Penyisipan Pesan Pada Citra Digital Menggunakan Metode Least Significant Bit

Muhammad Azlansyah dan Budi Setiyono
Departemen Matematika, Fakultas Matematika Komputasi dan Sains Data,
Institut Teknologi Sepuluh Nopember (ITS)
e-mail: mike09jhoni@gmail.com

***Abstrak*—Berbagai macam teknik untuk melindungi informasi yang dirahasiakan telah banyak dilakukan. Steganografi adalah salah satu teknik yang digunakan dalam penyembunyian pesan kedalam sebuah media sedemikian sehingga manusia sulit menyadari keberadaan pesan tersebut. Pada studi ini teknik steganografi yang digunakan metode *Least Significant Bit* (LSB). Metode *Least Significant* Bit (LSB) adalah metode menyembunyikan pesan pada bit terakhir citra digital sehingga tidak terjadi perubahan secara kasat mata. Pada metode *Least Significant Bit* (LSB) dibutuhkan 3 input berupa citra, pesan, dan kunci. Pertama kunci dan pesan dirubah kedalam bentuk ASCII, setelah itu dirubah kedalam bentuk biner. Kemudian diambil channel merah (red) dari citra yang dirubah kedalam bentuk biner. Selanjutnya pesan dan kunci disisipkan pada citra tersebut. Terakhir citra yang telah disisipkan pesan disimpan dengan nama *Stego-Image*. Untuk uji coba paenulis menggunakan 3 citra dengan karakteristik yang berbeda, yaitu: citra langit yang cenderung homogen, citra wajah yang sedikit komplek, dan citra pemandangan yang lebih komplek. Disamping itu uji coba juga dilakukan pada 4 tipe citra yang berbeda, yaitu bmp, jpeg, png, dan tiff. Hasil yang dicapai pada studi ini, citra yang disisipkan pesan tidak mengalami perubahan bentuk secara kasat mata, nilai rata-rata PSNR dari keempat tipe citra tersebut untuk penyisipan teks 250 kata adalah sebagai berikut: citra langit 68.22975 dB, citra wajah 72.228575 dB, citra pemandangan 74.322525 dB dan pesan yang disisipkan dapat dikembalikan seperti semula saat proses ekstraksi.**

***Kata Kunci*—Steganografi, Metode *Leas Significant Bit* (LSB), Citra Digital.**

I. PENDAHULUAN

DI dunia modern seperti sekarang ini, internet dan komputer telah banyak digunakan di banyak tempat sebagai alat komunikasi. Dengan adanya internet, kita dapat berkomunikasi secara mudah dari berbagai tempat, dimanapun, dan kapanpun. Akan tetapi, berkomunikasi jarak jauh mempunyai resiko yang besar. Kita tidak mengetahaui apakah pesan kita benar-benar sampai ke tangan orang yang kita tuju kemungkinan lain, data akan diretas oleh oknum untuk kepentingan yang tidak seharusnya. Hal inilah yang menuntut adanya pengamanan data agar tidak sampai dicuri oleh pihak lain. Oleh karena itu, pengguna teknologi semakin ramai mengembangkan suatu system pengamanan terhadap data yang biasa disebut kriptografi.

Dalam kriptografi muncul istilah steganograi, yaitu suatu teknik menyisipkan pesan ke dalam suatu media. Walaupun steganografi masih berkaitan dengan kriptografi, tapi teknik ini sangat berbeda. Kriptografi mengacak pesan sehingga tidak dimengerti apa isi pesan tersebut, sedangkan steganografi mnyembunyikan pesan sehingga tidak terlihat. Pesan dalam *ciphertext* hasil proses kriptografi mungkin akan menimbulkan kecurigaan, namun tidak pada pesan yang dibuat dengan steganografi.

