Pengaruh Jumlah Pesan Pada Steganografi Citra Digital dengan Metode *Least Significant Bit (LSB)*

The Influence Amount of Messages on Digital Image Steganography Method of Least Significant Bit (LSB)

Fauzi Nur Iman
Mahasiswa Pasca Sarjana, Universitas Budi Luhur
Jl. Ciledug Raya, Petukangan Utara, Jakarta Selatan, Indonesia, 12260
undzieweb@gmail.com

Fery Updi
Mahasiswa Pasca Sarjana, Universitas Budi Luhur
Jl. Ciledug Raya, Petukangan Utara, Jakarta Selatan, Indonesia, 12260
feryupdi@gmail.com

Abstract— Steganography is a technique to hide the message in digital image in order to make no one else realizing that in that image there are secret message.In the application of steganography use LSB algorithm or Least Significant Bit. In this research we use of a file image PNG extension as a experiment and the insertion of message use LSB algorithm. We will insert message varying the number to cover image by using Matlab application. Research objectives is measuring the quality of the image uses the method PSNR (Peak Signal to Noise Ratio).

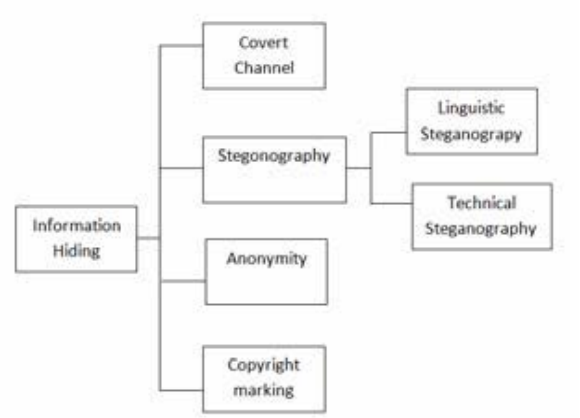
Keywords— Steganography; Matlab; LSB; PSNR***;***

Abstrak—Steganografi merupakan sebuah teknik menyembunyikan pesan dalam citra digital dengan tujuan agar orang lain tidak menyadari bahwa dalam citra tersebut terdapat pesan rahasia. Dalam penerapannya steganografi dapat menggunakan algoritma LSB atau Least Significant Bit. Dalam penelitian ini kami menggunakan file citra berekstensi PNG sebagai bahan percobaan dan penyisipan pesan menggunakan algoritma LSB. Kami akan menyisipkan pesan yang bervariasi jumlahnya terhadap cover image dengan menggunakan aplikasi MATLAB. Tujuan penelitian kami adalah menghitung kualitas penurunan citra menggunakan metode PSNR (Peak Signal to Noise Ratio).

Kata Kunci— Steganografi; Matlab; LSB; PSNR***;***

I. PENDAHULUAN

Untuk menjaga dalam pengirim informasi, diperlukan sebuah pengamanan agar informasi tersebut masih terjaga kerahasiaannya. Dan dibutuhkannya suatu pengenkripsi data pada saat pengiriman sehingga informasi tersebut hanya bisa dibuka oleh penerima. Akan tetapi enkripsi ini masih sangat mudah ditebak bahwa file atau pesan yang dikirim ini rahasia, karena dari pola filenya sudah tidak beraturan.



Gambar 1. Klasifikasi dari teknik penyembunyian informasi

Berbeda dengan steganography, pada saat pengiriman pesan tidak nampak sama sekali bahwa baik gambar, video, voice, ini ada sebuah pesan yang tersembunyi didalamnya, di karenakan adanya proses steganography.

II. PENELITIAN TERDAHULU

Penelitian terdahulu dilakukan oleh Ghazali Moenandar Male, Wirawan, Eko Setijadi (2012) dengan judul Analisa Kualitas Citra Pada Steganografi untuk Aplikasi E-Government. Pada penelitiannya mereka menjelaskan tentang penyisipan file doc pada gambar yang digunakan pada aplikasi E-Government. Mereka menggunakan sebuah file doc yaitu file dokumen dengan 1 ukuran dan cover image dengan ukuran yang beragam. Pada percobaan kedua mereka melakukan pengujian pada file text dengan ukuran file yang beragam dan di lakukan steganografi pada 1 file citra digital. Dari penelitian tersebut kami ingin melanjutkan penelitian serupa namun kami melakukan penelitian dengan acuan jumlah karakter pada file teks dan pada file citra grayscale.[3]

III. LANDASAN TEORI

A. *Steganografi*

Steganography adalah ilmu dan seni menulis atau menyembunyikan pesan ke dalam sebuah media sedemikian rupa sehingga keberadaan pesan tidak diketahui atau tidak disadari oleh orang selain pengirim dan penerima pesan tersebut. Kata steganography berasal dari bahasa Yunani, yaitu "steganos" yang berarti tersembunyi atau terselubung dan "graphein" yang berarti menulis.

Steganography membutuhkan dua aspek yaitu media penyimpan dan informasi rahasia yang akan disembunyikan. Metode *steganography* sangat berguna jika digunakan pada *steganography* komputer karena banyak format *file digital* yang dapat dijadikan media untuk menyembunyikan pesan. *Steganography digital* menggunakan media *digital* sebagai wadah penampung, misalnya teks, citra, suara, dan *video*. Data rahasia yang disembunyikan juga dapat berupa teks, citra, suara, atau *video*.

Steganography memanfaatkan kekurangan-kekurangan sistem indera manusia seperti mata (*Human Visual System*) dan telinga (*Human Auditory System*), sehingga tidak diketahui kehadirannya oleh indera manusia (indera penglihatan atau indera pendengaran) dan mampu menghadapi proses-proses pengolahan sinyal *digital* dengan tidak merusak kualitas data yang telah disisipi sampai pada tahap tertentu. Terdapat tiga aspek yang perlu diperhatikan dalam menyembunyikan pesan: kapasitas, keamanan, dan ketahanan. Kapasitas merujuk kepada besarnya informasi yang dapat disembunyikan oleh media, keamanan merujuk kepada ketidakmampuan pihak lain untuk mendeteksi keberadaan informasi yang disembunyikan, dan ketahanan merujuk kepada sejauh mana medium *steganography* dapat bertahan sebelum pihak lain menghancurkan informasi yang disembunyikan.[1]

B. *Teknik Steganografi*

Pada dasarnya, terdapat tujuh teknik yang digunakan dalam steganografi.[2]

