MIND Journal | ISSN (p): 2528-0015| ISSN (e): 2528-0902 | Vol. 3 | No. 2 | Halaman 36-46
DOI : https://doi.org/10.26760/mindjournal Desember 2018

Perbandingan Metode *Most Significant Bit* dan *Least Significant Bit* pada Steganografi untuk Keamanan Data Media Digital

DEWI ROSMALA, ANGGA KUSUMA K

Institut Teknologi Nasional

Email: rosmala@yahoo.com

ABSTRAK

Steganografi merupakan ilmu atau teknik menyembunyikan data ke dalam suatu media dengan tujuna orang lain tidak mengetahui keberadaan pesan tersebut. Metode Most Significant Bit (MSB) dan Least Significant Bit (LSB) adalah metode yang digunakan dalam teknik steganografi. Biner dari secret file disisipkan kedalam bit paling berpengaruh untuk MSBatau bit paling tidak berpengaruh untuk LSB ke dalam cover file. Dari proses penyisipan dapat dilihat perbandingan ukuran file image sebelum dan sesudah disisipi pesan rahasia. Untuk menganalisis metode mana yang lebih baik terdapat paramater yang haruis dipenuhi oleh masing-masing metode, suatu stego file dapat dikatakan bagus apabila memiliki nilai PSNR diatas 30 dB dan nilai MOS 3 (fair). Dari hasil penelitian dapat diambil kesimpulan, yaitu metode LSB lebih baik dari pada metode MSB, hal itu dibuktikan dengan nilai PSNR rata-rata metode LSB adalah sebesar 70.31 dB dan dengan rata-rata nilai MOS 4 (good) sedangkan metode MSB hanya memiliki nilai PSNR 28.31 dB dan rata-rata nilai MOS 2 (Poor).

***Kata kunci**: Steganografi, Most Significant Bit (MSB), Least Significant Bit (LSB).*

ABSTRACT

Steganography is the art and science which studies the way of confidential information hiding into a medium so human doesn't realize the message existence. Most Siginificant bit (MSB) and least significant bit (LSB) method are used method in steganography technique. Binary from secretfile inserted into most influenced bit for MSB or most uninfluenced bit for LSB into cover file. From inserted process, it can be seen the file image size comparison before and after inserted by secret message. To analyze which method are better, it has must fulfilled parameters for each methods, a stego file is good if it has above 30 dB of PSNR value and 3 (fair) of MOS value. From the test result it can be concluded, that LSB method is better than MSB method, that can be proved with 70.31 dB of LSB method PSNR average value and 4 (good) of MOS average value, meanwhile MSB method just has 28.31 dB of PSNR average value and 2 (poor) of MOS average.

***Keyword**: Steganography, Most Significant Bit (MSB), Least Significant Bit (LSB).*

1. PENDAHULUAN

Steganografi adalah metode untuk melakukan penyembunyian informasi atau pesan kedalam media lain seperti citra digital, teks, suara atau video. Metoda ini dilakukan untuk menghindari kecurigaan orang lain[1]. Dibutuhkan dua unsur dalam steganografi yaitu data rahasia (*secret file*) dan media penampung (*cover file*). Penerapan teknik steganografi membutuhkan sebuah metode dalam melakukan penyembunyian data. Metode yang bisa digunakan untuk menyembunyikan data adalah metode *Most Significant Bit* (LSB) dan metode *Least Significant Bit* (LSB). Dan dari penyembunyian data ke dalam media digital inilah dapat dilihat perbandingan kualitas media digital sebelum dan sesudah penyisipan dan mencari metode manakah yang lebih baik dalam melakukan pengamanan data.Penelitian yang dibahas adalah mengenai bagaimana data diamankan dengan metode MSB dan LSB, kemudian dilakukan perbandingan dengan parameter aspek *imperceptibility* dengan cara *Mean Opinion Score* (MOS) , aspek *fidelity* dengan parameter pengukurannya adalah *Peak Signal-to-Noise Ratio* (PSNR) dan aspek *recovery*.

Tujuan dari penelitian ini adalah melakukan analisis untuk mengetahui metode yang lebih baik antara MSB dan LSB, dengan parameter yang diuji adalah dari aspek*imperceptibility* dengan cara MOS, aspek*fidelity*dengan pengukuran PSNR dan aspek *recovery*.

Data rahasia yang disembunyikan berupa *message* dan file teks (*.txt,) Data digital yang digunakan sebagai media penyembunyian berupa file *image* (*.png). Perbandingan kualitas citra yang dievaluasi adalah aspek *imperceptibility* dengan cara*Mean Opinion Score* (MOS),aspek *fidelity* dengan pengukuran menggunakan *Peak Signal-to-Noise Ratio* (PSNR) dan aspek *recovery*.

