongrong ni at al. 2008. *PMI Steganography in JPEG Images Using Genetic Algorithm*. New York: Springer.

Putri Alatas , 2009, *Implementasi Teknik Steganografi dengan metode LSB pada Citra digital*. Universitas Gunadarma.

Reinhart Candra, dkk. 2010. *Analisis dan Implementasi Algoritma Genetika pada Watermarking Citra Digital dalam DCT*.

Rinaldi Munir, 2006, *Pengantar Kriptografi*. Penerbit Informatika ,Bandung.

T. Sutojo, S.Si., M.Kom, dkk. *Kecerdasan Buatan*. Penerbit ANDI, Yogyakarta.

entin.lecturer.pens.ac.id/.../Bab%207%20Algoritma%20Genetika.pdf diakses : Agustus 2015)