1. Injection, merupakan suatu teknik menanamkan pesan rahasia secara langsung ke suatu media. Salah satu masalah dari teknik ini adalah ukuran media yang diinjeksi menjadi lebih besar dari ukuran normalnya sehingga mudah dideteksi. Teknik itu sering juga disebut Embedding.
2. Substitusi, data normal digantikan dengan data rahasia. Biasanya, hasil teknik itu tidak terlalu mengubah ukuran data asli, tetapi tergantung pada file media dan data yang akan disembunyikan. Teknik subsitusikan bisa menurunkan kualitas media media yang ditumpangi.
3. Domain Transform, teknik pada ranah transform memfokuskan penyisipan pesan ke dalam frekuensi dari *cover-file*. Salah satu metode yang bekerja dalam *domain transform* adalah *Discrete Wavelet Transform (DWT)*. Steganografi memiliki dua buah proses, yaitu penyisipan dan ekstraksi pesan. Proses penyisipan pesan pada steganografi membutuhkan dua buah masukan, yaitu pesan yang ingin disembunyikan dan media penyisipan. Hasil dari proses ini disebut dengan *stego-object*, yaitu suatu media yang mempunyai kemiripan dengan media penyisipan yang telah terdapat pesan tersembunyi di dalamnya.
4. Spread Spectrum, sebuah teknik pentransmisian menggunakan *pseudo-noise code*, yang independen terhadap data informasi sebagai modulator bentuk gelombang untuk menyebarkan energy sinyal dalam sebuah jalur gelombang untuk menyebarkan energi sinyal dalam sebuah jalur komunikasi (*bandwitdth)* yang lebih besar dari pada sinyal jalur komunikasi informasi. Oleh penerima, sinyal dikumpulkan kembali menggunakan replika *pseudo-noise code* tersinkronisasi.
5. Statisikal Method, teknik ini disebut juga skema steganographic 1 bit. Skema tersebut menanamkan satu bit informasi pada media tumpangan dan mengubah statistik walaupun hanya 1 bit Perubahan statistik ditunjukkan dengan *indikasi* 1 dan jika tidak ada perubahan, terlihat indikasi 0. Sistem ini bekerja berdasarkan kemampuan penerima dalam membedakan antara informasi yang dimodifikasi dan yang belum.
6. Distortion, metode ini menciptakan perubahan atas benda yang ditumpangi oleh data rahasia.
7. Cover Generation, metode ini lebih unik dari pada metode lainnya karena cover object yang dipilih untuk menyembunyikan pesan. Contoh dari metode ini adalah spam mimic

C. *Kriteria Steganografi*

Penyembunyian data rahasia ke dalam citra digital akan mengubah kualitas citra tersebut. Kriteria yang harus diperhatikan dalam penyembunyian data adalah [2].

1. Fidelity, mutu citra penampung tidak jauh berubah. Setelah penambahan data rahasia, citra hasil steganografi masih dapat terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam citra tersebut terdapat pesan rahasia.
2. Robustness, data yang disembunyikan harus tahan (*robust*) terhadap berbagai operasi manipulasi yang dilakukan terhadap citra penampung.
3. Recovery, data yang disembunyikan harus dapat diungkapkan kembali (*reveal)*. Karena tujuan dari steganografi adalah penyembunyian data, maka sewaktu- waktu data rahasia di dalam citra penampung harus dapat diambil kembali untuk digunakan lebih lanjut.

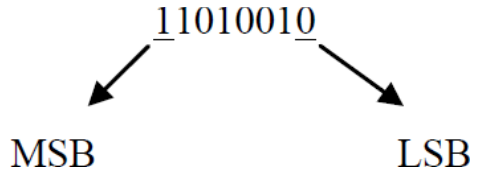
D. *Least Significant Bit*

Least *Significant Bit (LSB)* adalah cara paling umum untuk menyembunyikan pesan. *LSB* dilakukan dengan memodifikasi bit-bit yang termasuk bit *LSB* pada setiap *byte* warna pada sebuah piksel. Bit-bit *LSB* ini akan dimodifikasi dengan menggantikan setiap *LSB* yang ada dengan bit-bit pesan rahasia yang ingin disembunyikan. Setelah semua bit pesan rahasia menggantikan bit *LSB* file tersebut, maka pesan rahasia telah berhasil disembunyikan.

Metode ini membutuhkan syarat, yaitu jika dilakukan kompresi pada *file stego*, harus digunakan format *lossless compression*. Hal itu dikarenakan metode ini menggunakan

bit-bit pada setiap piksel pada *image*. Jika digunakan format *lossy compression*, pesan rahasia yang disembunyikan dapat hilang.

Contoh penggunaan *LSB*, sebuah susunan bit pada sebuah *byte:*



(MSB = Most Significant *Bit, LSB = Least Significant Bit)*

Gambar 2. Pola MSB dan LSB

Bit yang sesuai untuk ditukar adalah bit *LSB* karena perubahan pada daerah tersebut hanya akan menyebabkan nilai *byte* menjadi lebih tinggi 1 angka atau lebih rendah 1 angka dari nilai sebelumnya.

Untuk memperkuat teknik penyembunyian data, bit–bit data rahasia tidak digunakan mengganti *byte* yang berurutan namun dipilih susunan *byte* yang acak. Misalnya, jika terdapat 50 *byte* dan 6 bit data yang akan disembunyikan maka *byte* yang diganti bit *LSB*-nya dipilih secara acak, misalkan *byte* nomor 36, 5, 21, 10, 18, 49.

Bilangan acak dibangkitkan dengan *pseudo-random-number-generator* (*PRNG*) *cryptography*. *PRNG cryptography* merupakan algoritma *cryptography* yang digunakan untuk enkripsi dan dibangun dengan menggunakan algoritma *DES* (*Data Encryption Standard*). Misalkan segmen dari data sebelum ditukar adalah:

00110011 10100010 11100010 01101111

Setelah data ‘0110’ disembunyikan, segmen menjadi:

00110010 10100011 11100011 01101110

Jika digunakan *image* 24 bit *color* sebagai *cover*, sebuah bit dari masing-masing komponen *Red*, *Green*, dan *Blue*; dapat digunakan sehingga 3 bit dapat disimpan pada setiap piksel. Sebuah image 800 x 600 piksel dapat digunakan untuk menyembunyikan 1.440.000 bit (180.000 *bytes*) data rahasia. Misalnya, terdapat 3 piksel dari *image* 24 bit *color* :

(00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

Jika diinginkan untuk menyembunyikan karakter A dengan nilai biner 10000001 dihasilkan :

(00100111 11101000 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

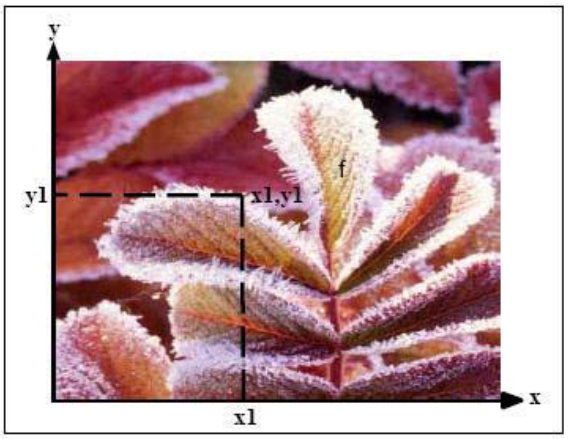
Dapat dilihat bahwa hanya 3 bit saja yang perlu diubah untuk menyembunyikan karakter A ini. Perubahan pada *LSB* ini akan terlalu kecil untuk terdeteksi oleh mata manusia sehingga pesan dapat disembunyikan secara efektif.

