

PENYEMBUNYIAN FILE PADA CITRA DIGITAL DENGAN MENGGUNAKAN METODE MODIFIKASI 2 BIT LSB (LEAST SIGNIFICANT BIT).

Dessy Sukmawardhani, Yerry Soepriyanto, Taufik Rachman

Program Studi Teknik Informatika
Sekolah Tinggi Teknologi STIKMA Internasional Malang
Jl.Raden Panji Suroso 91-A Malang
Email : ktopik@gmail.com

Abstrak

Keamanan suatu informasi dan data yang bersifat privacy adalah hal yang mutlak dilakukan, jangan sampai informasi atau data yang dikirim jatuh pada pihak lawan apalagi dimanipulasi selama proses pengiriman oleh pihak ketiga. Data-data penting dan rahasia tersebut biasanya berupa file sehingga dibutuhkan suatu teknik yang dapat mengamankan file-file tersebut.

Pada penelitian ini akan diimplementasikan teknik steganografi ke dalam program aplikasi untuk menyembunyikan data berupa file teks, file gambar, dan file suara ke dalam citra digital dan mengungkapkan file yang disembunyikan tersebut dengan menggunakan metode modifikasi 2 bit LSB (Least Significant Bit) yakni suatu metode yang mengubah bit ke-7 dan ke-8 dari byte-byte citra digital dengan bit-bit file yang disembunyikan sehingga data yang tersembunyi di dalam citra sulit dipersepsi oleh indera penglihatan manusia.

Penelitian ini diharapkan bisa menghasilkan program aplikasi yang dapat menyembunyikan file dan data yang lebih besar dari metode modifikasi 1 bit LSB. Dan dengan adanya penelitian ini akan didapatkan suatu kesimpulan mengenai kualitas citra hasil encoding dengan menggunakan 2 bit LSB dan perbedaanya dengan citra hasil encoding yang menggunakan metode 1 bit LSB.

Kata kunci : steganografi, citra, modifikasi 2 bit LSB

I. PENDAHULUAN

Latar Belakang

Keamanan suatu data-data penting mutlak diperlukan baik itu dalam proses penyimpanan, pengiriman data maupun pertukaran data. Apalagi jika data itu menyangkut *file-file* rahasia penting perusahaan atau instansi. Diaksesnya *file-file* yang bersifat *privacy* tersebut oleh pihak yang tidak berwenang akan menimbulkan banyak resiko dan kerugian bagi pemiliknya. Oleh karena itu, dibutuhkan mekanisme keamanan data berupa *file* sehingga data tersebut aman dari tangan pihak-pihak yang tidak bertanggung jawab.

Salah satu teknik yang digunakan untuk mengamankan data adalah teknik steganografi. Steganografi adalah teknik untuk menyembunyikan informasi ke dalam suatu media digital, sehingga keberadaan informasi tersebut tidak diketahui orang lain. Media digital tersebut dapat berupa teks, citra, audio atau video. Data rahasia yang disembunyikan juga dapat berupa citra, suara, teks, atau video.

Pada penelitian ini menggunakan teknik penyembunyian data dengan modifikasi 2 bit LSB pada citra digital dengan format **.bmp*, **.gif*, **.png*, **.jpg*, dan **.tif*. Sedangkan data yang disembunyikan berupa *file* multimedia yakni *file* teks, *file* gambar, atau *file* suara. Output yang dihasilkan dari proses penyembunyian *file* ini adalah *file* citra dengan format **.bmp*.

Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah membuat suatu program aplikasi penyembunyian *file* ke dalam citra digital dan mengungkapkan kembali *file* yang disembunyikan tersebut menggunakan metode modifikasi 2 bit LSB (*Least Significant Bit Modifications*).

Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut :

- Media penyembunyian data berupa citra dengan format **.bmp*, **.jpg*, **.gif*, **.png*, atau **.tif*.
- Data yang disembunyikan (*embedded messages*) adalah *file* multimedia berupa *file* teks, *file* gambar, atau *file* suara.
- Citra hasil *encoding* adalah citra *bitmap* dengan format **.bmp*.
- Citra penampung hanya dapat menyembunyikan 1 *file* saja.
- File* yang disembunyikan kapasitasnya harus lebih kecil dari kapasitas citra penampungnya.
- Maksimal ukuran gambar yang digunakan pada penelitian ini adalah 256x256 *pixel* dan *ratio* kapasitas *cover image* dengan *file* yang disembunyikan kurang dari 4 : 1.
- Hanya membahas mengenai proses dan desain sistem penyembunyian dan pengungkapan kembali *file* yang disembunyikan pada citra menggunakan teknik modifikasi 2 bit LSB serta implementasinya dalam *software*.

