IMPLEMENTASI STEGANOGRAFI PADA CITRA *DIGITAL* DENGAN METODE *LEAST SIGNIFICANT BIT*

PROPOSAL SKRIPSI

Disusun untuk melengkapi syarat-syarat guna memperoleh gelar
Sarjana Komputer

Oleh:
AMELIA APRILIANI
3145143626

PROGRAM STUDI ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA

2018

LEMBAR PERSETUJUAN

Dengan ini saya mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta

Nama	: Amelia Apriliani
No. Registrasi	: 3145143626
Jurusan	: Ilmu Komputer
Judul	: Implementasi Steganografi pada Citra *Digital* dengan Metode *Least Significant Bit*.

Menyatakan bahwa proposal ini telah siap diajukan untuk seminar pra skripsi.

Menyetujui,

Dosen Pembimbing I	Dosen Pembimbing II
Drs. Mulyono, M.Kom.	**Ratna Widyati, S.Si, M.Kom.**
NIP. 119660517 199403 1 003	NIP. 19750925 200212 2 002

Mengetahui,

Koordinator Program Studi Ilmu Komputer

Drs. Mulyono, M.Kom.

NIP. 119660517 199403 1 003

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

LATAR BELAKANG

1.1 Latar Belakang Masalah

Saat ini *internet* sudah berkembang menjadi salah satu media yang sangat populer di berbagai dunia [4]. Perkembangan *internet* memberikan pengaruh besar terhadap kemudahan dalam berkomunikasi dan menyampaikan informasi. Komunikasi merupakan salah satu hal yang penting bagi manusia. Manusia yang merupakan makhluk sosial cenderung melakukan komunikasi setiap hari, baik secara langsung maupun melalui media elektronik. Manusia melakukan komunikasi untuk bertukar informasi.

Kemudahan dalam berkomunikasi memberikan dampak positif dan negatif. Dampak positifnya yaitu cepatnya informasi dapat tersebar, baik antar daerah maupun antar negara. Dan dampak negatifnya adalah semakin berkembangnya kejahatan dalam penggunaan informasi. Dengan berbagai teknik, banyak orang yang mencoba untuk mengakses informasi yang bukan haknya. Maka dari itu harus berkembang juga pengamanan sistem informasi.

Teknik pengamanan informasi yang ada saat ini seperti kriptografi dan steganografi. Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikan pesan ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Kriptografi telah ada dan digunakan sejak berabad-abad yang lalu dikenal dengan istilah kriptografi klasik, yang bekerja pada mode karakter alfabet [19].

Steganografi adalah seni dan sains komunikasi pesan yang tidak terlihat. Hal ini dilakukan dengan menyembunyikan informasi dalam informasi lain, misalnya menyembunyikan keberadaan informasi yang dikomunikasikan. Kata steganografi bera-

sal dari kata Yunani "stegos" yang berarti "cover" dan "grafia" yang berarti "menulis" yang mendefinisikannya sebagai "tulisan tertutup" [12].

Salah satu metode steganografi adalah *Least Significant Bit* (LSB). Algoritma LSB, menggantikan bit paling signifikan pada *file cover* sesuai dengan bit pesan. Teknik ini adalah teknik yang paling populer digunakan dalam steganografi untuk menyembunyikan pesan. Teknik ini biasanya efektif, karena substitusi LSB tidak menyebabkan degradasi kualitas yang signifikan [10].

Pengimplementasian metode Least Significant Bit pada steganografi sudah pernah dilakukan penelitian oleh Fahri Perdana Prasetyo dengan format file *.TIFF menggunakan bahasa pemrograman MATLAB [17]. Selain itu juga pernah dilakukan penelitian oleh Adiria dengan format file *.BMP menggunakan bahasa pemrograman Delphi [1]. Sedangkan yang akan penulis buat nantinya adalah dengan mengkombinasikan kedua penelitian tersebut.

Dengan penjabaran di atas, penulis mengkombinasikan jurnal-jurnal tersebut untuk melakukan penelitian tentang "**Implementasi Steganografi pada Citral Digital dengan Metode *Least Significant Bit***". Dengan adanya penelitian ini diharapkan dapat memberikan informasi mengenai steganografi.

1.2 Batasan Masalah

Batasan masalah dalam tugas akhir ini mencakup:

- Format *file* citra *digital* yang dapat digunakan untuk menyimpan pesan adalah berformat *.bmp.
- Format *file* citra *digital* yang dihasilkan dari program steganografi ini adalah berformat *.bmp.
- Pesan yang dapat disimpan hanya berformat *.txt.

1.3 Rumusan Masalah

Rumusan masalah berdasarkan latar belakang di atas adalah:

1. Bagaimana cara mengimplementasikan steganografi dengan metode *Least Significant Bit* ke dalam citra *digital*?
2. Bagaimana perubahan dalam *file* citra hasil keluaran sebelum dan sesudah disisipkan pesan teks?