Jika digunakan *image* 8 *bit color* sebagai *cover*, hanya 1 bit saja dari setiap piksel warna yang dapat dimodifikasi sehingga pemilihan *image* harus dilakukan dengan sangat hati-hati, karena perubahan *LSB* dapat menyebabkan terjadinya perubahan warna yang ditampilkan pada *image.* Akan lebih baik jika berupa *grayscale* karena perubahan warnanya akan lebih sulit dideteksi oleh mata manusia.

Proses ekstraksi pesan *dapat* dengan mudah dilakukan dengan mengekstrak *LSB* dari masing-masing piksel pada *stego file* secara berurutan dan menuliskannya ke *output file* yang akan berisi pesan tersebut.[3][6]

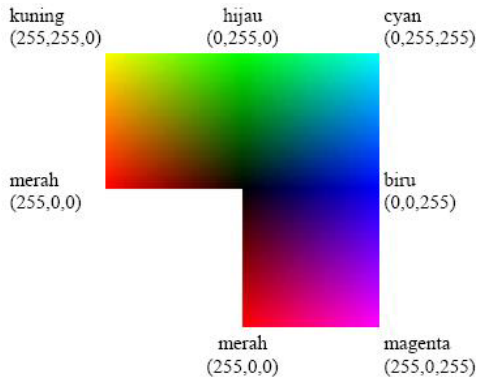
E. Citra Digital

Citra digital dapat didefinisikan sebagai fungsi dua variabel, f(x,y), dimana x dan y adalah koordinat spasial dan nilai f(x,y) adalah intensitas citra pada koordinat tersebut, hal tersebut diilustrasikan pada gambar dibawah ini. Teknologi dasar untuk menciptakan dan menampilkan warna pada citra digital berdasarkan pada penelitian bahwa sebuah warna merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau, dan biru (Red, Green, Blue - RGB).[5]



Gambar 3. Ilustrasi Citra Digital

RGB adalah suatu model warna yang terdiri dari merah, hijau, dan biru, digabungkan dalam membentuk suatu susunan warna yang luas. Setiap warna dasar, misalnya merah, dapat diberi rentang-nilai. Untuk monitor komputer, nilai rentangnya paling kecil = 0 dan paling besar = 255. Pilihan skala 256 ini didasarkan pada cara mengungkap 8 digit bilangan biner yang digunakan oleh mesin komputer. Dengan cara ini, akan diperoleh warna campuran sebanyak 256 x 256 x 256 = 1677726 jenis warna. Sebuah jenis warna, dapat dibayangkan sebagai sebuah vektor di ruang 3 dimensi yang biasanya dipakai dalam matematika, koordinatnya dinyatakan dalam bentuk tiga bilangan, yaitu komponen-*x*, komponen-*y* dan komponen-*z*. Misalkan sebuah vektor dituliskan sebagai **r** = (x,y,z). Untuk warna, komponen-komponen tersebut digantikan oleh komponen R(ed), G(reen), B(lue). Jadi, sebuah jenis warna dapat dituliskan sebagai berikut: warna = RGB(30, 75, 255). Putih = RGB (255,255,255), sedangkan untuk hitam= RGB(0,0,0).



Gambar 4. Pewarnaan dalam RGB

F. Biner

Sistem bilangan biner atau sistem bilangan basis dua adalah sebuah sistem penulisan angka dengan menggunakan dua simbol yaitu 0 dan 1. Sistem bilangan biner modern ditemukan oleh Gottfried Wilhelm Leibniz pada abad ke-17. Sistem bilangan ini merupakan dasar dari semua sistem bilangan berbasis digital. Dari sistem biner, kita dapat mengkonversinya ke sistem bilangan *Oktal* atau *Hexadesimal*. Sistem ini juga dapat kita sebut dengan istilah bit, atau *Binary Digit*. Pengelompokan biner dalam komputer selalu berjumlah 8, dengan istilah 1 Byte. Dalam istilah komputer, 1 Byte = 8 bit. Kode-kode rancang bangun komputer, seperti ASCII, *American Standard Code for Information Interchange* menggunakan sistem peng-*kode*-an 1 Byte.

G. Bit

Bit digit sistem angka biner satuan teori komputasi informasi digital. Teori informasi juga sering merujuk pada sebuah dalam (basis 2). Sebagai contoh, angka 1001011 memiliki panjang 7 bit. Digit biner hampir selalu digunakan sebagai terkecil dalam penyimpanan dan komunikasi informasi di dalam dan menggunakan digit natural, disebut komputasi kuantum qubit, sebuah potongan informasi dengan kemungkinan informasi tersebut bernilai benar. *nit* atau *nat*. Sementara,menggunakan satuan Bit juga digunakan sebagai satuan ukuran, yaitu kapasitas informasi dari sebuah digit biner. Lambang yang digunakan adalah bit, dan kadang-kadang (secara tidak resmi) b (contohnya, modem dengan kecepatan 56 kbps atau 56 kilo bit per second/detik). Satuan ini dikenal juga sebagai shannon, dengan lambang Sh.

H. Byte

Bita Bahasa Inggris: penyimpanan komputer. Satu bita terdiri dari delapan bit. (*Byte*) adalah istilah yang biasa dipergunakan sebagai satuan dari data dalam Huruf Cakram keras_B digunakan dalam singkatan kepada Byte. (bit menggunakan singkatan b.) seperti kB = kilobita. (*hard disk*) berkapasitas 40GB secara mudahnya bermaksud cakram keras tersebut mampu menyimpan hingga 40 ribu juta (milyar) bita atau gigabita data.