Steganografi merupakan ilmu yang mempelajari tentang seni menyembunyikan pesan atau informasi. Steganografi dapat digolongkan sebagai salah satu bagian dari ilmu komunikasi. Pada era informasi digital, steganografi merupakan teknik dan seni menyembunyikan informasi dan data digital dibalik informasi digital lain, sehingga informasi digital yang sesungguhnya tidak kelihatan. Seni dan ilmu ini telah diterapkan sejak dahulu oleh orang Yunani kuno yang menyembunyikan pesan dengan cara membuat tato di kepala pembawa berita yang dibotaki dan menunggu sampai rambutnya tumbuh.

Teknik steganografi lainnya adalah dengan menggunakan *invisible ink* (tinta yang tidak tampak). Tulisan yang ditulis dengan menggunakan *invisible ink* ini hanya dapat dibaca jika kertas tersebut diletakkan di atas lampu atau diarahkan ke matahari. Ketika perang dunia pertama, orang Jerman menyembunyikan pesan dalam bentuk *microdot*, yaitu titik-titik yang kecil. Agen dapat membuat foto kemudian mengecilkannya sampai sekecil titik di tulisan dalam buku. Buku ini kemudian bisa dibawa-bawa tanpa ada yang curiga bahwa tanda titik di dalam tulisan dibuku itu berisi pesan ataupun citra.

Dalam steganografi ada beberapa metode yang dapat digunakan salah satunya adalah metode *Least Significant Bit(LSB)*. Metode *LSB* merupakan teknik penyisipan pesan dalam steganografi dimana penyisipan pesan dilakukan dengan mengganti deretan bit-bit data yang paling belakang dalam segmen citra dengan deretan bit-bit pesan yang akan disisipkan.

II. DASAR TEORI

A. Citra Digital

Citra digital dapat didefinisikan sebagai sebuah fungsi 2 dimensi, $f(x,y)$ dimana x dan y adalah koordinat bidang datar

dan harga fungsi *f* disetiap pasangan koordinat (*x,y*) disebut intensitas/level keabuan (*gray level*) dari gambar dititik itu [1]. Citra digital merupakan suatu matriks dimana indeks baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar / *pixel* / piksel / pels / *picture element*) menyatakan tingkat keabuan pada titik tersebut [2]. Matriks pada citra digital berukuran *M* (baris/tinggi) x *N* (kolom/lebar).

B. Citra Warna

Red (Merah), *Green* (Hijau) dan *Blue* (Biru) merupakan warna dasar yang dapat diterima oleh mata manusia. Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari ketiga warna dasar RGB. Setiap titik pada citra warna membutuhkan data sebesar 3 byte. Setiap warna dasar memiliki intensitas tersendiri dengan nilai minimum nol (0) dan nilai maksimum 255 (8 bit). Citra high color biasanya disebut citra warna 16 bit setiap pixelnya diwakili dengan 2 byte memory. Citra 16 bit memiliki warna 65.536 warna. Dalam formasi bitnya, nilai red dan blue mengambil tempat di 5 bit kanan dan kiri. Komponen green memiliki 5 bit ditambah 1 bit ekstra.

C. Steganografi

Kata steganografi berasal dari bahasa yunani, *steganos* yang artinya tersembunyi dan *graphien* yang artinya tulisan yang dapat diterjemahkan menjadi tulisan yang tersembunyi. Menurut Munir bahwa Steganografi didefinisikan sebagai ilmu dan seni untuk menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan pesan tidak terdeteksi oleh indera manusia. Media yang digunakan umumnya merupakan suatu media yang berbedadengan media pembawa informasi rahasia, disinilah fungsi dari teknik *steganography* yaitu sebagai teknik penyamaran menggunakan media lain yang berbeda sehingga informasi rahasia dalam media awal tidak terlihat secara jelas. Umur steganografi hamper sama tuanya dengan kriptografi. Steganografitertua ditulis oleh Herodatus (485 – 525 BC), sejarawan Yunani padatahun 440 BC di dalam buku *Histories of Herodatus*. Kisah perangantara kerajaan Persia dan rakyat Yunani.Herodatus menceritakan cara Histaiaeus mengirim pesan kepadaAristagoras dari Miletusuntuk melawan Persia.Caranya:dipilih beberapa budak. Kepala budak dibotaki, ditulisi pesan dengan cara tato, rambut budak dibiarkan tumbuh, budak dikirim dan di tempat penerima kepala budak digunduli agar pesan bisa dibaca [3]. Ada juga penggunaan steganografi pada perang dunia II dengan cara menggunakan teknik microdot. Pesan yang akan dikirim diperkecil sampai hanya terlihat seperti 1 titik. Kemudian titik itu disisipkan ke dalam buku. Sehingga sipembawa pesan seperti tidak membawa sesuatu hal yang rahasia.