Tujuan

Tujuan dari penelitian ini adalah menyembunyikan *file* multimedia berupa teks, gambar, maupun suara ke dalam citra digital dengan menggunakan metode modifikasi 2 bit LSB, sehingga informasi berupa *file* tersembunyi dalam citra dan tidak diketahui keberadaanya oleh pihak lain kecuali pihak yang mempunyai kunci untuk membukanya.

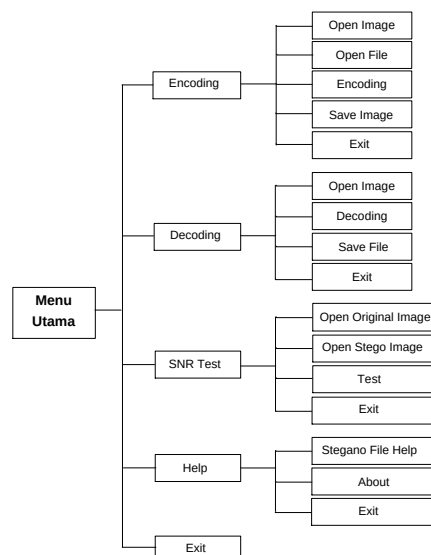
II. RANCANGAN DAN DESAIN SISTEM

Bagian ini berisi tentang uraian rancangan dan desain sistem dalam bentuk diagram yang nantinya digunakan sebagai acuan dalam pembuatan software.

Desain Struktur Menu

Struktur menu pada program ini terbagi menjadi beberapa menu utama dan sub menu yaitu :

- a. **Encoding** : berfungsi untuk masuk ke *form* proses penyembunyian *file* pada gambar. Pada *form* ini terdapat beberapa sub menu antara lain :
 1. *Open Image* : berfungsi untuk membuka citra atau gambar yang akan dijadikan media penyembunyian *file*. Setelah *file* gambar dipilih kemudian ditampilkan pada *form* untuk selanjutnya bisa dilakukan proses penyembunyian *file*.
 2. *Open File* : berfungsi untuk membuka *file* yang akan disembunyikan pada gambar.
 3. *Encoding* : berfungsi untuk melakukan proses penyembunyian *file* pada gambar.
 4. *Save Image* : berfungsi untuk menyimpan gambar yang telah disisipi *file*.
 5. *Exit* : berfungsi untuk keluar dari *form*.
- b. **Decoding** : berfungsi untuk masuk ke *form* proses pengungkapan *file* yang tersembunyi di dalam gambar sehingga *file* tersebut dapat di-load nantinya oleh *user*. Pada *form* ini terdapat beberapa sub menu antara lain :
 1. *Open Image* : berfungsi untuk membuka gambar yang telah disisipi *file* (*stego image*) untuk kemudian dilakukan proses pengungkapan *file*.
 2. *Decoding* : berfungsi untuk melakukan proses pengungkapan *file* yang tersembunyi pada gambar.
 3. *Save File* : berfungsi untuk menyimpan *file* yang telah berhasil diekstraksi dari gambar.
 4. *Exit* : berfungsi keluar dari *form*.
- c. **SNR Test** : berfungsi untuk melakukan pengujian pada gambar sehingga diketahui besar nilai SNR. Pada *form* ini terdapat beberapa sub menu sbb:
 1. *Open Original Image* : membuka gambar yang belum disisipi oleh *file*.
 2. *Open Stego Image* : berfungsi membuka gambar yg telah disisipi oleh *file*.
 3. *Test* : berfungsi untuk melakukan proses perhitungan SNR.
 4. *Exit* : berfungsi untuk keluar dari *form*.