I. Piksel

Piksel inci. adalah unsur gambar atau representasi sebuah titik terkecil dalam sebuah gambar grafis yang dihitung per inci. Piksel sendiri berasal dari akronim bahasa Inggris resolusi, mesin cetak gambar berwarna dapat menghasilkan hasil cetak yang memiliki lebih dari 2.500 titik per *Picture Element* yang disingkat menjadi *Pixel*. Pada ujung tertinggi skala resolusi. mesin cetak gambar berwarna dapat menghasilkan hasil cetak yang memiliki lebih dari 2.500 titik perinci dengan pilihan 16 juta warna lebih untuk setiap inci, dalam istilah komputer berarti gambar seluas satu inci persegi yang bisa ditampilkan pada tingkat resolusi tersebut sepadan dengan 150 juta bit informasi.

Monitor atau layar datar yang sering kita temui terdiri dari ribuan piksel yang terbagi dalam baris-baris dan kolom-kolom. Jumlah piksel yang terdapat dalam sebuah monitor dapat kita ketahui dari resolusinya. Resolusi maksimum yang disediakan oleh monitor adalah 1024x768, maka jumlah piksel yang ada dalam layar monitor tersebut adalah 786432 piksel. Semakin tinggi jumlah piksel yang tersedia dalam monitor, semakin tajam gambar yang mampu ditampilkan oleh monitor tersebut.

J. Model Warna Red Green Blue (RGB)

Model warna *RGB* adalah sebuah model warna tambahan dalam jenis merah, hijau, dan biru muda yang ditambahkan secara bersama dalam berbagai cara untuk memproduksi sebuah kesatuan warna secara luas. Nama dari model ini berasal dari inisial ketiga zat warna primer, yaitu *Red* (merah), *Green* (hijau), dan *Blue* (biru).

Tujuan utama model warna *RGB* adalah untuk menyajikan, dan menampilkan gambar di dalam sistem elektronik, seperti televisi dan komputer, dan digunakan pula pada fotografi konvensional. Sebelum zaman elektronik, model warna *RGB* telah mempunyai suatu teori yang kuat di belakang itu, yang didasarkan persepsi manusia terhadap warna.

Tipe alat yang menggunakan input *RGB* adalah televisi, kamera video, *scanner*, dan kamera digital. Tipe alat yang menggunakanan output *RGB* adalah televisi satuan dengan berbagai teknologi (*CRT*, *LCD*, plasma), komputer, dan layar telepon genggam, proyektor video, dan layar besar seperti Jumbotron, dan lain-lain. Warna *printer*, bukanlah *RGB*, tetapi warna *subtractive* (model warna *CMYK*).

IV. ANALISA DAN PERANCANGAN

A. Sistem Steganografi

Sistem steganografi yang dibahas akan di fokuskan kepada bagaimana cara membangun suatu sistem steganografi pada citra digital file gambar yang efisien dan untuk mengeksploitasi keterbatasan sistem penglihatan manusia dengan cara menurunkan kualitas warna pada file gambar yang disisipi pesan rahasia. Sehingga dengan keterbatasan tersebut manusia sulit menemukan gradasi penurunan kualitas warna file gambar yang telah disisipi informasi atau pesan rahasia. Sistem ini terdiri dari dua buah sub sistem yaitu : sistem penyisipan informasi atau pesan dan sistem pengekstrakkan informasi atau pesan.

Sistem penyisipan informasi atau pesan berfungsi untuk melakukan proses penyembunyian pesan ke file citra digital

gambar. Komponen dari sistem penyisipan ini yaitu terdapat komponen untuk menuliskan pesan yang dipakai untuk menempatkan penulisan pesan rahasia.

Sistem pengekstrakkan informasi atau pesan berfungsi untuk melakukan pengekstrakkan file untuk memeperoleh pesan yang telah disisipkan ke dalam file citra digitl gambar tersebut. Komponen pada sistem pengekstrakkan ini terdapat komponen untuk membaca pesan yang digunakan untuk menempatkan pesan rahasia yang akan dibaca, sehingga keluarannya adalah informasi atau pesan rahasia dari file citra digital gambar.

B. Rancangan Algoritma LSB pada citra digital

Secara garis besar jalannya aplikasi ini adalah terbagi dua proses utama yaitu hide message atau penyisipan pesan dan extract message atau pendekteksian kembali pesan yang tersembunyi.

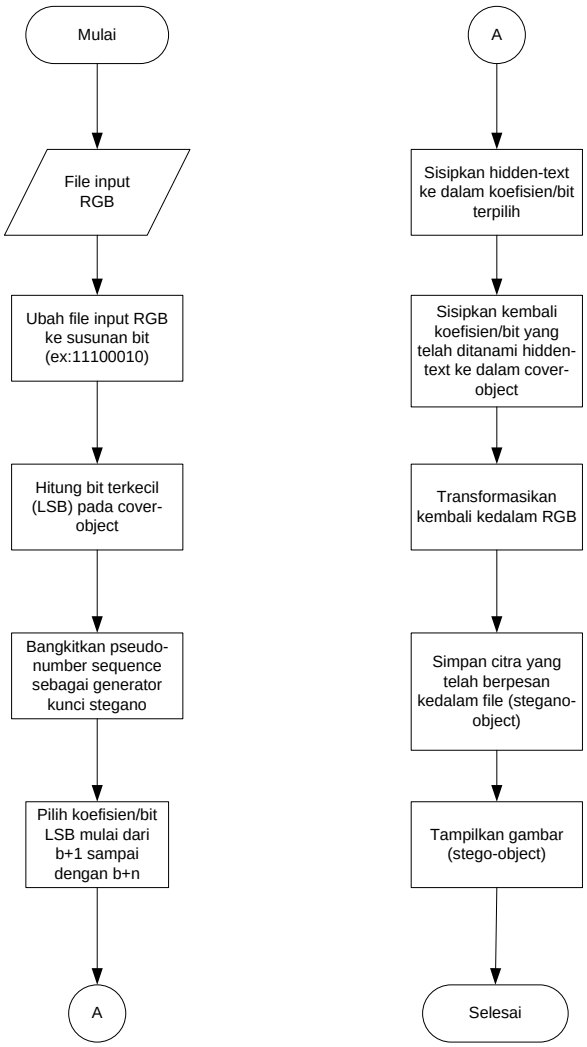
Pada proses penyisipan pesan (embedding message) dimulai dengan memilih gambar yang akan dijadikan cover object untuk menyisipkan dan menyembunyikan pesan ke dalam gambar kemudian menentukan key file yang akan digunakan sebagai password dalam proses extract dan menuliskan isi pesan text yang akan disisipkan kedalam gambar.