Dalam penyembunyian pesan ada beberapa kriteria yang perlu diperhatikan:

1. *Imperceptibility*. Keberadaan pesan rahasia tidak dapat dipersepsi oleh indra.
2. *Fidelity*. Mutu media penampung tidak berubah banyak akibat penyisipan.
3. *Recovery*. Pesan yang disembunyikan harus dapat diungkap kembali agar sewaktu – waktu pesan rahasia dapat diambil kembali untuk digunakan lebih lanjut [4].

D. Metode Least Significant Bit

Least Significant Bit (LSB) merupakan salah satu teknik dalam Steganografi. LSB menambahkan bit data pesan yang akan disembunyikan di bit terakhir yang paling cocok atau kurang berarti. Misalkan bit pada *image* dengan ukuran 3 piksel sebagai berikut:

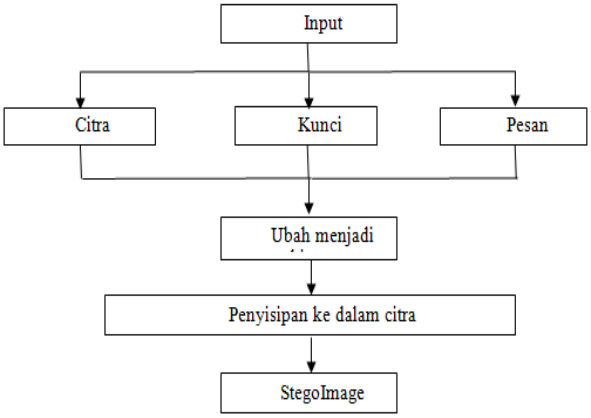
$$\begin{pmatrix} 0011111 & 11101001 & 11001000 \\ 0011111 & 11001000 & 11101001 \\ 1100000 & 00100111 & 11101001 \end{pmatrix}$$

Pesan yang akan disisipkan adalah karakter ‘A’ yang memiliki biner 10000001, *stego image* yang akan dihasilkan adalah:

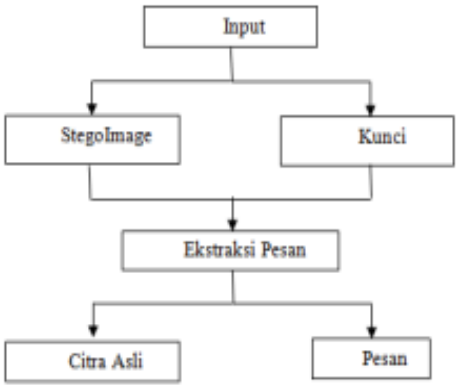
$$\begin{pmatrix} 0011111\mathbf{1} & 1110100\mathbf{1}\mathbf{0} & 1100100\mathbf{0} \\ 0011111\mathbf{0} & 1100100\mathbf{0} & 1110100\mathbf{1}\mathbf{0} \\ 1100000\mathbf{0} & 0010011\mathbf{1}\mathbf{1} & 1110100\mathbf{1}\mathbf{1} \end{pmatrix}$$

Ada dua teknik yang dapat digunakan pada LSB, yaitu penyisipan secara sekuensial dan secara acak. Penyisipan sekuensial dilakukan berurutan sedangkan acak dilakukan dengan acak pada *image* dengan memasukan kata kunci (*stego key*) [5].

E. Metodologi



Gambar 1. Diagram alir proses penyisipan teks ke dalam citra.



Gambar 2. Diagram alir proses ekstraksi pesan.