2. METODOLOGI PENELITIAN

1.1 Subjek penelitian

Subjek penelitian pada penelitian ini adalah tiga puluh *original file* dan *stego file* digunakan untuk menganalisis perubahan kualitas citra, lalu tujuh koresponden digunakan menganalisis perubahan kualitas citra dengan *Mean Opinion Score*(MOS).Berikut adalah tabel parameter MOS

Tabel 1. Parameter Mos

Nilai	Kualitas	Keterangan
5	*Excellent*	Tidak terdapat *noise* sama sekali (*imperceptible*)
4	*Good*	*Noise* terlihat/terdengar namun tidak mengganggu (*Perceptible but not annoying*)
3	*Fair*	*Noise* sedikit mengganggu (*Slightly annoying*)
2	*Poor*	*Noise* mengganggu (*Annoying*)
1	*Good*	*Noise* sangat mengganggu (*Very annoying)*

1.2 Teknik pengumpulan data

Pengujian *Mean Opinion Score* menggunakan teknik Simple Random Sampling (SRS). SRS digunakan apabila penelitian bersifat umum, deskriptif dan dipilih secara acak. Setiap unit yang ada didalam populasi memiliki kesempatan yang sama untuk dapat terpilih sebagai sampel **(Nurfadli,2009).**

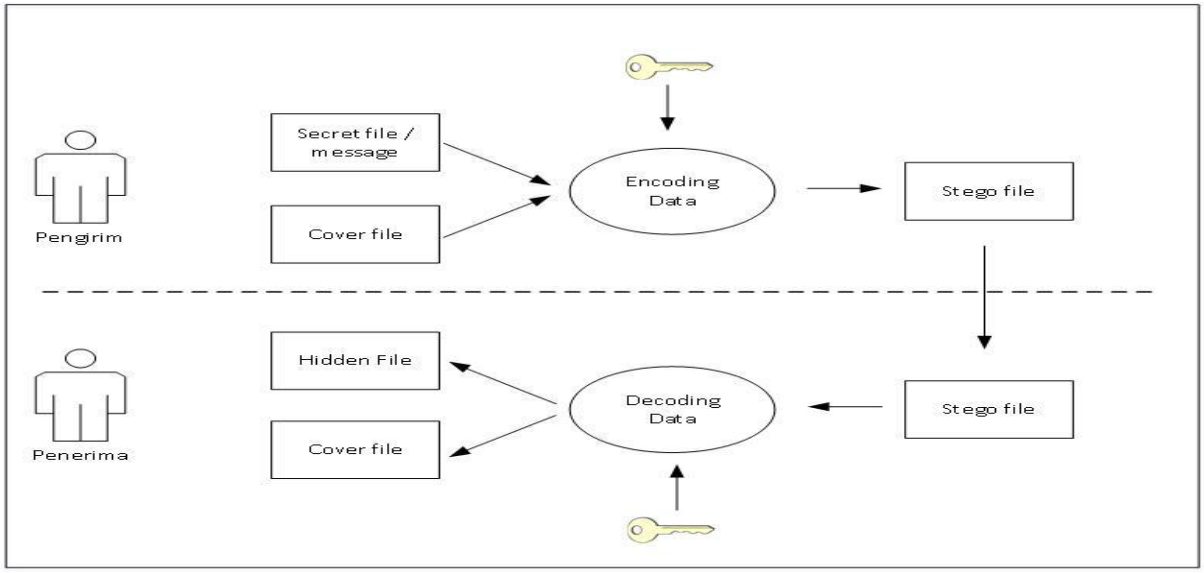
1.3 Studi Literatur

Studi ini dilakukan dengan cara mencari sekaligus mempelajari beberapa literatur dan artikelmengenaisteganografi sebagai acuan dalam perencanaan dan pembuatan sistem atau aplikasi.