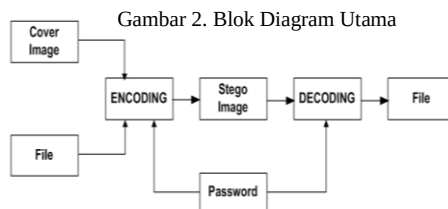
Gambar. 1 Struktur menu utama

Desain Sistem

Desain sistem merupakan proses merancang suatu sistem informasi dimana diharapkan hasil rancangan sistem informasi tersebut nantinya sesuai dengan kebutuhan pemakai. Adapun desain sistem dalam proses penyembunyian *file* ke dalam gambar dan proses pengungkapannya dapat digambarkan sebagai berikut :

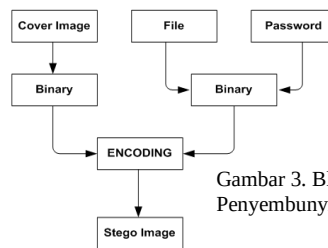
a. Blok Diagram Utama

Gambar blok diagram utama di samping merupakan gambaran secara garis besar teknik penyembunyian dan pengungkapan *file* dengan menggunakan teknik steganografi. Metode steganografi yang dipakai adalah metode modifikasi 2 bit LSB, seperti yang telah dijelaskan pada bagian sebelumnya. Pada program aplikasi yang dibuat dalam penelitian ini terdapat 2 proses. Proses pertama adalah proses *Encoding*, yaitu proses untuk menyembunyikan *file* ke gambar. Proses *Encoding* akan menghasilkan output sebuah gambar baru yang didalamnya terdapat suatu *file*. Gambar ini kita sebut sebagai *stego image*. Sedangkan proses kedua adalah proses *Decoding*, yaitu proses untuk mengungkapkan *file* yang telah disembunyikan pada *stego image*. Pada proses *Decoding* ini akan dihasilkan output berupa *file* yang tersembunyi di dalam *stego image* tersebut. *Password* yang digunakan pada proses *Encoding* harus sama dengan yang digunakan pada proses *Decoding*. Diagram yang lebih detail dari setiap proses diatas digambarkan pada penjelasan di bawah ini.



b. Blok Diagram Penyembunyian File (Encoding)

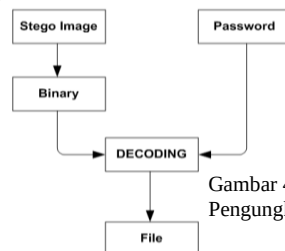
Diagram diatas merupakan alur proses penyembunyian data. Terdiri dari 3 input yakni *cover image* (citra asli atau gambar asli), *file* dan *password*. Masing-masing input tersebut dibinerkan datanya kemudian dilakukan proses *encoding* yakni dengan cara memodifikasi 2 bit LSB dari setiap byte-byte *cover image*-nya. 2 bit tersebut diganti dengan bit-bit *file* dan *password*. Sehingga akan didapatkan output berupa gambar baru yang disini disebut *stego image*. Secara kasat mata *stego image* hampir sama dengan *cover image* walaupun di dalam *stego image* terdapat *file* yang disembunyikan tetapi karena kelemahan mata manusia sehingga perbedaan warna yang kecil tidak akan terlihat secara nyata.



Gambar 3. Blok Diagram Penyembunyian File

c. Blok Diagram Pengungkapan File (Decoding)

Diagram diatas menggambarkan proses pengungkapan *file* yang disembunyikan pada gambar. Input berupa *stego image* dan *password*. Apabila *password*-nya sesuai (sama) dengan *password* yang dimasukkan pada waktu proses *Encoding* maka proses *Decoding* atau pengungkapan *file* dapat dilakukan. Serangkaian bit-bit yang telah disembunyikan ke dalam gambar dapat diungkapkan dengan cara mengumpulkannya kembali dengan menggunakan kebalikan metode penyembunyian yaitu dengan cara mencari bit-bit pada LSB pertama dan kedua (yaitu bit ke-7 dan ke-8) pada *stego image*, lalu bit-bit tersebut dikumpulkan menjadi serangkaian data bit untuk dijadikan per satu *byte*. Kemudian *byte-byte* data tersebut direpresentasikan menjadi sebuah *file* kembali. Sehingga akan didapatkan output berupa *file* yang sama dengan *file* yang diinputkan pada proses *encoding*.