Gambar 1 merupakan diagram alir proses penyisipan teks ke dalam citra. User menginputkan pesan, kunci dan citra. Kemudian pesan, kunci dan piksel citra dirubah menjadi kode biner. Selanjutnya pesan dan kunci disisipkan ke bit akhir citra dan disimpan sebagai *StegoImage*. Gambar 2 merupakan diagram alir proses ekstraksi pesan. User menginputkan citra yang telah disisipkan pesan (*StegoImage*) dan Kunci. Selanjutnya dilakukan proses ektraksi pesan sehingga didapat citra asli dan data teks yang disisipkan.

III. PERANCANGAN DAN IMPLEMENTASI SISTEM

A. Analisa Sistem Perangkat Lunak

Sistem perangkat lunak yang dirancang ini memiliki beberapa tahapan:

1. Penyisipan Pesan

Untuk memperjelas metode ini, maka akan diberikan contoh sebagai berikut:

Misalkan terdapat pesan “jul” dan kunci “R” yang memiliki kode ASCII sebagai berikut:

106	117	108
j	u	l

82
R

Kemudian pesan dan kunci dirubah menjadi kode biner:

01101010	01110101	01101110
j	u	l

01010010
R

Misalkan citra yang akan digunakan sebagai wadah adalah citra “langit.jpg” (Gambar 3)

Gambar 3. Citra sebelum disisipkan pesan.

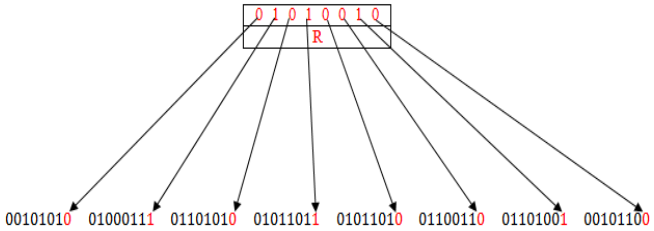
Dan misalkan diambil channel red (merah) untuk tempat menyisipkan pesan memiliki matriks 10 x 4 sebagai berikut:

$$\begin{bmatrix} 41 & 21 & 90 & 31 & 71 & 75 & 91 & 90 \\ 65 & 51 & 81 & 66 & 74 & 79 & 95 & 93 \\ 50 & 87 & 96 & 78 & 92 & 71 & 40 & 61 \\ 65 & 43 & 76 & 82 & 99 & 78 & 72 & 39 \\ 65 & 85 & 55 & 49 & 95 & 70 & 75 & 35 \\ 73 & 67 & 50 & 47 & 69 & 77 & 45 & 34 \end{bmatrix}$$

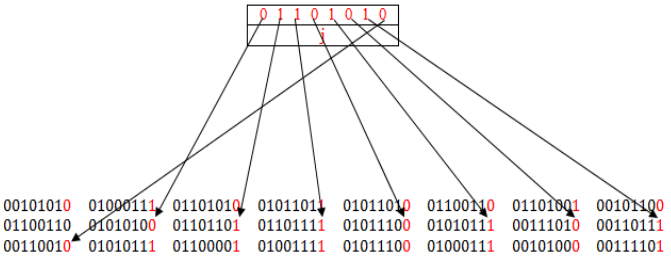
Kemudian matriks di atas dirubah menjadi bentuk biner

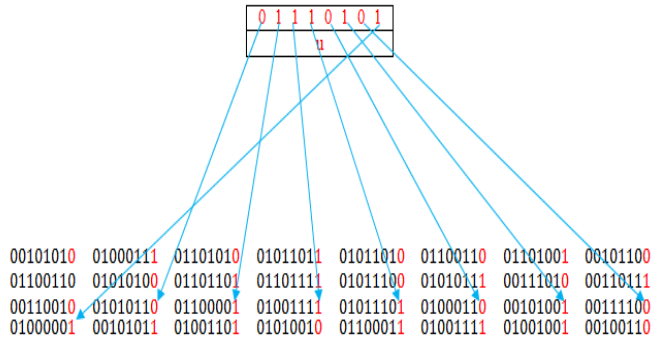
00101011 01000111 01101010 01011010 01011010 01100110 01101001 00101101
01100110 01010101 01101100 01101110 01011101 01010111 00111010 00110111
00110010 01010111 01100000 01001110 01011100 01000111 00101000 00111101
01000001 00101011 01001100 01010010 01100011 01001110 01001000 00100111
01000001 01010101 00110111 00110001 01011111 01000110 01001011 00100011
01001001 01000011 00110010 00101111 01000101 01001101 00101101 00100010