3. ANALISIS DAN PEMBAHASAN

3.1. Proses Kerja Aplikasi Steganografi

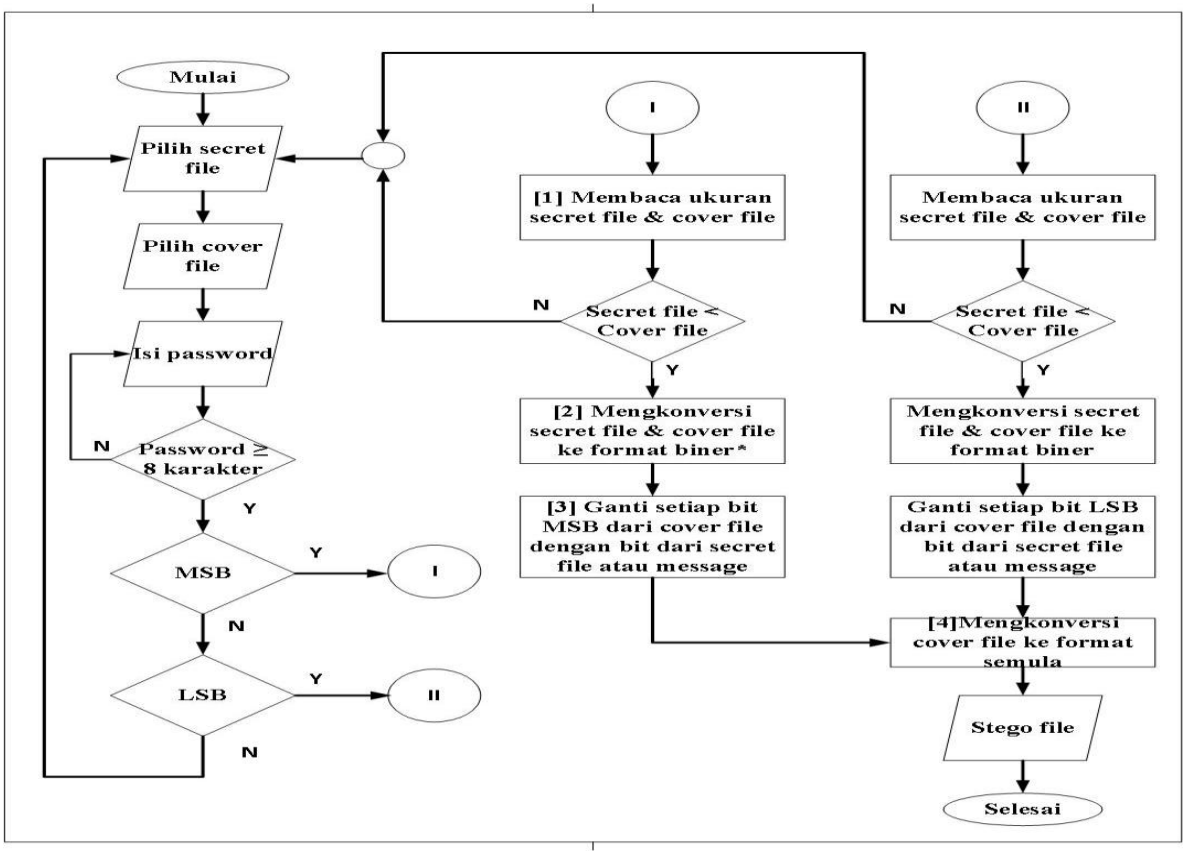
Aplikasi steganografi mempunyai 2 proses yaitu proses penyisipan data (*encode*) dan proses pengambilan data (*decode*). Pada Gambar 1dapat dilihat gambaran dari proses steganografi.



Gambar 1. Gambaran Aplikasi

3.2 Proses Penyisipan Data (*encode).*

Pada proses *encode*dibutuhkan 2 media yangdigunakan sebagai media penampung (*cover file)* dan media yang akan disisipkan (*secret file).Cover file* yang digunakan *file* berformat *image* yang memiliki ekstensi *.png sedangkan *secret file* yang digunakan *file* berformat **.txt.* Proses penyisipan data dapat dilihat pada Gambar 2.Contoh*cover file* dan *secret file*dapat dilihat pada Gambar 3. Proses selanjutnya adalah pengecekan *password,* dimana *password* minimal 8 karakter, apabila kurang dari 8 karakter, maka Proses MSB dan LSB tidak bias diproses.



Gambar 2. Proses Encode

Gambar 3. Cover file dan secret file

1. Membaca ukuran *secret file* dan *cover file*

Pada proses membaca ukuran *secret file* dan *cover file* program melakukan pengecekan ukuran *bytes* dari *secret file* maupun *cover file*, apabila jumlah *bytes secret file* lebih besar dari *cover file* maka proses selanjutnya tidak bisa dilakukan, apabila ukuran *secret file* lebih kecil dari 1/8 dari *cover file* maka proses selanjutnya dapat dilanjutkan. Berikut adalah contoh proses membaca ukuran *secret file* dan *cover file* ditunjukan oleh Gambar 4.

3kb	24kb	✓
5kb	24kb	×

Gambar 4. Membaca ukuran *secret file* dan *cover file*

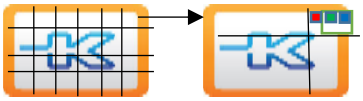
Pada Gambar 4 bagian atas menunjukan bahwa ukuran *secret file* lebih kecil dari1/8 ukuran *cover file* yang memiliki kapasitas maksimun 4kb. Sedangkan bagian bawah menunjukan bahwa ukuran *secret file* lebih besar dari 1/8 ukuran *coverfile.*

2. Konversi Ke Format Biner

Proses konversi ke format biner dibagi menjadi 2 tahapan, yaitu konversi *cover file* dan konversi *secret file*. Berikut adalah penjelasan dari masing-masing tahapan.