Gambar 4. Blok Diagram Pengungkapan File

III. IMPLEMENTASI DAN PEMBAHASAN

Proses Penyembunyian File (*Encoding*)

Proses penyembunyian *file* ke dalam gambar ini dimulai dengan mengidentifikasi data citra dan mengubahnya menjadi data biner kemudian diteruskan dengan mengidentifikasi *stream* data dari suatu *file* yang akan disembunyikan kemudian *data stream* tersebut dirubah menjadi data biner. Setelah itu, pada bit ke-7 dan ke-8 (LSB pertama dan LSB kedua) dari setiap *byte* data citra, dilakukan modifikasi, yaitu dengan mengganti bit-bit tersebut dengan bit-bit *stream file* yang disembunyikan.

Sebelum bit-bit *stream* diletakkan atau dilekatkan pada data citra, dilakukan pemberian tanda atau penyembunyian karakter tertentu pada citra tersebut. Hal ini untuk menunjukkan bahwa gambar tersebut terdapat *file* rahasianya. Sehingga pada waktu proses *decoding file*, gambar tersebut dapat langsung diidentifikasi apakah dlm gambar tsb terdapat *file* rahasianya atau tidak.

Selain bit-bit *stream* yang disembunyikan juga dilakukan penyembunyian *password* dan panjang *password*, nama *file* yang disembunyikan berikut ekstensinya serta banyaknya karakter dari nama *file* tersebut. *User* juga dapat menambahkan pesan singkat berupa teks untuk disembunyikan pada gambar. Format file citra hasil *encoding* ini adalah format bitmap dgn ekstensi **.bmp*.

Proses Pengungkapan (*Decoding*)

Proses pengungkapan file yang tersembunyi pada suatu citra dilakukan dengan mengumpulkan nilai bit ke-7 dan ke-8 (LSB kedua dan LSB pertama) pada setiap *byte* data citra. Deretan bit-bit tersebut dipisah-pisah menjadi 1 *byte* kemudian dikonversi menjadi desimal. Nilai-nilai desimal tersebut kemudian direpresentasikan menjadi *file* kembali. Sehingga akan didapatkan data *file* yang sesuai dengan data aslinya.

Proses pengungkapan *file* dari gambar hanya dapat dilakukan apabila *password*-nya cocok, dalam hal ini *password* untuk *decoding* harus sama dengan *password* yang dimasukkan pada waktu proses *encoding*. Apabila *password*-nya tidak sama maka ekstraksi *file* dari gambar tidak dapat dilakukan dan *file* tetap tidak bisa dibuka.

Pengujian dan Analisa

a. Pengujian Recovery

Pengujian ini juga dimaksudkan untuk menguji apakah program dapat mengungkapkan kembali file yang telah disembunyikan pada citra. Di bawah ini disajikan beberapa contoh hasil penyembunyian *file* dengan menggunakan program *stegano file* dan hasil pengungkapannya.

1.	Cover image/citra asli : fruit-orig(256x256).bmp (256x256 pixel)		Data yang disembunyikan : a. <i>File</i> : gaji.doc (25,5 KB) b. Text : ini pesan rahasia tlg dijaga/dirahasiakan
	Citra hasil : Fruit.bmp (256x256 pixel)		Hasil Pengungkapan : a. <i>File</i> : Gaji.doc (25,5 KB) b. Teks : ini pesan rahasia tlg dijaga/dirahasiakan. <i>File</i> terekstraksi dengan baik

2.	Cover <i>image</i> /citra asli : <i>Sunflowers.gif</i> (230x256 <i>pixel</i>)		Data yang disembunyikan : a. <i>File</i> : <i>vbregserver.zip</i> (5 KB) b. Teks : ini refrensi untuk Anda
	Citra hasil : <i>Sunflowers.bmp</i> (230x256 <i>pixel</i>)		Hasil Pengungkapan : a. <i>File</i> : <i>vbregserver.zip</i> (5 KB) b. Teks : ini refrensi untuk Anda. (<i>File</i> terekstraksi dengan baik.)
3.	Cover <i>image</i> /citra asli : <i>tulips.jpg</i> (256x229 <i>pixel</i>)		Data yang disembunyikan : a. <i>File</i> : <i>woody woodpeker.wav</i> (13,4 KB)
	Citra hasil : <i>tulips.bmp</i> (256x229 <i>pixel</i>)		Hasil Pengungkapan : a. <i>File</i> : <i>woody woodpeker.wav</i> (13,4 KB) (<i>File</i> terekstraksi dengan baik.)
4.	Cover <i>image</i> /citra asli : 300px.png (101 KB)		Data yang disembunyikan b. <i>File</i> : <i>vbregserver.zip</i> (5 KB)
	Citra hasil : 300px.bmp (176 KB)		Hasil Pengungkapan : a. <i>File</i> : <i>vbregserver.zip</i> (5 KB) (<i>File</i> terekstraksi dengan baik)
5.	Cover <i>image</i> /citra asli : Camera.tif (79 KB)		Data yang disembunyikan c. <i>File</i> : 5.jpg (2,06 KB)
	Citra hasil : Camera.bmp (193 KB)		Hasil Pengungkapan : d. <i>File</i> : 5.jpg (2,06 KB) (<i>File</i> terekstraksi dengan baik)

