IMPLEMENTASI *STEGANOGRAPHY* MENGGUNAKAN METODE *LEAST SIGNIFICANT BIT (LSB*) PADA *MOBILE PHONE* BERBASIS *SYMBIAN OS*

DIKDIK PRASETIA
10101823

ABSTRAK

Steganography adalah ilmu dan seni untuk menyembunyikan informasi pada suatu media sehingga para eavesdropper tidak menyadari adanya pesan rahasia dalam media tersebut.

Pada tugas akhir ini penulis akan mencoba membuat suatu aplikasi steganography citra digital pada mobile phone. Sehingga kita dapat saling berkomunikasi dan bertukar informasi rahasia dengan media citra pada mobile phone. Metode steganography yang akan digunakan adalah metode LSB (Least Significant Bit) dan OPAP (Optimal Pixel Adjustment Process).

Citra hasil proses steganography (stego-image) akan dianalisa baik dengan menggunakan penilaian obyektif, yaitu menggunakan PSNR (Peak Signal to Noise Ratio), untuk membandingkan kualitas citra yang dihasilkan kedua metode.

Kata Kunci : Steganography, LSB (Least Significant Bit), OPAP (Optimal Pixel Adjusment Process) , PSNR (Peak Signal to Noise Ratio).

ABSTRACT

Steganography is a science and art for hiding information on a media so the eavesdropper doesn't realize that actually there is a secret message on the media.

On this final assignment the writer will try to make a digital image steganography application on mobile phone, so we could communicate and exchange secret information within the image media on mobile phone. The steganography methods which will be use are LSB (Least Significant Bit) and OPAP (Optimal Pixel Adjustment Process).

The result image from the steganography process (stego-image) will be analyze with objective analysis, using PSNR (Peak Signal to Noise Ratio) and a subjective analysis to compare the image quality produced by both methods.

Keywords: Steganography, LSB (Least Significant Bit), OPAP (Optimal Pixel Adjusment Process) , PSNR (Peak Signal to Noise Ratio).

I. PENDAHULUAN

1.1. Latar Belakang

Teknologi komunikasi bergerak saat ini berkembang dengan sangat cepat. Selain komunikasi suara banyak fasilitas yang diberikan oleh operator komunikasi bergerak. Seperti pengiriman data atau informasi melalui *GPRS*, atau menggunakan *MMS*.

Tetapi seiring perkembangan teknologi komunikasi yang semakin pesat sehingga semakin banyak pula kebutuhan akan transfer data-data penting yang mungkin saja dirahasiakan melalui alat komunikasi seperti *mobile phone*. Hal ini sering kali dimanfaatkan oleh sebagian orang-orang yang tidak bertanggung jawab untuk menyadap data atau pesan penting tersebut diketahui oleh pihak ketiga dan disebarkan kepada orang lain. Oleh karena itu, agar tidak diketahui oleh pihak ketiga dari pesan (*message*) yang sedang didistribusikan/ditransmisikan, meningkat suatu kebutuhan akan proteksi kerahasiaan sebuah pesan dan autentifikasi data multimedia. Salah satu cara untuk memenuhi kebutuhan-kebutuhan akan proteksi kerahasiaan sebuah pesan dan autentifikasi data multimedia adalah dengan *steganography.* Karena *steganography* menyediakan layer proteksi pada pesan yang akan didistribusikan/ditransmisikan, yang mana pesan tersebut akan disisipkan pada media lain yang kemudian hasil dari

penyisipan ini yang akan ditransmisikan melalui jaringan.

Steganography adalah suatu teknik penyembunyian pesan sehingga kasat mata. Pada dunia multimedia, ada beberapa media digital yang dapat digunakan sebagai cover untuk menyembunyikan keberadaan sebuah informasi, yaitu: file gambar, file audio, dan lain-lain. Salah satu kriteria suatu teknik *steganography* dikatakan layak diterapkan, jika akurasi pesan yang diekstraksi dari media *cover* tersebut cukup baik.

Dengan asumsi teknologi fisik komunikasi data dan perkembangan protokol komunikasi data yang semakin maju, sehingga pentransferan citra digital menjadi sangat akurat, maka hal yang harus diperhatikan adalah proses penyisipan pesan pada citra digital tersebut. Proses penyisipan pesan ke dalam suatu citra digital memerlukan metode yang dapat mencapai tujuan utama yaitu mempertahankan kualitas citra hasil *steganography* supaya tidak jauh berbeda dengan citra asli dan keakuratan hasil ekstrak pesan yang disisipkan ke citra *cover*.