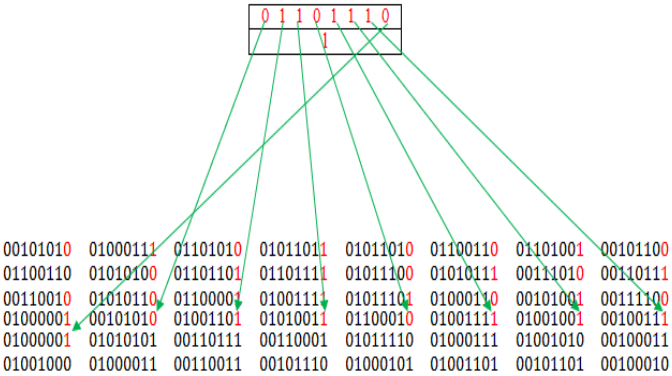
Selanjutnya kunci yang telah dirubah menjadi bentuk biner disisipkan ke elemen awal matriks citra yang telah dirubah menjadi kode biner



Selanjutnya penyisipan pesan yang telah dirubah menjadi bentuk matriks dimulai satu elemen setelah elemen terakhir kunci disisipkan







Kemudian matriks di atas dirubah kembali menjadi bentuk desimal

41	21	90	31	71	75	91	90
65	51	81	66	74	79	95	93
50	87	96	79	92	70	41	60
64	43	77	82	99	78	73	38
64	85	55	49	94	71	74	35
72	67	51	46	69	77	45	34

Terakhir simpan sebagai *StegoImage*.

Gambar 4. Citra sesudah disisipkan pesan

2. Ekstraksi Pesan

Pada proses ektraksi, citra yang diinputkanadalahcitra yang telahdisisipkanpesan (*Stego-Image*)

Gambar 5. Citra sesudah disisipkan pesan

$$\begin{bmatrix} 41 & 21 & 90 & 31 & 71 & 75 & 91 & 90 \\ 65 & 51 & 81 & 66 & 74 & 79 & 95 & 93 \\ 50 & 87 & 96 & 79 & 92 & 70 & 41 & 60 \\ 64 & 43 & 77 & 82 & 99 & 78 & 73 & 38 \\ 64 & 85 & 55 & 49 & 94 & 71 & 74 & 35 \\ 72 & 67 & 51 & 46 & 69 & 77 & 45 & 34 \end{bmatrix}.$$

Matriks di atas merupakan matriks dari *Stego-Image*, kemudian matriks dirubah menjadi bentuk biner menjadi:

00101010	01000111	01101010	01011011	01011010	01100110	01101001	00101100
01100110	01010100	01101101	01101111	01011100	01010111	00111010	00110111
00110010	01010110	01100001	01001111	01011101	01000110	00101001	00111100
01000001	00101010	01001101	01010011	01100010	01001111	01001001	00100111
01000001	01010101	00110111	00110001	01011110	01000111	01001010	00100011
01001000	01000011	00110011	00101110	01000101	01001101	00101101	00100010

Lalu inputkan kunci

82
R

Kemudian kunci dirubah menjadi bentuk biner

01010010
R

Selanjutnya sistem akan menyocokkan kunci yang diinputkan dengan kunci yang ada dalam piksel citra. Apabila kunci sesuai maka sistem mengambil bit-bit pesan yang disisipkan. Terakhir, sistem mengubah bit-bit pesan menjadi teks dan menampilkannya pada perangkat lunak.