Tabel.1 Percobaan penyembunyian data dengan software *stegano file*

Dari tabel di atas dapat dilihat bahwa *file* yang disembunyikan dapat terekstraksi dengan baik. Ini berarti kriteria *recovery* (data yang disembunyikan harus dapat diungkap kembali) dalam teknik steganografi telah terpenuhi.

b. Pengujian *Robustness* (ketahanan)

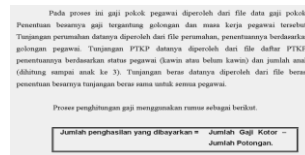
Setelah pengujian *recovery* berhasil dilakukan, pengujian kemudian dilanjutkan untuk melihat apakah pesan yang disisipkan masih dapat diungkap kembali meskipun citra hasil *encoding* mengalami beberapa perubahan. Beberapa contoh pengujian yang dilakukan adalah memanipulasi citra hasil *encoding* dengan beberapa operasi, antara lain:

1. *Horizontal Flip*. *Horizontal Flip* adalah teknik untuk membalik sebuah citra secara horizontal.
2. *Rotation*. Rotasi adalah teknik pembalikan citra dengan derajat tertentu. Rotasi yang dilakukan disini yakni memutar citra sebesar 90 derajat.
3. Penambahan efek *Blur*. *Blur* adalah perubahan warna tiap-tiap pixel dalam citra agar citra terlihat kabur.
4. Penambahan efek *Sharpen*. *Sharpen* adalah sebuah efek yang menonjolkan elemen *pixel* sehingga gambar terlihat lebih tajam.

Di bawah ini adalah gambar citra asli sebelum dilakukan manipulasi dan hasil pengungkapannya.



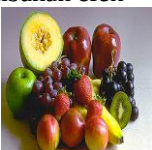
(a)



(b)

Gambar 5. (a) Citra hasil *encoding* : *stego image.bmp* (256x256 pixel)

(b) File hasil *encoding* : *gaji.doc* (26 KB)

No	Citra yang telah dimanipulasi	Hasil Pengungkapan
1.	<i>Horizontal Flip</i> 	File yang disembunyikan tidak dapat diungkap. Muncul pesan “Tidak ada file yang disembunyikan dalam gambar ini”
2.	<i>Rotation</i> 	File yang disembunyikan tidak dapat diungkap. Muncul pesan “Tidak ada file yang disembunyikan dalam gambar ini”
3.	Penambahan efek <i>blur</i> 	File yang disembunyikan tidak dapat diungkap. Muncul pesan “Tidak ada file yang disembunyikan dalam gambar ini”
4.	Penambahan efek <i>sharpen</i> 	File yang disembunyikan tidak dapat diungkap. Muncul pesan “Tidak ada file yang disembunyikan dalam gambar ini”

Dari beberapa percobaan diatas diperoleh pembuktian bahwa ternyata *file* yang disembunyikan pada gambar gagal untuk diekstraksi atau diungkap apabila media penampung (citra *stego* atau citra hasil *encoding*) yang digunakan dikenai perubahan. Hal ini terjadi karena nilai *pixel* dari citra tersebut berubah. Kesimpulan yang dapat diperoleh dari pengujian di atas bahwa metode modifikasi 2 bit LSB ini tidak memenuhi kriteria *robustness* (tidak tahan terhadap manipulasi gambar).

c. Pengujian Kualitas Citra Hasil *Encoding*

- Berdasarkan kapasitas file dengan membandingkan antara metode LSB 1 bit dengan metode LSB 2 bit

Pengujian ini dilakukan dengan cara menyembunyikan file pada citra dengan menggunakan 2 macam metode yakni 1 bit LSB dan 2 bit LSB. Kemudian dilakukan

pengujian kualitas citra hasil encoding antara kedua metode tersebut. File yang disembunyikan mempunyai kapasitas yang berbeda-beda. Pengujian ini dilakukan untuk mengetahui sejauh mana pengaruh kapasitas file terhadap citra hasil *encoding* dan untuk mengetahui perbedaan antara metode modifikasi 1 bit LSB dengan metode modifikasi 2 bit LSB.