Salah satu metode penyisipan pesan ke media *cover* adalah dengan menggunakan metode *LSB* (*Least Significant Bit*) dan metode *LSB* dengan tambahan *OPAP (Optimal Pixel Adjustment Process)*. Dimana dengan menggunakan metode ini bit-bit dari pesan (*message*) akan disisipkan secara langsung ke media *cover*. Akibat dari pengubahan bit-bit tersebut tentu akan mengakibatkan perubahan dimana citra digital yang ditempeli pesan akan sedikit berbeda dengan citra aslinya. Oleh karena itu citra yang akan disisipi pesan sebaiknya melewati suatu proses yang mentoleransi perubahan intensitas warna pada citra, sehingga perbedaan intensitas warna antara citra asli dengan citra *steganography* menjadi kasat mata.

Dikarenakan banyak sekali perangkat lunak yang digunakan sebagai proteksi kerahasiaan suatu informasi baik pada dalam jaringan internet maupun jaringan transmisi lainnya maka penulis mencoba membuat "IMPLEMENTASI *STEGANOGRAPHY* MENGGUNAKAN METODE *Least Significant Bit (LSB)* PADA *MOBILE PHONE* BERBASIS *SYMBIAN OS"* yang nantinya mudah-mudahan bisa menjaga kerahasiaan suatu data rahasia dalam melakukan pengiriman informasi dengan menggunakan *mobile phone.*

1.2. Perumusan Masalah

Permasalahan yang dijadikan objek penelitian dan pengembangan tugas akhir ini dapat dilihat melalui poin-poin pertanyaan berikut ini :

1. Bagaimana mengimplementasikan teknik Steganography menggunakan metode LSB dan OPAP pada mobile phone ?
2. Bagaimana cara penyisipan pesan ke citra digital menggunakan metode yang digunakan ?
3. Bagaimana cara untuk mengurangi error yang terjadi akibat penyisipan pesan pada piksel citra untuk mengurangi perbedaan yang mencolok antara citra asli dengan citra hasil?
4. Bagaimana cara mendapatkan piksel-piksel yang telah disisipi pesan sehingga pesan dapat di-ekstrak kembali?

1.3. Tujuan Penelitian

Tujuan yang ingin dicapai adalah :

1. Merancang dan mengimplementasikan perangkat lunak steganography pada mobile phone sebagai tool untuk keamanan data.
2. Membandingkan kualitas citra yang dihasilkan oleh metode LSB dengan OPAP dan LSB tanpa menggunakan OPAP.
3. Menganalisa nilai PSNR yang dihasilkan oleh, metode LSB dan

OPAP Mengetahui kualitas citra-stego yaitu citra yang telah disisipi pesan dengan dilakukannya pembandingan antara citra asli dengan citra stego.

1.4 Batasan Masalah

Untuk menghindari meluasnya materi pembahasan tugas akhir ini, maka penulis membatasi permasalahan dalam tugas akhir ini hanya mencakup hal-hal berikut :

1. Handset yang digunakan untuk perancangan adalah Nokia dengan platform seri 60.
2. Data yang disisipkan berupa teks dengan ukuran maksimal 500 karakter.
3. Metoda steganography yang digunakan adalah metoda Least Significant Bit dan Optimal Pixel Adjustment Process.
4. Citra pembawa yang akan disisipi pesan merupakan citra bitmap 24-bit yang telah direscaling menjadi 176 x 132, sesuai dengan ukuran lebar citra yang ditampilkan oleh layar seri 60.

1.5 Metodologi Penelitian

Metode penelitian yang digunakan dalam pembuatan tugas akhir ini disusun langkah-langkah sebagai berikut :

Pengumpulan data

Studi pustaka.

Pada tahap ini, dilakukan studi kepustakaan yaitu dengan mengumpulkan data melalui buku-buku, jurnal ilmiah dan internet yang berhubungan dengan masalah yang akan dibahas.

Observasi

Untuk menguatkan penelitian maka dilakukan observasi pada beberapa perangkat lunak penerjemah bahasa yang sudah ada sebelumnya.

Pembuatan perangkat lunak

Pada tugas akhir ini digunakan metoda Waterfall Model. Dengan pendekatan sekuensial untuk pengembangan perangkat lunak dimulai pada tingkat sistem dan berkembang sampai dengan pengetesan dan perbaikan kesalahan. Berikut ini adalah gambar siklus pembangunan perangkat lunak dengan pendekatan waterfall model.