3.1 Pemrogaman

a. Implementasi Masukkan Citra

Masukkan pada perangkat lunak ini berupa citra warna 16 bit. Proses tersebut diimplementasikan dalam *source code* berikut :

```
[filename,path]=uigetfile([{'*.bmp'};{'*.jpg'}
;{'*.tiff'};{'*.png'}],'open data');
ifisequal(filename,0)
return
end
% bacacitrasesuaidirektori
img=imread(fullfile(path,filename));
% tampilkancitra
axes(handles.axes1)
imshow(img)
% simpan data imgdlm figure
setappdata(handles.figure1,'img',img)
```

b. Implementasi Penyisipan Citra

Pada proses ini dilakukan penyisipan pesan setelah *user* memasukkan citra masukkan. Selanjutnya *user* memasukkan kunci dan pesan pada perangkat lunak dan proses penyisipan bisa dilakukan. Proses penyisipan diimplementasikan dalam *sourcecode* berikut:

```
k=0;
fori = 1 : size(c,2) % lebar citra
for j = 1 : size(c,1) % panjang citra
     k = k + 1;
ig=c(j,i); % baca nilai pixel citra baris ke I
dan kolom ke j
bg(k,:)=dec2bin(ig,8); % ubah kebiner
if (k <= size(b,1)) % sisipkan sebanyak nilai
biner (text)
bg(:,end)=num2str(b(k)); % sisipkan dibagian
akhir array
end
     % setelah penyisipan ubah kembali ke
nilai desimal (pixel)
     d=(bg(k,:));
hg(k,:)=bin2dec(d);
end
end

s=reshape(hg,size(c)); % ubah ukuran seperti
citra awal
s=uint8(s); % transpose dan ubah kenilai 8 bit
ky=size(b,1)/8; % ukuran panjang nilai biner
imgs(:,:,1)=s; % ubah chanel r (merah)
dengannilaihasilpenyisipan
imgs(end,end,1)=ky; % simpan jumlah nilai
biner
imgs(end,end-1,1)=l; % simpan jumlah panjang
kata
% tampilkan di axes 2
axes(handles.axes2)
imshow(imgs)
% simpan hasil penysisipan
[filename,path]=uiputfile([{'*.bmp'};{'*.jpg'}
;{'*.tiff'};{'*.png'}],'save stego');
ifisequal(filename,0)
```

```
return
end
if strcmp(filename(:,end-2:end),'bmp') ||
strcmp(filename(:,end-2:end),'iff')
imwrite(imgs,fullfile(path,filename));
else
imwrite(imgs,fullfile(path,filename),
'Mode','lossless');
end
```

c. Implementasi Ekstraksi Citra

Pada proses enkripsi masukan hanya ada 2 yaitu citra yang disisipi pesan dan kunci. Proses ekstraksi diimplementasikan dalam *source code* berikut :

```
% ekstraksi
for h=1:ky
si=s(h,:);% baca pixel yang disiipkan
db=dec2bin(si,8); % ubahkebiner
ds(h,:)=str2num(db(:,end)); % ubahkenumerik
end
% ubahkebiner string
for k = 1:size(ds,1);
ib=ds(k);
if(ib == 1)
es(k) = '1';
else
es(k) = '0';
end
end
% ubah kedalam ukuran biner awal
tx=reshape(es,[8,l]);
% ubah nilai biner kedesimal kemudian di ubah
ke text
fori=1:size(tx,2)
thisString=char(tx(:,i))'; % baca nilai biner
string
thisChar(:,i) = char(bin2dec((thisString))); %
ubah dari biner kedesimal kemudian ke text
(string)
end
```

IV. ANALISIS DAN PEMBAHASAN

A. Data Uji Coba

Uji coba pada perangkat lunak dalam studi ini dilakukan terhadap 3 citra warna yang berbeda dan panjang kunci dan pesan yang disisipkan juga ada 3 jenis (Tabel 1).