Sedangkan pada proses pendeteksian pesan (extraction message) dimulai dengan memilih file gambar atau covert object yang akan akan di extract dan memasukan key file, yang hasil y ekstraksi pesannya dapat disimpan pada satu file tertentu yang dipilih.

C. Diagram alir proses embedding message

Pada gambar 3.1 dibawah adalah flowchart proses embedding message kedalam file citra (cover-object) dimulai dengan membaca file citra ke RGB, Seperti kita ketahui untuk file bitmap 24 bit maka setiap pixel (titik) pada gambar tersebut terdiri dari susunan tiga warna merah, hijau dan biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Setelah membaca pixel dari file citra langkah selanjutnya menentukan bit terkecil (LSB) pada cover-object.

Setelah menentukan bit terkecil dari cover-object yang akan digunakan maka langkah selanjutnya yaitu membangkitkan pseudo-number sequence yang akan digunakan sebagai generator kunci stegano dengan menentukan key yang akan dipakai sebagai password untuk mengenkripsi pesan kedalam cover-object. Proses selanjutnya adalah memilih koefisien bit terpilih mulai dari b+1 sampai b+n untuk disisipkan hiddentext kedalamnya,

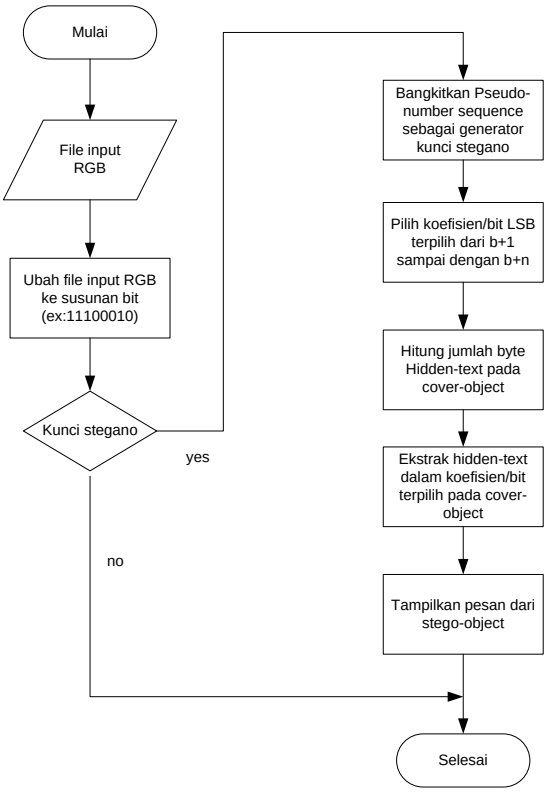


Gambar 3.1 Flowchart Embedding Message (encoder)

Selanjutnya adalah setelah memilih koefisien atau bit-bit terpilih maka proses berikutnya adalah menyisipkan hidden-text ke dalamnya koefisien atau bit-bit tersebut sehingga akan dihasilkan koefisien atau bit-bit yang baru yang telah mengandung pesan,dan menyisipkannya kembali kedalam cover-object, yang kemudian koefisien tersebut selanjutnya akan di transformasikan kembali kedalam nilai RGB yang baru dan menyimpan citra yang telah berpesan ke dalam cover-object sehingga diperoleh atau dapat ditampilkan sebuah gambar baru yang telah disisipkan pesan atau stego-object.

D. Diagram alir proses extraction message

Pada gambar 3.2 dibawah adalah flowchart proses extraction message dari stegoobject menghasilkan hidden-text yang terdapat didalamnya atau untuk mengungkap kembali pesan yang disisipkan kedalam file citra, proses awalnya dimulai dengan membaca file citra ke RGB, dan mengubah file input RGB kedalam format biner.



Gambar 3.2. Flowchart Extraction Message (decoder)

Kemudian langkah selanjutnya adalah memeriksa kunci stegano yang digunakan sebagai password saat mengenkripsi pesan, jika kunci stegano yang dimasukkan benar maka akan beralih ke proses selanjutnya yakni membangkitkan nilai PRNG atau pseudo number gernerator yang menyimpan bit-bit atau koefisien terpilih yang secara acak berada pada file citra atau stego-object. Setelah diperoleh koefiesien atau bit-bit yang terpilih yang mengndung pesan maka proses ekstraksi akan berjalan dan menghitung jumlah byte hidden-text pada cover-object. Setelah diperoleh byte yang tersembunyi pada cover-onject maka proses berikutnya adalah mengekstrak kembali pesan yang tersembunyi (hidden-text) yang terdapat didalamnya sehingga pesan dapat ditampilkan kembali.

E. Peak Signal to Noise Ratio (PNSR)

Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan desibel. Pada penelitian ini, PSNR digunakan untuk mengetahui perbandingan kualitas citra sebelum dan sesudah disisipkan pesan.

Untuk menentukan PSNR, terlebih dahulu harus ditentukan nilai rata-rata kuadrat dari error (MSE - *Mean Square Error*).

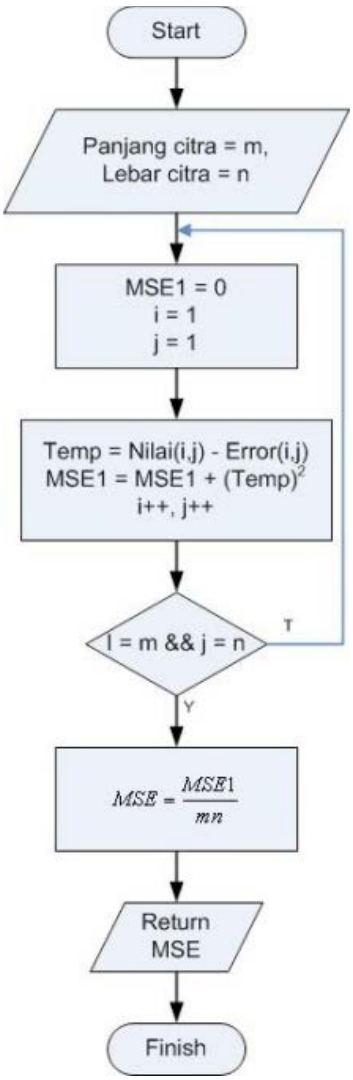
Perhitungan MSE adalah sebagai berikut :

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

Dimana (1):

MSE = Nilai *Mean Square Error* dari citra tersebut
m = panjang citra tersebut (dalam piksel)
n = lebar citra tersebut (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

Dari rumus di atas, dapat dibuat diagram alir perhitungan MSE seperti ditunjukkan pada gambar 3.3 di bawah ini :