Rekayasa sistem adalah kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa subset dari kebutuhan ke perangkat lunak.

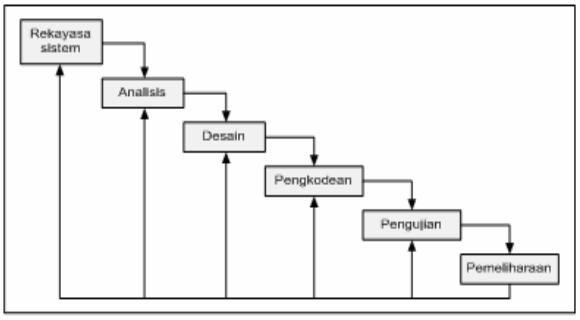
Analisis adalah proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak.

Desain adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda.

Pengkodean adalah desain harus diterjemahkan ke dalam bentuk bahasa mesin yang bisa dibaca.

Pengujian adalah proses yang berfokus pada logika internal perangkat lunak.

Pemeliharaan adalah perangkat lunak akan mengalami perubahan setelah disampaikan kepada pelanggan, pemeliharaan perangkat lunak mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.



Gambar 1.1 model penelitian waterfall

II. LANDASAN TEORI

2.1. *Steganography*

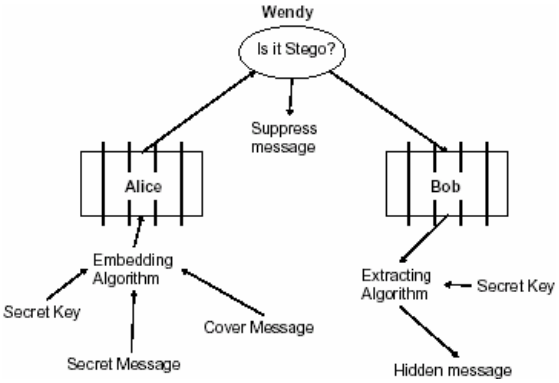
Steganography adalah suatu ilmu dari komunikasi “kasat mata”, teknik steganography berusaha untuk menyembunyikan keberadaan pesan itu sendiri dari pengamat. Pada steganography, tidak seperti bentuk komunikasi lainnya,

kesadaran seseorang akan terjadinya komunikasi antara pengirim dan penerima sangatlah penting. Oleh karena itu, kebutuhan pertama dari sistem steganography ini adalah "undetectable" (tidak terdeteksi). Dengan kata lain system steganography ini dianggap tidak aman, jika seseorang dapat membedakan cover-object dan stego-object[1]. Steganography berbeda dengan cryptography, Dengan menggunakan cryptography kita dapat menyembunyikan isi dari suatu pesan. Tapi masih terlihat jelas bahwa ada pesan yang ingin disampaikan. Tujuan dari steganography adalah untuk menyembunyikan keberadaan dari suatu komunikasi.

Formulasi steganography sering diistilahkan dengan istilah prisoner's problem[1] dimana Alice dan Bob adalah dua orang tahanan yang ingin berkomunikasi untuk merencanakan suatu pelarian. Akan tetapi, semua komunikasi diperiksa oleh sipir secara spesifik model dari model umum dari steganography. Diilustrasikan pada gambar 2.4. Alice ingin mengirimkan pesan rahasia m ke Bob. Untuk melakukan itu Alice menanamkan m kedalam cover-object c, dan mendapatkan stego-object s. kemudian strego-object s dikirimkan melalui saluran publik.

Cover-object : adalah objek yang dijadikan sebagai pembawa untuk menanamkan pesan rahasia. Banyak objek yang digunakan untuk ditanamkan pesan seperti : citra, suara, dan video.

Stego-object : adalah objek yang dibawa pesan rahasia. Dari Cover-object dan pesan yang ada, tujuan dari steganographer adalah untuk menghasilkan stego-object yang membawa suatu pesan.