2.1 Cover file

File citra memiliki piksel (titik) pada gambar mempunyai tiga susunan warna yaitu merah, hijau dan biru (RGB) dimana disusun oleh 8 bit (byte) desimal dari 0 sampai dengan 255, untuk mendapat nilai desimal RGB maka harus menggunakan algoritma *color picker*.Misalkan suatu citra 3x3 memiliki nilai piksel RGB sebagai berikut .Nilai paling kiri adalah nilai dari Red, di tengah adalah nilai dari Green, dan nilai paling kanan adalah nilai dari Blue.



RGB

182, 211, 229	182, 211, 229	182, 211, 229
182, 211, 229	183, 212, 230	180, 211, 229
182, 211, 229	182, 211, 225	180, 211, 229

Gambar 5. Proses Cover File

cara konversi desimal ke biner adalah dengan membagi nilai desimal dengan 2 kemudian apabila hasil bagi memiliki sisa, maka masukan angka 1, apabila tidak ada sisa maka masukan angka 0, ulangi proses ini dengan tidak memasukan angka sisa (belakang koma) sampai tidak bisa dibagi dengan 2 lagi ,misalnya konversi desimal 182 ke dalam format biner:

182 ÷ 2 = 91tidak ada sisa 0
91 ÷ 2 = 45ada sisa 1
45 ÷ 2 = 22ada sisa 1
22 ÷ 2 = 11tidak ada sisa 0
11 ÷ 2 = 5 ada sisa 1
5 ÷ 2 = 2 ada sisa 1
2 ÷ 2 = 1 tidak ada sisa 0
1 ÷ 2 = 0 ada sisa 1

Maka nilai biner dari 182 adalah 10011101, pembacaan biner tersebut dimulai dari hasil pembagian angka yang terkecil terlebih dahulu. Dikonversikan ke dalam bentuk biner menjadi sebagai berikut:

10110110, 11010011, 11100101	10110110, 11010011, 11100101	10110110, 11010011, 11100101
10110110, 11010011, 11100101	10110111, 11010100, 11100110	10110100, 11010011, 11100101
10110110, 11010011, 11100101	10110110, 11010011, 11100001	10110100, 11010011, 11100101

2.2 Secret file

Setelah mengubah *cover file,* langkah selanjutnya adalah mengubah *secret file* kedalam biner. Misalkan *secret file* yang akan disispkan berupa text "aku", apabila direpresentasikan ke dalam biner yang mengacu pada tabel ASCII, kata "aku" ini menjadi sebagai berikut:

Char	Biner
A	01100001
K	01101011
U	01110101

3. Mengganti bit dari *cover file* dengan bit dari *secret file*

Pada proses MSB *bit* dari *cover file* yang diganti adalah *bit* yang paling berpengaruh atau *bit* yang terdepan. Berikut adalah perubahan biner RGB pada poin 2 setelah disisipi oleh *secret file.*

00110110, **1**1010011, **1**1100101	**0**0110110, **0**1010011, **0**1100101	**0**0110110, **1**1010011, **0**1100101
10110110, **1**1010011, **0**1100101	**1**0110111, **0**1010100, **1**1100110	**1**0110100, **0**1010011, **1**1100101
10110110, **1**1010011, **0**1100101	**1**0110110, **0**1010011, **1**1100001	10110100, 11010011, 11100101

Sedangkan proses LSB *bit* dari *cover file* yang diganti adalah *bit* yang paling tidak berpengaruh atau *bit* yang paling belakang. Berikut adalah perubahan biner RGB pada poin 2 setelah disisipi oleh *secret file.*

1011011**0**, 1101001**1**, 1110010**1**	1011011**0**, 1101001**0**, 1110010**0**	1011011**0**, 1101001**1**, 1110010**0**
1011011**1**, 1101001**1**, 1110010**0**	1011011**1**, 1101010**0**, 1110011**1**	1011010**1**, 1101001**0**, 1110010**1**
1011011**1**, 1101001**1**, 1110010**0**	1011011**1**, 1101001**0**, 1110000**1**	10110100, 11010011, 11100101

4. Konversi *cover file* ke format decimal

Format biner dari *cover file* yang telah disisipi *secret file,* formatnya dirubah kembali menjadi bilangan desimal RGB. Untuk metode MSB konversi dilakukan sebagai berikut:

54, 211, 229	54, 83, 101	182, 211, 54
182, 211, 101	183, 84, 230	180, 83, 229
182, 211, 101	182, 83, 225	180, 211, 229

Misalkan nilai 8 bit 10110110 dikonversi menjadi nilai desimal, dengan memakai rumus seperti sebagai berikut:

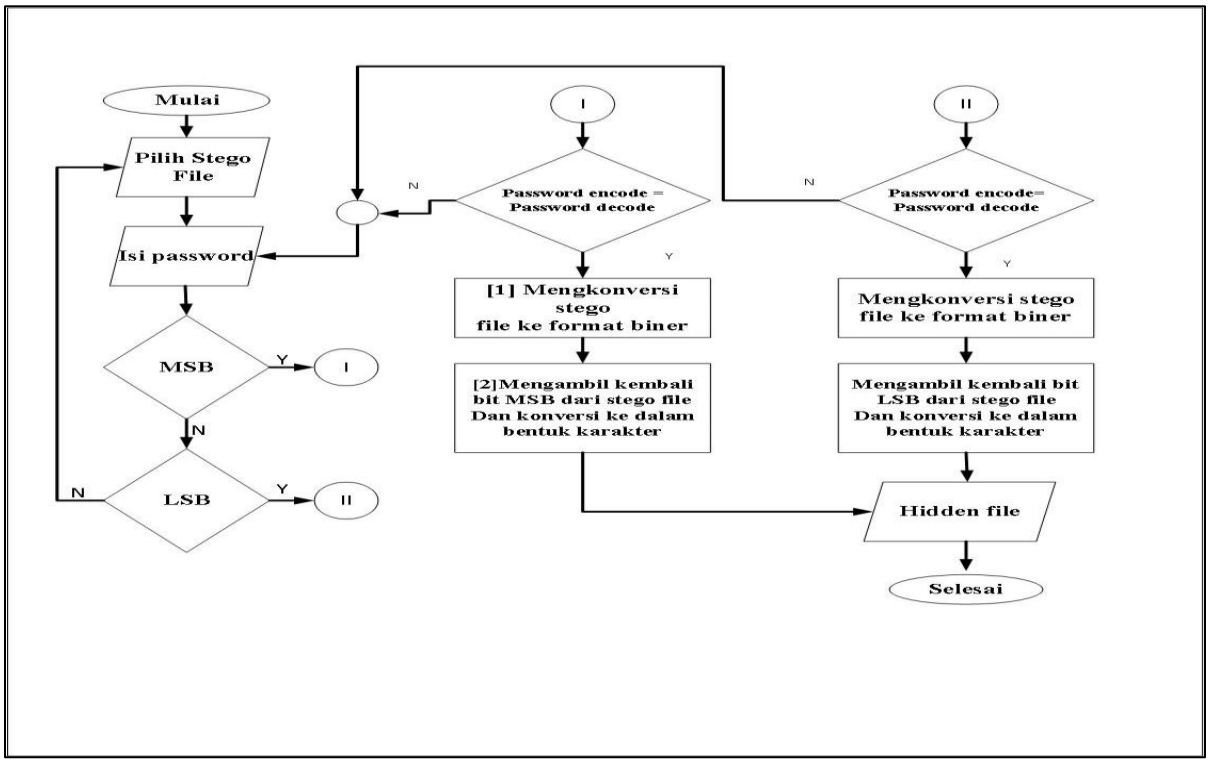
128	**64**	**32**	**16**	**8**	**4**	**2**	**1**
1	**0**	**0**	**1**	**1**	**1**	**0**	**1**

Proses konversi, nilai dari baris atas di mana ada angka 1 di baris bawah tabel, kemudian ditambahkan nilai-nilai tersebut bersamaan. Misalnya, dalam contoh, menjumlahkan angka pada baris atas yang diwakili oleh angka 1 pada baris bawah maka dijumlahkan seperti ini :
128 + 16 + 8 + 4 + 1 = 182. berikut adalah perubahan biner *cover file* pada poin 3 ke dalam format semula dengan menggunakan metode LSB. Berikut adalah perubahan biner dari poin 3, menjadi format semula yaitu desimal RGB dengan metode LSB.

182, 211, 229 183, 211, 228	182, 210, 228 183, 212, 231	182, 211, 228 181, 210, 229
183, 211, 228	183, 210, 225	180, 211, 229

5. Proses Pengambilan Data (Decode)

Pada proses *decode*dibutuhkan 1 media *image* yang sudah disisipi pesan rahasia (*stego file). Stego file* tersebut memiliki *password,* apabila *password* yang dimasukan salah maka proses selanjutnya tidak bisa dilakukan. Proses MSB dan LSB yang dijelaskan sebagai berikut.Proses *decode* digambarkan oleh diagram alir dan ditunjukkan oleh Gambar 6



Gambar 6 Proses Decode

1. Konversi stego file ke format biner

Pada proses konversi *stego file* ke format biner hampir sama dengan konversi *cover file* pada proses *encode.* Yaitu merubah desimal RGB ke nilai biner. Berikut adalah contoh proses mengubah *stego file* ke dalam format biner pada proses MSB.