Citra yang diuji adalah citra warna 24 bit fruit.bmp ukuran 256x256 pixel. Dan file yang disembunyikan berupa *file* teks dengan ekstensi *.txt

Dari hasil pengujian didapatkan data sebagai berikut :

No	Nama file yang disembunyikan	Kapasitas file yang disembunyikan	SNR (db)	
			Metode 1 bit LSB	Metode 2 bit LSB
1.	<i>Teks1.txt</i>	1 KB	59,96	55,93
2.	<i>Teks2.txt</i>	2 KB	56,86	53,08
3.	<i>Teks3.txt</i>	3 KB	55,1	51,25
4.	<i>Teks4.txt</i>	4 KB	53,83	49,88
5.	<i>Teks5.txt</i>	5 KB	53,24	46,73

Tabel 2. Hasil pengujian berdasarkan kapasitas file, membandingkan antara metode LSB 1 bit dgn metode LSB 2 bit

Dari kelima percobaan di atas dapat diambil beberapa kesimpulan :

1. Semakin besar kapasitas *file* yang disembunyikan pada citra maka kualitas citra hasil *encoding* akan semakin berkurang hal ini ditandai dengan semakin turunnya nilai SNR. Semakin kecil nilai SNR semakin jelek kualitas citra hasil *encoding*-nya.
2. Citra hasil *encoding* dengan menggunakan teknik modifikasi 1 bit LSB kualitasnya lebih bagus daripada menggunakan metode modifikasi 2 bit LSB.

- Berdasarkan jenis data yang disembunyikan pada citra

Pengujian ini dilakukan dengan cara menyembunyikan file pada citra dengan menggunakan metode 2 bit LSB. File yang disembunyikan pada tiap percobaan berbeda-beda jenis datanya, tetapi kapasitasnya sama, dengan maksud mengetahui apakah jenis data yang disembunyikan dapat mempengaruhi kualitas citra hasil encoding.

Citra yang diuji adalah citra warna 24 bit fruit.bmp ukuran 256x256 pixel. File yang disembunyikan berkapasitas 10 KB.

Dari hasil pengujian, didapatkan data sebagai berikut :

No	Jenis data	Nama file yang disembunyikan	Kapasitas file yang disembunyikan	SNR (db)
1.	File teks (*.txt)	<i>Teks 10K.txt</i>	10 KB	45, 49
2.	File doc	<i>Document.rtf</i>	10 KB	42,75
3.	File Gambar	<i>Walpaper4.jpg</i>	10 KB	43,38
4.	File Suara	<i>WinXPStart.mp3</i>	10 KB	45,59

Tabel 3. Hasil pengujian SNR berdasarkan jenis data yang disembunyikan

Dari percobaan di atas dapat diambil kesimpulan bahwa jenis data tidak mempengaruhi kualitas dari citra hasil encoding karena perubahan nilai-nilai SNR dari percobaan-percobaan di atas tidak terlalu significant.

- Berdasarkan format file citra penampung yang digunakan

Pengujian ini dilakukan dengan cara menyembunyikan file pada citra dengan menggunakan metode 2 bit LSB. File yang disembunyikan pada tiap percobaan sama baik jenis datanya maupun kapasitasnya. Pengujian ini dilakukan untuk mengetahui format citra yang kurang bagus untuk digunakan sebagai media penyembunyian dilihat dari segi kualitas citra encoding yang dihasilkan. Sehingga dengan pengujian ini kita bisa tahu format citra yang sebaiknya tidak digunakan pada *software* ini sebagai media penampung.

Citra yang diuji adalah citra warna 24 bit ukuran 256x256 *pixel*. Dan *file* yang disembunyikan adalah *file* “*teks 10K.txt*” dengan kapasitas file sebesar 10 KB.