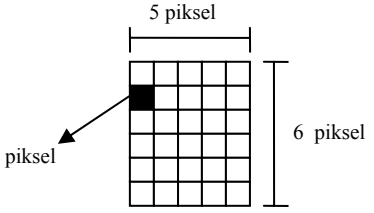


Gambar 2.4 ilustrasi steganography

2.2. Citra

Definisi citra [1], adalah suatu representasi, kemiripan, atau imitasi dari suatu obyek atau benda. Citra dapat dikelompokkan menjadi citra tampak dan citra tak tampak. Banyak contoh citra tampak dalam kehidupan sehari-hari: foto keluarga, lukisan, apa yang nampak di layar monitor dan televisi. Sedangkan citra tak tampak misalnya: data gambar dalam *file* (citra digital), dan citra yang direpresentasikan menjadi fungsi matematis. Hanya citra digital yang dapat diolah menggunakan komputer, jenis citra lain harus diubah dahulu menjadi citra digital, misalnya dengan dipindai (*scan*) dengan *scanner*.

Setiap citra digital memiliki beberapa karakteristik, antara lain ukuran citra, resolusi, dan format nilainya. Umumnya citra digital berbentuk persegi panjang yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau piksel sehingga ukuran citra selalu bulat.



Gambar 2.1 Ilustrasi piksel pada citra berukuran 5 x 6 piksel

Pada tugas akhir ini, citra yang digunakan adalah citra warna (*true color*) 24 bit. Pada citra warna, setiap titik

mempunyai warna spesifik yang merupakan kombinasi dari 3 warna dasar, yaitu: merah, hijau, dan biru. Format ini sering disebut sebagai RGB (*red-green-blue*). Setiap warna dasar mempunyai intensitas sendiri dengan nilai maksimum 255 (8 bit), misalnya warna kuning merupakan kombinasi warna merah dan warna hijau sehingga nilai RGB-nya adalah 255-255-0. Dengan demikian setiap titik pada citra warna membutuhkan data 3 byte.

Jumlah kombinasi warna yang mungkin untuk format citra ini adalah 2^{24} atau lebih dari 16 juta warna, dengan demikian bisa dianggap mencakup semua warna yang ada, inilah sebabnya format ini dinamakan *true color* [1].

2.3. Teori *Adaptive LSB*

Pada citra 24 bit, setiap piksel mempunyai 3 bytes untuk menentukan warna dari piksel. Ini berarti ada 1 bytes untuk setiap warna (merah,hijau,biru), jadi setiap warna mempunyai 256 level intensitas. Mata manusia tidak dapat melihat perbedaan antara level intesitas dengan intensitas level lain yang berdekatan (nilainya tidak jauh berbeda). Jadi jika kita mengubah *least significant bit* (LSB), bit yang paling tidak signifikan, manusia tidak akan melihat perbedaan. Ini dapat dimanfaatkan untuk menyembunyikan informasi pada citra bitmap.

Misal :

Pesan : **Jumlah Bit** : 3

Karakter "l" = 108 desimal (ASCII), 1 bytes
= 01101100 biner.

Piksel Asli:	**Piksel Baru :**
R = 10101010	R = 10101011
G = 10101010	G = 10101011
B = 10101010	B = 10101000

2.4 OPAP

Optimal Pixel Adjusment Process (OPAP)[4] digunakan untuk meningkatkan kualitas gambar dari *stego-image* yang didapat dari LSB. Misal p_i, p_i'dan p_i'' adalah nilai suatu channel dari piksel media pembawa, *stego-image* dari hasil LSB, dan perbaikan *stego-image* yang didapat setelah OPAP. Misal $\delta_i = p_i' - p_i$ adalah *embedding error* antara p_i dan p_i' . Menurut proses penanaman pada LSB, p_i' didapat dengan mengganti sejumlah k bit yang paling tidak signifikan dari p_i dengan k bit pesan. Oleh karena itu

$$-2^k < \delta_i < 2^k$$

nilai dari δ_i di segmentasikan menjadi 3 interval :

interval I : $2^{k-1} < \delta_i < 2^k$.
interval II : $-2^{k-1} \le \delta_i \le 2^{k-1}$.
Interval III : $-2^k < \delta_i < -2^{k-1}$.

Berdasarkan ketiga interval, OPAP yang dimana mengubah p_i' untuk membentuk *stego-pixel* p_i'' dapat dijelaskan sebagai berikut :

Case I **($2^{k-1} < \delta_i < 2^k$)**;
IF $p_i' \ge 2^k$, then $p_i'' = p_i' - 2^k$; ELSE $p_i'' = p_i'$
Case II **($-2^{k-1} \le \delta_i \le 2^{k-1}$)**;
$p_i'' = p_i'$
Case III **($-2^k < \delta_i < 2^{k-1}$)**;
IF $p_i' < 256 - 2^k$, then $p_i'' = p_i' + 2^k$;
ELSE $p_i'' = p_i$

Dengan menggunakan metode OPAP ini, embedding error yang dihasilkan oleh LSB dapat dikurangi.