Gambar 3.3 Flowchart Perhitungan MSE

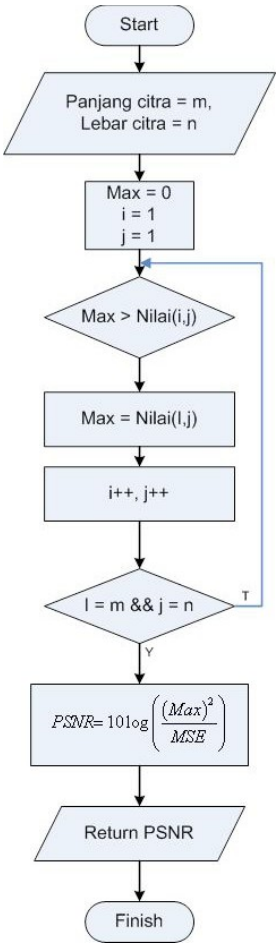
Sementara nilai PSNR dihitung dari kuadrat nilai maksimum sinyal dibagi dengan MSE. Apabila diinginkan PSNR dalam desibel, maka nilai PSNR akan menjadi sebagai berikut :

$$PSNR = 10 \cdot \log\left(\frac{MAX_I^2}{MSE}\right) = 20 \cdot \log\left(\frac{MAX_I}{\sqrt{MSE}}\right) \quad (2)$$

Dimana (2) :

PSNR = nilai PSNR citra (dalam dB)
MAXi = nilai maksimum piksel
MSE = nilai MSE

Dari rumus perhitungan PSNR di atas, dapat dibuat diagram alir untuk menghitung PSNR seperti ditunjukkan pada gambar 3.4 di bawah ini.



Gambar 3.4 Flowchart Perhitungan PSNR

Dari flowchart PSNR di atas, maka berikut ini adalah penerapan algoritma PSNR yang dipakai pada penulisan ini. Algoritma PSNR yang digunakan dibuat menggunakan bahasa pemrograman matlab mengingat matlab merupakan bahasa pemrograman sangat baik untuk mengolah file citra karena dilengkapi fungsi-fungsi yang memudahkan pemakaiannya. Dibawah ini merupakan program yang dignakan untuk mengetahui nilai PSNR dari setiap file citra sebelum dan sesudah disisipkan pesan.[4]

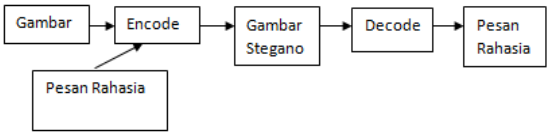
F. Perbandingan kualitas citra dengan PSNR

Dari proses penyisipan pesan ke dalam file citra tentunya akan ada perbedaan kualitas citra sebelum dan sesudah proses penyisipan pesan, untuk mengetahui seberapa besar penurunan kualitas citra maka akan dilakukan perhitungan nilai PSNR seperti yang telah dijelaskan pada bab sebelumnya.

Berikut ini akan dilakukan pengujian terhadap enam buah citra uji yang telah berisikan pesan tersembunyi melalui aplikasi steganography yang telah dibuat. Keenam citra uji ini akan disisipkan sejumlah karakter dengan jumlah yang bervariasi mulai dari 100 karakter hingga 600 karakter, hal ini dimaksudkan untuk mengetahui seberapa besar perubahan yang terjadi pada citra uji yang diukur dengan besarnya perubahan nilai PSNR dari setiap citra uji tersebut.

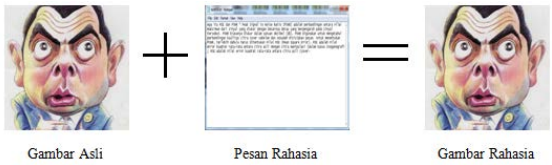
V. ANALISA DAN PENGUJIAN

A. Proses penyisipan pesan kedalam gambar



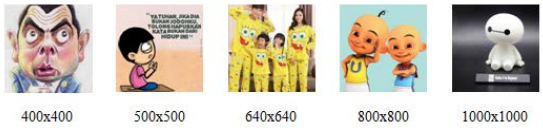
Gambar 5. Proses penyisipan pesan kedalam gambar

Alur penyisipan pesan adalah kita menyiapkan gambar dan pesan dalam bentuk file text kemudian di encode menjadi sebuah gambar stegano. Gambar stegano disini memiliki bentuk ukuran dan resolusi yang sama dengan gambar asli, kita tidak bisa membedakan mana gambar yang asli dan mana gambar stegano dengan mata biasa. Kemudian untuk membaca pesan yang ada pada gambar kita lakukan proses decode untuk meng-ekstrak pesan yang ada pada gambar.



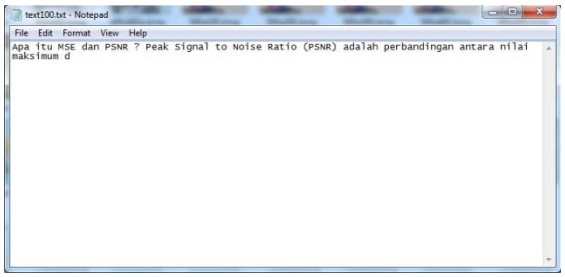
Gambar 6. Steganografi

Dalam percobaan ini kami akan memasukkan pesan kedalam 5 buah gambar yang berbeda ukuran, resolusi, namun semuanya memiliki format gambar yang sama yakni 24bit – bitmap (.bmp).

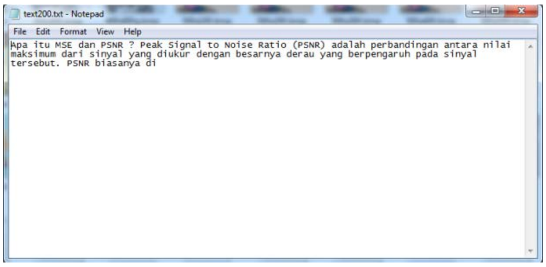


Gambar 7. Citra yang akan dilakukan percobaan Steganografi

Gambar – gambar diatas akan dilakukan penyisipan pesan dengan jumlah pesan yang bervariasi antara 100, 200, 300, 400, 500 dan 600 pesan. Setiap gambar akan dilakukan 6 kali penyisipan dengan jumlah pesan yang telah disebutkan sebelumnya. Kemudian berikut adalah screenshot dari pesan yang akan diinputkan, terdiri dari 100, 200, 300, 400, 500 dan 600 karakter. Dan semuanya merupakan file teks berekstensi .txt.