54, 211, 229	54, 83, 101	182, 211, 54
182, 211, 101	183, 84, 230	180, 83, 229
182, 211, 101	182, 83, 225	180, 211, 229

Dikonversikan ke dalam bentuk biner menjadi sebagai berikut:

00110110, 11010011, 11100101	00110110, 01010011, 01100101	00110110, 11010011, 01100101
10110110, 11010011, 01100101	10110111, 01010100, 11100110	10110100, 01010011, 11100101
10110110, 11010011, 01100101	10110110, 01010011, 11100001	10110100, 11010011, 11100101

Sedangkan untuk metode LSB adalah sebagai berikut

182, 211, 229	182, 210, 228	182, 211, 228
182, 211, 228	183, 212, 231	181, 210, 229
183, 211, 228	183, 210, 225	180, 211, 229

Dikonversikan ke dalam bentuk biner menjadi sebagai berikut:

10110110, 11010011, 11100101	10110110, 11010010, 11100100	10110110, 11010011, 11100100
10110110, 11010011, 11100100	10110111, 11010100, 11100111	10110101, 11010010, 11100101
10110111, 11010011, 11100100	10110111, 11010010, 11100001	10110100, 11010011, 11100101

2. Mengambil kembali Bit ke dalam bentuk karakter

Pada proses ini yaitu menggambil kembali *bit* MSB dalam bentuk karakter, untuk menjadi 1 karakter, dibutuhkan 8 *bit* yang diambil dari setiap *bit* sesuai metode yang ada di setiap piksel.Berikut adalah nilai biner MSB yang diambil untuk di konversi ke dalam bentuk karakter.

00110110, **1**1010011, **1**1100101	**0**0110110, **0**1010011, **0**1100101	**0**0110110, **1**1010011, 01100101

011000010 dikonversi menjadi karakter adalah huruf "a".

00110110, 11010011, 11100101	00110110, 01010011, 01100101	00110110, 11010011, **0**1100101
10110110, **1**1010011, **0**1100101	**1**0110111, **0**1010100, **1**1100110	**1**0110100, 01010011, 11100101

01101011 dikonversi menjadi karakter adalah huruf "k".

00110110, 11010011, 11100101	00110110, 01010011, 01100101	00110110, 11010011, 01100101
10110110, 11010011, 01100101	10110111, 01010100, 11100110	10110100, **0**1010011, **1**1100101
10110110, **1**1010011, **0**1100101	**1**0110110, **0**1010011, **1**1100001	10110100, 11010011, 11100101

01110101 dikonversi menjadi karakter adalah huruf "u". Dan akan menghasilkan *hidden file* yaitu kata "aku". Sedangkan untuk nilai LSB adalah sebagai berikut.

1011011**0**, 1101001**1**, 1110010**1**	1011011**0**, 1101001**0**, 1110010**0**	1011011**0**, 1101001**1**, 11100100

011000010 dikonversi menjadi karakter adalah huruf "a".

10110110, 11010011, 11100101	10110110, 11010010, 11100100	10110110, 11010011, 1110010**0**
1011011**1**, 1101001**1**, 1110010**0**	1011011**1**, 1101010**0**, 1110011**1**	1011010**1**, 11010010, 11100101

01101011 dikonversi menjadi karakter adalah huruf "k".

10110110, 11010011, 11100101	10110110, 11010010, 11100100	10110110, 11010011, 11100100
10110110, 11010011, 11100100	10110111, 11010100, 11100111	10110101, 11010010, 1110010**1**
1011011**1**, 1101001**1**, 1110010**0**	1011011**1**, 1101001**0**, 1110000**1**	10110100, 11010011, 11100101

01110101 dikonversi menjadi karakter adalah huruf "u". Dan akan menghasilkan *hidden file* yaitu kata "aku".