B. Pengujian Kualitatif

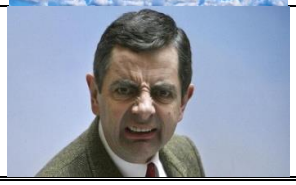
Pengujian kualitatif dilakukan berdasarkan pengamatan visual dari citra yang disisipi pesan (*Stego-Image*). Dalam penelitian ini pengujian kualitatif dibagi menjadi 4 kriteria, yaitu:

1. Baik Sekali : *Stego-Image* tidak dapat dibedakan dengan citra asli.
2. Baik : *Stego-Image* dapat dibedakan dengan citra asli dengan pengamatan yang teliti.
3. Jelek : *Stego-Image* yang disisipi dapat dibedakan dengan citra asli dengan mudah.
4. Jelek Sekali : perbedaannya telihat jelas.

C. Pengujian Penyisipan dan Ekstraksi Pesan

Pada subbab ini dilakukan proses penyisipan dan ekstraksi pesan untuk menguji apakah kedua hal tersebut dapat dilakukan untuk semua format citra. Hasil dari pengujian tersebut disajikan pada Tabel 2:

Tabel 1.
Data citra yang digunakan

No.	Nama	Resolusi	Citra
1	Woodstock vermont.jpg	736 x 490	
2	Langit Biru.jpg	350 x 229	
3	Bean.jpg	454 x 542	

Tabel 2.
Hasil pengujian system pada proses penyisipan dan ekstraksi untuk semua format citra

Fomat Citra	Penyisipan	Ektraksi
Png	Berhasil	Berhasil
Bmp	Berhasil	Berhasil
Jpg	Berhasil	Berhasil
Tif	Berhasil	Berhasil

D. Pengujian Nilai PSNR (Peak Signal to Noise Ratio)

Tabel 3.
Hasil pengujian nilai *PSNR* pada pengujian Pertama

No.	Nama	Format	*Stegoimage*	*PSNR*
1	stego vermont 1	Bmp	1.03 MB	80.7642 dB
2	stego vermont 2	Jpg	785 KB	80.7716 dB
3	stego vermont 3	Tiff	1.03MB	81.7241 dB
4	stego vermont 4	Png	624 KB	81.7241 dB
5	stego Bean 1	Bmp	197 KB	78.8726 dB
6	stego Bean 2	Jpg	85.6 KB	78.8726 dB
7	stego Bean 3	Tiff	88 KB	78.7162 dB
8	stego Bean 4	Png	100 KB	78.7162 dB
9	stego Langit Biru 1	Bmp	722 KB	74.2547 dB
10	stego Langit Biru 2	Jpg	15.5KB	74.2621 dB
11	stego Langit Biru 3	Tiff	721KB	83.1329 dB
12	stego Langit Biru 4	Png	191KB	83.1329 dB

Tabel 4.
Hasil pengujian nilai *PSNR* pada pengujian Kedua

No.	Nama	Format	*Stegoimage*	*PSNR*
1	stego vermont 1	Bmp	1.03 MB	77.8809 dB
2	stego vermont 2	Jpg	785 KB	77.8961 dB
3	stego vermont 3	Tiff	1.03MB	77.6092 dB
4	stego vermont 4	Png	624 KB	77.6092 dB
5	stego Bean 1	Bmp	197 KB	75.9648 dB
6	stego Bean 2	Jpg	85.6 KB	75.9648 dB
7	stego Bean 3	Tiff	88 KB	75.1251 dB
8	stego Bean 4	Png	100 KB	75.1251 dB
9	stego Langit Biru 1	Bmp	722 KB	71.3568 dB
10	stego Langit Biru 2	Jpg	15.5KB	71.3567 dB
11	stego Langit Biru 3	Tiff	721KB	71.8211 dB
12	stego Langit Biru 4	Png	191KB	71.7211 dB

Tabel 5.
Hasil pengujian nilai *PSNR* pada pengujian ketiga

No.	Nama	Format	*Stegoimage*	*PSNR*
1	stego vermont 1	Bmp	1.03 MB	74.3684 dB
2	stego vermont 2	Jpg	785 KB	74.3819 dB
3	stego vermont 3	Tiff	1.03MB	74.2699 dB
4	stego vermont 4	Png	624 KB	74.2699 dB
5	stego Bean 1	Bmp	197 KB	72.5819 dB
6	stego Bean 2	Jpg	85.6 KB	72.587 dB
7	stego Bean 3	Tiff	88 KB	71.8727 dB
8	stego Bean 4	Png	100 KB	71.8727 dB
9	stego Langit Biru 1	Bmp	722 KB	68.0512 dB
10	stego Langit Biru 2	Jpg	15.5KB	68.037 dB
11	stego Langit Biru 3	Tiff	721KB	68.4154dB
12	stego Langit Biru 4	Png	191KB	68.4154dB