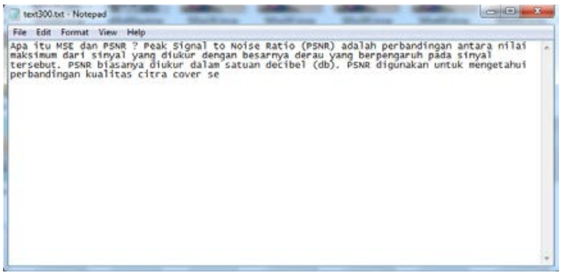
Gambar 8. Pesan rahasia 100 karakter


text200.txt - Notepad

File Edit Format View Help

Apa itu MSE dan PSNR ? Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya di

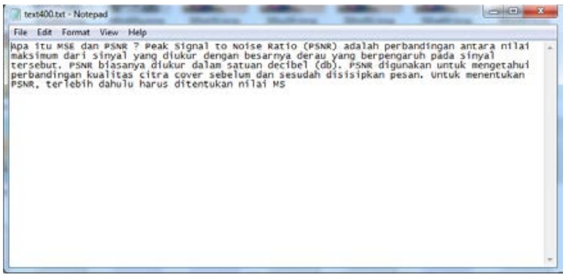
Gambar 9. Pesan rahasia 200 karakter


text300.txt - Notepad

File Edit Format View Help

Apa itu MSE dan PSNR ? Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan decibel (db). PSNR digunakan untuk mengetahui perbandingan kualitas citra cover se

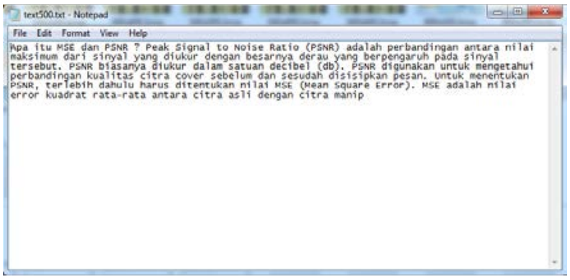
Gambar 10. Pesan rahasia 300 karakter


text400.txt - Notepad

File Edit Format View Help

Apa itu MSE dan PSNR ? Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan decibel (db). PSNR digunakan untuk mengetahui perbandingan kualitas citra cover sebelum dan sesudah disisipkan pesan. Untuk menentukan PSNR, terlebih dahulu harus ditentukan nilai MS

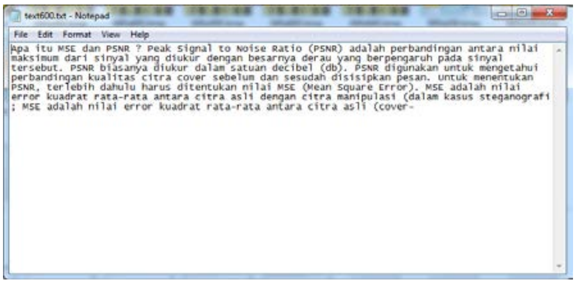
Gambar 11. Pesan rahasia 400 karakter


text500.txt - Notepad

File Edit Format View Help

Apa itu MSE dan PSNR ? Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan decibel (db). PSNR digunakan untuk mengetahui perbandingan kualitas citra cover sebelum dan sesudah disisipkan pesan. Untuk menentukan PSNR, terlebih dahulu harus ditentukan nilai MSE (Mean Square Error). MSE adalah nilai error kuadrat rata-rata antara citra asli dengan citra manip

Gambar 12. Pesan rahasia 500 karakter

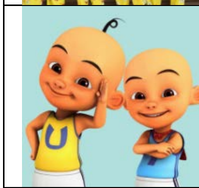
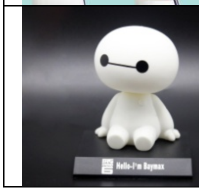

text600.txt - Notepad

File Edit Format View Help

Apa itu MSE dan PSNR ? Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan decibel (db). PSNR digunakan untuk mengetahui perbandingan kualitas citra cover sebelum dan sesudah disisipkan pesan. Untuk menentukan PSNR, terlebih dahulu harus ditentukan nilai MSE (Mean Square Error). MSE adalah nilai error kuadrat rata-rata antara citra asli dengan citra manipulasi (dalam kasus steganografi ; MSE adalah nilai error kuadrat rata-rata antara citra asli (cover-

Gambar 13. Pesan rahasia 600 karakter

Tabel 1. percobaan steganografi

Gambar	**Teks**	**Decode Encode**	**Encode**
	100	Berhasil	Berhasil
	200	Berhasil	Berhasil
	300	Berhasil	Berhasil
	400	Berhasil	Berhasil
	500	Berhasil	Berhasil
	600	Berhasil	Berhasil
	100	Berhasil	Berhasil
	200	Berhasil	Berhasil
	300	Berhasil	Berhasil
	400	Berhasil	Berhasil
	500	Berhasil	Berhasil
	600	Berhasil	Berhasil
	100	Berhasil	Berhasil
	200	Berhasil	Berhasil
	300	Berhasil	Berhasil
	400	Berhasil	Berhasil
	500	Berhasil	Berhasil
	600	Berhasil	Berhasil
	100	Berhasil	Berhasil
	200	Berhasil	Berhasil
	300	Berhasil	Berhasil
	400	Berhasil	Berhasil
	500	Berhasil	Berhasil
	600	Berhasil	Berhasil
	100	Berhasil	Berhasil
	200	Berhasil	Berhasil
	300	Berhasil	Berhasil
	400	Berhasil	Berhasil
	500	Berhasil	Berhasil
	600	Berhasil	Berhasil

Dari percobaan steganografi yang kami lakukan dengan 5 buah gambar dan 6 file teks yang berbeda secara keseluruhan dapat berjalan dengan baik. Seluruh isi file teks pada gambar stegano dapat diterima dengan kondisi yang sesuai dengan teks aslinya.

B. Pengukuran perubahan Citra menggunakan PSNR

Setelah melakukan percobaan penyisipan gambar steganografi terdapat 5 gambar utama atau gambar asli dan 30 gambar stegano. Jadi sebuah gambar akan memiliki 6 hasil steganografi. Kemudian setelah disisipkan pesan seluruh gambar akan kami uji perubahannya dengan menggunakan metode PSNR (Peak signal to Noise Ratio).

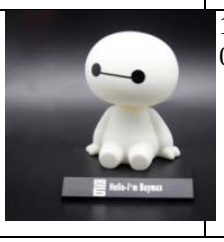

400x400 500x500 640x640 800x800 1000x1000

Gambar 14. Gambar pembanding PSNR

Gambar diatas adalah gambar pembanding dimana berfungsi sebagai acuan perubahan yang terjadi pada gambar hasil penyisipan steganografi.Jadi cara kerjanya adalah kami

membandingkan gambar diatas dengan gambar hasil steganografi.