2.5 Kriteria Penilaian Citra-Stego

Jika sebuah citra disisipkan dengan pesan (artinya, citra-stego), maka citra tersebut pasti mengalami error yang diakibatkan penyisipan pesan tersebut. Untuk menentukan tingkat kualitas suatu citra-stego antara citra sebelum dan sesudah disisipi pesan, maka diperlukan suatu kriteria yang dapat digunakan untuk menilai apakah citra-stego tersebut dapat menyembunyikan pesan yang dikandungnya sehingga kasat mata atau tidak. Maka adapun kriteria penilaian tersebut menggunakan kriteria Objektif yaitu yang didasarkan pada batas error yang diperbolehkan untuk citra yang diolah. Untuk citra asal f (x,y) dan citra yang telah disisipi pesan g(x,y) dengan ukuran yang sama M x N. parameter yang dijadikan sebagai kriteria penilaian obyektif pada tugas akhir ini adalah menggunakan nilai PSNR.

1. Peak Signal to Noise Ratio (PSNR)

   Nilai PSNR ini biasanya digunakan dalam satuan decibel (dB). Nilai 255 adalah nilai tertinggi intensitas suatu piksel.

$$\text{PSNR} = 10\,Log_{10}\left(\frac{255^2}{MSE}\right)$$

2. Mean Square Error (MSE)

Untuk citra hitam Black.bmp digunakan alat ukur Mean Square Error (MSE), dimana MSE merupakan nilai rata-rata dari jumlah kuadrat Absolute error antara citra asli dan citra-stego

$$\text{MSE}=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}|g(x,y)-f(x,y)|^2$$

III. ANALISIS

3.1. Analisis Sistem Steganography

Implementasi *steganography* ini merupakan salah satu *tool* dalam hal keamanan data. Sistem *steganography* yang dibuat terdiri dari dua buah sub sistem, yaitu: penyisipan (*embed*) dan pengekstrakan.

Sub sistem penyisipan berguna untuk melakukan penyisipan file-data ke file-citra, sehingga file-data tersebut kasat mata. *Input* yang diperlukan dalam penyisipan ini adalah file-citra, file-data, dan *stego-key*, sedangkan *output*-nya adalah citra-*stego*.

Sub sistem pengekstrakan berguna untuk melakukan pengekstrakan citra-*stego* untuk memperoleh file-data yang telah disisipkan ke dalam citra-*stego* tersebut. *Input* yang diperlukan adalah citra-*stego* dan *stego-key*, sedangkan *output*-nya adalah file-data.

3.2. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem pemenuhan *tool* pendukung yang dibutuhkan, berupa spesifikasi terhadap *hardware* dan *software* yang dibutuhkan selama pengembangan sistem.

3.2.1. Spesifikasi perangkat keras pendukung

Perangkat keras yang digunakan selama pengembangan sistem ini memiliki spesifikasi sebagai berikut :

a. *Processor* : AMD Athlon XP 1.8 GHz
b. *Memory* : 256 MB DDR
c. *Harddisk* : 80 GB 7200 rpm
d. *VGA* : 64 MB
e. Monitor : SVGA 15”
f. *Keyboard* dan *mouse*

3.2.2. Spesifikasi perangkat lunak pendukung

Perangkat lunak yang digunakan selama pengembangan sistem ini adalah sebagai berikut :

1 Sistem Operasi Windows XP Profesional
2 Bahasa pemrograman Microsoft Visual Studio .NET
3 ActivePerl-5.6.1.638-MSWin32-x86
4 J2re-1_3_1_13-windows-i586.exe
5 SDK seri 60 Developer platform 2.1 .NET.
6 Borland Delphi 7, Digunakan hanya untuk membuat tools pendukung kualitas citra.

3.3. Analisis Hasil

Dalam teknik *steganography* pada citra digital, kualitas citra hasil sangat menentukan keberhasilan dari teknik *steganography* tersebut. Oleh karena itu, pada tahap ini, ditentukan jenis pengukuran yaitu pengukuran secara obyektif.