Pengujian PNSR (*Peak Signal to Noise Ratio*) digunakan untuk mengukur kualitas citra yang dihasilkan. Metode PNSR adalah ukuran perbandingan antara nilai piksel *citra awal* dengan nilai piksel pada *StegoImage* yang dihasilkan. Hasil dari pengujian nilai *PSNR* disajikan pada Tabel 3-5:

Berdasarkan pengujian *PSNR* didapatkan bahwa nilai *PSNR* di atas 30dB, berarti kualitas antara citra asli dengan *StegoImage* tidak mengalami perubahan yang signifikan. Jadi keberadaan dari file yang tersembunyi tidak mudah terdeteksi oleh indra penglihatan manusia.

V. KESIMPULAN DAN SARAN

Berdasarkan analisisi dari hasil pengujian perangkat lunak steganografi menggunakan metode *Least Significant Bit(LSB)*ini, dapat ditarik kesimpulan sebagai berikut :

1. Hasil dari penerapan untuk penyisipan pesan rahasia pada gambar berjalan dengan baik. Pesan yang disisipkan pada citra dapat diperoleh kembali secara utuh atau dengan kata lain pesan yang disisipkan sebelum proses penyisipan dan setelah proses ekstraksi mempunyai hasil yang sama tanpa ada perubahan pesan.
2. Citra yang telah disisipkan pesan (*StegoImage*) tidak mengalami perubahan yang signifikan dengan citra asli, sehingga secara kasat mata tidak dapat diketahui bahwa terdapat pesan rahasia pada citra tersebut.
3. Dari uji coba diperoleh hasil bahwa citra yang lebih komplek mempunyai hasil yang lebih baik, terbukti dengan memiliki rata-rata nilai PSNR 74.322525 dB.

Berdasarkan hasil yang dicapai pada penelitian ini, ada beberapa hal yang penulis sarankan untuk pengembangan selanjutnya yaitu:

1. Pada penelitian ini penulis hanya menggunakan media citra digital sebagai media penampung, diharapkan untuk penelitian selanjutnya menggunakan media audio, video, dan lain sebagainya.
2. Program masih menyisipkan pesan dalam bentuk *plaintext* pada penelitian berikutnya diharapkan pesan tersebut terenkripsi.

DAFTAR PUSTAKA

[1] B. Rakmat and M. Fairuzabadi, “Steganografi menggunakan Metode Least Significant Bit dengan Kombinasi Algoritma Kriptografi Vigneredan RC4,” *J. Din. Inform.*, vol. 5, no. 2, 2010.

[2] D. Chopra, “Lsb Based Digital Image Watermarking For Gray Scale Image,” *IOSR J. Comput. Eng.*, vol. 6, no. 1, pp. 36–41, 2012.

[3] Irfan, “Penyembunyian Informasi (steganography) Gambar Menggunakan Metode LSB (Least Signifcant Bit),” *Rekayasa Teknol.*, vol. 5, no. 1, 2013.

[4] A. Muadzani, O. Nurhayati, and I. P. Windasari, “Penyisipan Media Teks dan Citra Menggunakan Teknik Steganografi pada Media Pembawa Citra Digital,” *J. Teknol. dan Sist. Komput.*, vol. 4, no. 3, pp. 470–478, 2016.

[5] M. Ineke Pakereng, Y. Richard Beeh, S. Endrawan, and U. Kristen Duta Wacana Yogyakarta, “Perbandingan Steganografi Metode Spread Spectrum dan Least Significant Bit (LSB) Antara Waktu Proses dan Ukuran File Gambar.”