Tabel 2. Hasil perhitungan PSNR

Gambar	Ukuran	Type	Teks	PSNR (db)
	400x400	Gambar .bmp	100	72.24256
			200	69.19425
			300	67.00384
			400	65.66161
			500	64.81565
			600	63.95630
	500x500	Gambar .bmp	100	74.96472
			200	71.56500
			300	69.63349
			400	68.38817
			500	67.41419
			600	66.61589
	640x640	Gambar .bmp	100	75.61176
			200	70.67615
			300	68.93208
			400	67.91283
			500	66.67062
			600	66.04301
	800x800	Gambar .bmp	100	78.58234
			200	75.15789
			300	73.24680
			400	71.99086
			500	71.02963
			600	70.23905
	1000x1000	Gambar .bmp	100	81.41618
			200	77.92541
			300	76.05893
			400	74.74386
			500	73.81644
			600	72.97965

Dari percobaan diatas kita mendapatkan hasil yang beragam dimana semakin banyak jumlah karakter yang disisipkan maka akan lebih kecil nilai PSNR-nya. Semakin besar resolusi gambar juga mempengaruhi nilai PSNR.

Setelah percobaan diatas kita juga akan melakukan percobaan pengukuran terhadap gambar grayscale, sebagai percobaan maka kita gunakan gambar dengan ukuran 400x400. Kita sediakan 2 buah gambar yaitu gambar 24bit dan gambar grayscale.

Gambar 17. Perbandingan Gambar percobaan kedua

Tabel 3. Perbandingan PSNR antara gambar asli dan gambar grayscale

Gambar	Jenis	Type	Teks	PSNR (db)
	24bit	Gambar .bmp	100	72.24256
			200	69.19425
			300	67.00384
			400	65.66161
			500	64.81565
			600	63.95630
	Grayscale	Gambar .bmp	100	67.58861
			200	64.05281
			300	62.26042
			400	60.68098
			500	59.69679
			600	58.54614
	24bit	Gambar .bmp	100	74.96472
			200	71.56500
			300	69.63349
			400	68.38817
			500	67.41419
			600	66.61589
	Grayscale	Gambar .bmp	100	67.58861
			200	64.05281
			300	62.26042
			400	60.68098
			500	59.69679
			600	58.54614

Dari percobaan diatas dapat disimpulkan bahwa semakin banyak teks yang diinputkan atau disembunyikan maka semakin kecil nilai PSNR dan semakin besar resolusi dari gambar makan semakin besar PSNRnya. Dan untuk perbandingan antara gambar 24bit dengan Grayscale maka dapat disimpulkan bahwa nilai PSNR pada gambar Grayscale lebih kecil.

VI. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari penulisan ini maka dapat disimpukan bahwa apilkasi Steganografi yang telah dihasilkan dari implementasi algoritma LSB (Least Significant Bit) dapat digunakan dengan baik untuk menyembunyikan pesan di dalam pesan sebuah image atau file citra digital sedemikian rupa sehingga orang lain tidak

menyadari ada sesuatu di dalam pesan tersebut. Pada proses ekstraksi, pesan atau informasi yang disisispkan pada file citra uji dalam aplikasi Steganografi ini, dapat diperoleh kembali secara utuh atau dengan kata lain pesan yang disisipkan sebelum proses penyisipan dan setelah proses ekstraksi sama tanpa ada perubahan atau gangguan yang menyebabkan isi pesan tidak dapat diperoleh sepenuhnya

Hasil pengujian nilai PSNR terhadap image atau file citra digital yang dihasilkan dari aplikasi Steganografi inipun menunjukkan nilai yang cukup baik bergantung pada besar ukuran file citra yang digunakan dan besarnya jumlah karakter yang disisipkan pada file citra tersebut. Semakin besar ukuran file citra yang digunakan maka semakin baik nilai PSNR dalam decibel (db) yang diperoleh di bandingkan dengan file citra yang berukuran lebih kecil dengan jumlah sisipan karakter yang sama. Hal ini menunjukkan bahwa untuk memperoleh file citra yang baik setelah proses penyisipan, dan tidak mengalami perubahan yang cukup berarti dari file citra sebelumnya maka besar ukuran file citra dalam piksel dan banyaknya karakter yang akan disisipkan perlu diperhatikan untuk memperoleh hasil yang baik. Serta pengujian terhadap citra grayscale memiliki nilai PSNR yang lebih kecil daripada citra berwarna dengan ukuran, resolusi dan pesan masukan yang sama.

B. Saran

File citra yang dihasilkan setelah proses penyisipan mengalami pengurangan kualitas yang cukup banyak bergantung dari jumlah karakter yang disisipkan, dimana semakin banyak karakter yang disisipkan maka semakin besar pula pengurangan kualitas citra yang diperoleh yang ditandai dengan pengurangan nilai PSNR. Oleh karena itu, untuk meningkatkan kualitas citra dihasilkan maka kedepannya diharapkan dapat dikembangkan suatu aplikasi Steganografi dengan metode lain yang lebih baik agar kualitas citra yang dihasikan tidak jauh berbeda dengan kualitas citra sebelumnya.

DAFTAR PUSTAKA

[1] Rahmat, Basuki, Fairuzabadi. Steganografi Menggunakan Metode Least Significant Bit Dengan Kombinasi Algoritma Kriptografi Vigenère Dan Rc4. Jurnal Dinamika Informatika Volume 5, Nomor 2, September 2010.

[2] Armada, Implementasi Steganography Untuk Pesan Multimedia Menggunakan Android, http://Jurnal.Stmikelrahma.Ac.Id/Assets/File/ARMADA_Stmikelrahma.Pdf (Diakses Pada 29 Oktober 2015).

[3] Sugeng Santoso, Padeli, Arisman. Steganografi Audio (Wav) Menggunakan Metode Lsb (Least Significant Bit). NS-CCIT RAHARJA 2015.

[4] Ghazali Moenandar Male, Wirawan, Eko Setijadi. Analisa Kualitas Citra Pada Steganografiuntuk Aplikasi E-Government. Prosiding Seminar Nasional Manajemen Teknologi XV. 2012.

[5] Fani Soniavita Hijjati, Asep Mulyana, Analisis Dan Implementasi Aplikasi Pengolahan Citra Berbasis Android Dengan Metode Cross Process Universitas Telkom.

[6] Prasetyo, Fahri Perdana. Steganografi Menggunakan Metode Lsb Dengan Software Matlab. Universitas Islam Negeri Syarif Hidayatullah. 2010.