6. Skenario Pengujian

Pengujian yang dilakukan terdiri dari 3 aspek, yaitu aspek *imperceptibility, fidelity,* dan *recovery.* Untuk aspek *imperceptibility* pengujian yang dilakukan Pengujian ini dilakukan dengan cara memperlihatkan citra original dengan citra yang sudah disisipi pesan rahasia baik dengan metode MSB maupun LSB kepada 7 orang koresponden dengan menggunakan parameter*Mean opinion score* (MOS). MOS merupakan sebuah metode pengujian subjektif yang dilakukan untuk mengukur kualitas suatu media berdasarkan deskripsi kualitatif dari apa yang dilihat atau didengar (*imperceptibility*)[2], misalkan "sangat bagus" atau "sangat buruk". Untuk aspek *fidelity* pengujian yang dilakukan adalah dengan cara menghitung nilai PSNR dari masing-masing citra. PSNR adalah*Peak Signal-to-Noise Ratio* (PSNR) merupakan sebuah metode pengujian objektif untuk mengukur kualitas suatu data digital (*fidelity*) dengan menghitung kemiripan dari data digital asli dengan data digital yang telah disisipi data rahasia.PSNR dihitung dari pengukuran distorsi atau *error* dari sebuah digital yang telah disisipi data rahasia. Nilai *Mean Square Error* (MSE) ditentukan terlebih dahulu Untuk menentukan PSNR. MSE adalah nilai *error* kuadrat rata-rata antara citra asli (*cover-image*) dengan citra hasil penyisipan (*stego-image*). PSNR adalah skala logaritmik dalam *decibel* (dB). Apabila Nilai PSNR dibawah 30 dB maka kualitas yang bisa dikatakan rendah,karena distorsi penyisipan terlihat jelas. Kualitas *stego-image* yang tinggi berada pada nilai lebih dari 40dB (Smita, 20017). Pengujian selanjutnya adalah pengujian aspek *recovery* dengan melakukan pengujian berhasil atau tidaknya pengambilan *secret file* dari *stego file*

.

7. Pengujian Aspek *Imperceptibility*

Berikut adalah hasil MOS untuk metode MSB dan LSB dapat dilihat padatabel 2 dan tabel 3Hasil pengujian MOS dengan metode MSB menunjukkan bahwa *stego file* yang dihasilkan30 memiliki kualitas yang buruk, karena nilai MOS rata-rata hanya berada di angka 2, artinya terdapat *noise* dan mengganggu.Sedangkan metode LSB memiliki nilai MOS rata-rata diangka 4, artinya terdapat *noise* namun tidak mengganggu.

Tabel 2 Pengujian MOS MSB

No.	Cover File		Secret file	Stego File	MOS
	Nama File	Ukuran (Bytes)	Ukuran (Bytes)	Ukuran (Bytes)	
1	Mobil.png	420864	8253	723968	2
2	Mobil.png	420864	13516	740352	2
3	Jembatan.png	613376	15360	923648	2
4	Jembatan.png	613376	22528	941056	2
5	Jembatan.png	613376	28672	955392	2
6	Payung.png	657408	30720	946176	2
7	Payung.png	657408	30720	946176	2
8	Payung.png	657408	36864	957440	2
9	bumi.png	681984	39936	896000	1
10	bumi.png	681984	50176	903168	1

Tabel 3 Pengujian MOS LSB

No.	Nama File	Ukuran (Bytes)	Ukuran (Bytes)	Ukuran (Bytes)	MOS
1	Mobil.png	420864	8253	721920	4
2	Mobil.png	420864	13516	737280	4
3	Jembatan.png	613376	15360	903168	4
4	Jembatan.png	613376	22528	914432	4
5	Jembatan.png	613376	28672	919552	4
6	Payung.png	657408	30720	926720	4
7	Payung.png	657408	30720	926720	4
8	Payung.png	657408	36864	933888	4
9	bumi.png	681984	39936	886784	3
10	bumi.png	681984	50176	891904	3

8. Perbandingan Citra

Berikut adalah perbandingan antara citra original dan citra yang sudah disisipi pesan baik menggunakan metode MSB atau metode LSB. Ditunjukkan oleh Gambar 5.

Citra Original | Metode MSB | Metode LSB

Gambar 5. Perbandingan Citra

9. Pengujian Aspek *Fidelity*

Pengujian selanjutnya adalah pengujian aspek *fidelity* dengan mencari nilai MSE dan PSNR dari masing-masing metode setelah *secret fIile* disisipkan ke dalam *cover file*. Tabel 4dan tabel 5menunjukan hasil MSE dan PSNR metode MSB dan LSB.Hasil penelitian metode MSB menunjukkan nilai rata-rata MSE adalah 696.24 dan PSNR 20.01dB menunjukkan bahwa citra yang sudah disisipi *secret file*tingkat kesalahan atau kerusakannya sangat besar dan berbeda jauh dengan citra aslinya. Sedangkan untuk metode LSB menunjukkan nilai rata-rata MSE 0.06177 dan PSNR 61.78dB, artinya citra yang sudah disisipi tingkat kesalahan atau kerusakannya sangat sedikit dan hamper mirip dengan citra aslinya. Untuk presentase perubahan ukuran *file,* metode MSB mengalami perubahan ukuran *file* sebesar 50.5%, sedangkan metode LSB mengalami perubahan ukuran file sebesar 47.78%.