Dari hasil pengujian, didapatkan data sebagai berikut :

No	Citra asli	Citra hasil encoding	SNR (db)
1.	 <i>fruit_ori.bmp</i>	 <i>dari_bmp.bmp</i>	45, 49
2.	 <i>fruit_ori.jpg</i>	 <i>dari_jpg.bmp</i>	45,42
3.	 <i>fruit_ori.gif</i>	 <i>dari_gif.bmp</i>	39,12
4.	 <i>fruit_ori.png</i>	 <i>dari_png.bmp</i>	45,49
5.	 <i>fruit_ori.tif</i>	 <i>dari_ori.bmp</i>	45,49

Dari hasil SNR dan gambar yang dihasilkan terlihat bahwa citra GIF yang digunakan sebagai media penampung memiliki SNR yang paling rendah diantara keempat format lain yang digunakan dalam pengujian ini. Hasil citra *encoding*-nya pun terlihat buruk ini berarti kualitasnya sangat jelek karena terlihat nyata sekali kerusakan citra hasil *encoding*-nya.

Tabel 4. Hasil pengujian SNR berdasarkan format *file* yang dijadikan media penampung

IV. KESIMPULAN DAN SARAN

Kesimpulan

1. Penyembunyian data pada citra dengan menggunakan metode modifikasi 2 bit LSB tidak tahan terhadap operasi manipulasi yang dilakukan pada citra penampung seperti rotasi, *flip*, *blur*, *sharpen* atau penajaman dan operasi-operasi manipulasi lainnya yang biasa dilakukan pada citra.
2. Semakin besar ukuran file yang disembunyikan maka kualitas citra hasil *encoding* semakin buruk.
3. File yang disembunyikan ukurannya harus lebih kecil dari media atau citra penampungnya.
4. Penyembunyian file dengan menggunakan metode modifikasi LSB 2 bit tidak bagus untuk citra penampung dengan format GIF karena kualitas citra hasil *encodingnya* mengalami penurunan.
5. Metode modifikasi 2 bit LSB dapat menyembunyikan file atau data dengan ukuran yang lebih besar apabila dibandingkan dengan menggunakan metode 1 bit LSB tetapi kualitas citra hasil *encoding-nya* masih lebih baik jika menggunakan metode 1 bit LSB.
6. Jenis data yang disembunyikan tidak berpengaruh pada kualitas citra hasil *encoding-nya*.

Saran

- a. Masih perlu adanya pengembangan *software* untuk diimplementasikan pada citra dengan format lain dan output yang dihasilkan sesuai dengan format asli dari citra penampung dan file yang disembunyikan kapasitasnya bisa lebih besar lagi.
- b. Untuk pengembangan selanjutnya bisa dilakukan metode modifikasi 2 bit LSB untuk menyembunyikan lebih dari satu file pada citra.
- c. Perlu adanya studi tentang metode steganografi yang cocok untuk diterapkan pada citra GIF.
- d. Pada penelitian selanjutnya diharapkan teknik steganografi dapat diimplementasikan pada media audio dan video.

DAFTAR PUSTAKA

- a. Jain, Anil K, "*Fundamentals of Digital Image Processing*", Prentice Hall International, 1989.
 - b. Khodijah, "*Penyembunyian Informasi ke dalam Gambar Bitmap dengan Metode Pelekatan pada LSB*", Tugas Akhir Jurusan Informatika, 2005.
 - c. Masaleno, Andino, "*Pengantar Steganografi*", 2003-2006.
 - d. Modul Mata Kuliah Multimedia, "*Kompresi Citra*", Fakultas Teknik Informatika Universitas Kristen Duta Kencana, 2005-2006.
 - e. Munir, Rinaldi, "*Pengolahan Citra Digital dengan Pendekatan Algoritmik*", Informatika, Bandung, 2004
 - f. Munir, Rinaldi, "*Kriptografi*", Informatika, Bandung, 2006.
 - g. Pamungkas, "*Tip & Trik Microsoft Visual Basic 6.0*", Elex Media Komputindo, Jakarta, 2000.
 - h. Raharjo, Budi, "*Keamanan Sistem Informasi Berbasis Internet*", 1998-2005.
 - i. <http://budi.insan.co.id/courses/el7010/2003/msw-report.pdf>, diakses tanggal 20 November 2006
- Sutedjo, Budi dan An, Michael, "*Algoritma dan Teknik Pemrograman*", Andi, Yogyakarta, 1997.