Untuk citra asli f(x,y) dan citra yang telah disisipi pesan (citra-*stego*) g(x,y) dengan ukuran yang sama M x N, dimana x,y merupakan koordinat titik tertentu pada citra digital, maka parameter-parameter yang dijadikan sebagai kriteria penilaian obyektif adalah seperti dibawah ini [3]:

1. *Mean square error* (MSE)

   Merupakan nilai rata-rata dari jumlah kuadrat *Absolute Error* antara citra asli dengan citra-*stego*.

$$\text{MSE}=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}|g(x,y)-f(x,y)|^2$$

2. *Peak Signal to Noise Ratio* (PSNR)

Nilai PSNR ini biasanya digunakan dalam satuan *decibel* (dB). Nilai 255 adalah nilai tertinggi intensitas suatu piksel.

$$\text{PSNR} = 10\,Log_{10}\left(\frac{255^2}{MSE}\right)$$

Berdasarkan parameter-parameter penilaian citra diatas maka citra-*stego* dianalisis untuk menentukan sejauh mana pengaruh bit-bit pesan yang telah disisipkan tersebut terhadap kualitas citra-*stego*.

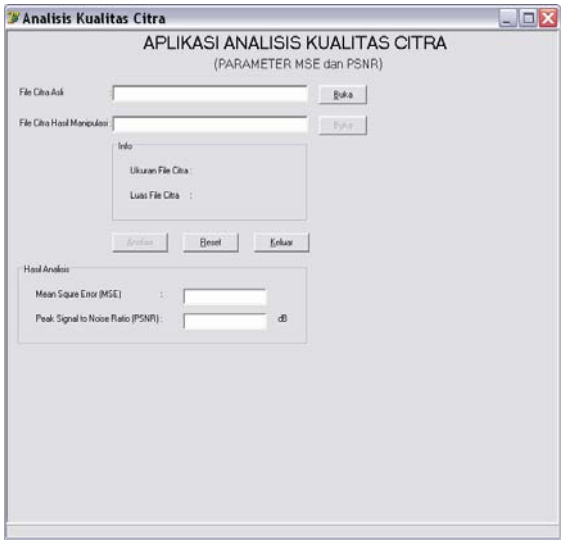
IV. IMPLEMENTASI

Tahap implementasi ini mengacu pada dua sub sistem utama pada sistem steganography, yaitu penyisipan dan pengekstrakan.

Tampilan pada program yang diimplementasikan dapat dilihat seperti gambar berikut ini:



Gambar 4.1 Form utama (penyisipan)



Gambar 4.2 Form Hasil Analisis

V. KESIMPULAN

Dari hasil analisis yang dilakukan, maka dapat diambil kesimpulan sebagai berikut :

1. Aplikasi Steganography pada Mobile Phone ini dapat melakukan penyisipan pesan ke dalam citra, dan melakukan ekstraksi pesan dari citra-stego
2. Dari hasil percobaan dan analisa yang telah dilaksanakan dan dibahas pada Bab IV , maka dapat disimpulkan beberapa hal sebagai berikut :

- Dalam analisa maupun dari hasil histogram ternyata tidak telalu terlihat ada perubahan dalam hasil penyisipan asli maupun citra-stego dikarenakan penyisipan hanya dapat memasukkan file teks dengan ukuran yang kecil.
- OPAP tidak dapat mengurangi embedding error yang dihasilkan oleh LSB, hanya pada kondisi dimana nilai piksel yang digunakan kurang dari 2k atau lebih dari 256 – 2k untuk citra 24 bit, dimana k adalah jumlah bit yang digunakan untuk menyisipkan pesan.

- OPAP akan mengurangi embedding error pada LSB, apabila jumlah bit yang digunakan lebih dari 1.

DAFTAR PUSTAKA

[1] S.S, Alan 2006. Tugas Akhir Implementasi Steganography pada Citra Digital dengan Menggunakan Metoda Adaptive LSB (Least significant Bits). Jurusan Teknik Informatika Universitas Komputer Indonesia

[2] Chan, Chi-Kwong, 2003, Hiding data in images by simple LSB substitution, City University of Hong Kong, Hong Kong

[3] Munir Rinaldi,Pengolahan Citra Digital Dengan Pendekatan Algoritmik, Informatika Bandung, 2004

[4] Nokia Corporation, Developer Platform 2.0 for Series 60: Designing C++ Applications; Version 1.0 October 2003.

[5] Nokia Corporation, Series 60 Developer Platform 1.0/2.0: Basics; Version 1.0, April 2, 2004.

.