10. Pengujian Aspek *Recovery*

Berikut adalah hasil pengujian aspek *recovery* untuk metode MSB dan LSB, ditunjukkan olehTabel 6dan tabel 7.Hasil penelitian menunjukkan bahwa proses pengambilan data kembali (*decode)* pada aspek *recovery* baik metode MSB atau LSB berhasil dilakukkan.

Tabel 4. Hasil Aspek Recovery MSB

No.	Cover File			Secret file		Proses	
	Nama File	Ukuran (Bytes)	Kapasitas Maksimum (Bytes)	Nama File	Ukuran (Bytes)	Encode	Decode
1	Mobil.png	420864	52608	1.txt	8253	v	v
2	Mobil.png	420864	52608	2.txt	13516	v	v
3	Jembatan.png	613376	76672	3.txt	15360	v	v
4	Jembatan.png	613376	76672	4.txt	22528	v	v
5	Jembatan.png	613376	76672	5.txt	28672	v	v
6	Payung.png	657408	82176	6.txt	30720	v	v
7	Payung.png	657408	82176	7.txt	30720	v	v
8	Payung.png	657408	82176	8.txt	36864	v	v
9	bumi.png	681984	85248	9.txt	39936	v	v
10	bumi.png	681984	85248	10.txt	50176	v	v

Tabel 5. Hasil aspek recovery LSB

No.	Cover File			Secret file		Proses	
	Nama File	Ukuran (Bytes)	Kapasitas Maksimum (Bytes)	Nama File	Ukuran (Bytes)	Encode	Decode
1	Mobil.png	420864	52608	1.txt	8253	v	v
2	Mobil.png	420864	52608	2.txt	13516	v	v
3	Jembatan.png	613376	76672	3.txt	15360	v	v
4	Jembatan.png	613376	76672	4.txt	22528	v	v
5	Jembatan.png	613376	76672	5.txt	28672	v	v
6	Payung.png	657408	82176	6.txt	30720	v	v
7	Payung.png	657408	82176	7.txt	30720	v	v
8	Payung.png	657408	82176	8.txt	36864	v	v
9	bumi.png	681984	85248	9.txt	39936	v	v
10	bumi.png	681984	85248	10.txt	50176	v	v

4. KESIMPULAN

Kesimpulan pada penelitian ini adalah sebagai berikut:

1. Metode LSB memiliki nilai MOS rata-rata 2 (*poor*), artinya *stego file* yang dihasilkan memiliki *noise* yang mengganggu, sedangkan metode MSB memiliki nilai MOS rata-rata 4 (*good*), artinya terdapat *noise* namun tidak mengganggu.
2. Presentase perubahan ukuran file yang terjadi dengan menggunakan metode MSB adalah 50.5% sedangkan metode LSB adalah 47.78%.
3. Metode MSB memiliki nilai rata-rata MSE sebesar 696.24, artinya tingkat kesalahan atau kerusakan citra sangat besar, sedangkan metode LSB memiliki nilai rata-rata MSE sebesar 0.06, artinya tingkat kesalahan atau kerusakan citra sangat kecil.
4. Nilai PSNR yang didapat oleh metode MSB adalah 20.01dB, menunjukkan bahwa kualitas citra sangat buruk, dan berbeda dengan citra originalnya. Metode LSB memiliki nilai rata-rata 61.78 dB, menunjukkan bahwa kualitas citra yang dihasilkan baik dan mirip sesuai originalnya.

DAFTAR RUJUKAN

Ardhyana, A., Stavia, Juarna Asep. (2008). "*Implementasi Teknik Steganografi Dengan Metode Lsb Pada Citra Digital".*

Fauzi, Adji, Zulfiqar, Dewi Rosmala. (2010). *"PembangunanAplikasiMultimedia steganografi sebagaiteknik perlindungandata rahasiamenggunakanmetodeLeast Significant Bit"*. Institut Teknologi Nasional.

Nurfadli, Ahmad (2009). (Diakses online) "*Teknik Sampling".* mistercela21.wordpress.com.

Popa, Richard (1998). "*Analysis of Steganographic Techniques".*
Journal of University of Timisoara.

Rahul, Lokesh, Salony (2013). *"Image Steganograhy with LSB".* International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 2, Issue 1.

Rohit, Tarun (2012). "*Comparison of LSB & MSB Based Steganography in Gray-Scale Images".*

Smitha, B., Navas, K.A. (2007). "*Spatial Domain-High Capacity Data Hiding in ROI Images".* India. IEEE-ICSN-2007, pp.528-533