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Memberikan informasi bagaimana steganografi dapat diimplementasikan ke dalam citra *digital* dengan menggunakan metode *Least Significant Bit*.
2. Mengetahui perubahan yang terjadi dari hasil keluaran *file* citra *digital*.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Bagi Penulis, diharapkan dapat menambah pengetahuan dan pemahaman tentang steganografi.
2. Bagi Program Studi Ilmu Komputer, Penulisan penelitian ini memberikan gambaran bagi seluruh mahasiswa khususnya bagi mahasiswa program studi Ilmu Komputer Universitas Negeri Jakarta tentang bagaimana stegaografi dalam *file* citra *digital*.

3. Bagi Masyarakat, diharapkan dapat menjadi salah satu solusi dalam mengamankan *file* mereka dari orang-orang yang tidak mempunyai hak untuk melihatnya.

1.6 Jenis Penelitian

Jenis Penelitian yang dijalani oleh Peneliti berjenis Kajian Teori. Jenis penelitian ini mengarahkan penulis kepada penerapan metode *Least Significant Bit* dalam pengembangan steganografi pada citra *digital*.

BAB II

KAJIAN TEORI

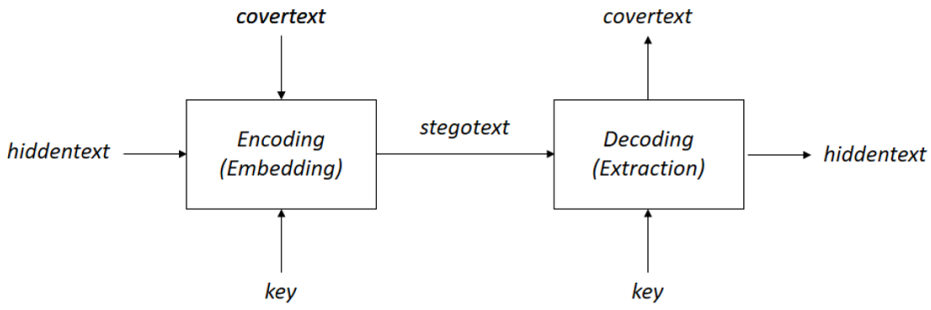
2.1 Steganografi

2.1.1 Pengertian Steganografi

Menurut **Gary C. Kessler** dalam jurnalnya *Steganography Hiding Data Within Data*:

"Steganografi adalah ilmu menyembunyikan informasi. Tujuan steganografi adalah untuk menyembunyikan data dari pihak ketiga." [11].

Secara umum, steganografi adalah seni untuk menyembunyikan pesan ke dalam media lain sedemikian rupa sehingga membuat orang lain tidak menyadari adanya pesan di media tersebut.



Gambar 2.1: Diagram penyisipan dan ekstraksi pada pesan

Istilah di dalam steganografi:

1. *Covertext* merupakan media atau tempat pesan yang digunakan untuk menyembunyikan *hiddentext*. *Covertext* bisa berupa teks, gambar, audio, video, dll.

2. *Hiddentext* atau biasa disebut *embedded message* merupakan pesan atau informasi yang ingin disembunyikan. Contohnya bisa berupa teks, gambar, audio, video, dll.

3. *Stegotext* merupakan pesan yang sudah berisi *embedded message*.

4. *Encoding* yaitu penyisipan pesan ke dalam media *covertext*.

5. *Decoding* yaitu ekstraksi pesan dari *stegotext*.

Menurut **Munir**, ada kriteria yang harus diperhatikan dalam penyembunyian pesan, yaitu meliputi *Imperceptible*, *Fidelity*, *Recovery* dan *Capacity*.

1. *Imperceptible*

   Keberadaan pesan rahasia tidak dapat dipersepsi secara visual atau secara audio. Jika *covertext* berupa *file* citra, maka *stegotext* yang dihasilkan harus sukar dibedakan oleh kasat mata dengan *covertext*-nya. Dan jika *covertext* berupa *file* audio, maka telinga tidak dapat mendeteksi perubahan yang ada pada audio *stegotext*-nya.

2. *Fidelity*

   Kualitas *file* citra penampung tidak jauh berubah. Setelah penambahan pesan rahasia, citra hasil steganografi masih terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam citra tersebut terdapat pesan rahasia.

3. *Recovery*

   Pesan yang disembunyikan harus dapat diekstrak kembali. Karena tujuan steganografi adalah menyembunyikan pesan atau informasi, maka jika informasi itu dibutuhkan harus dapat diambil kembali untuk dapat digunakan.

4. *Capacity*

   Ukuran pesan yang akan disembunyikan sedapat mungkin besar. Agar dapat memaksimalkan manfaat dari steganografi itu sendiri [14].

2.1.2 Sejarah Steganografi

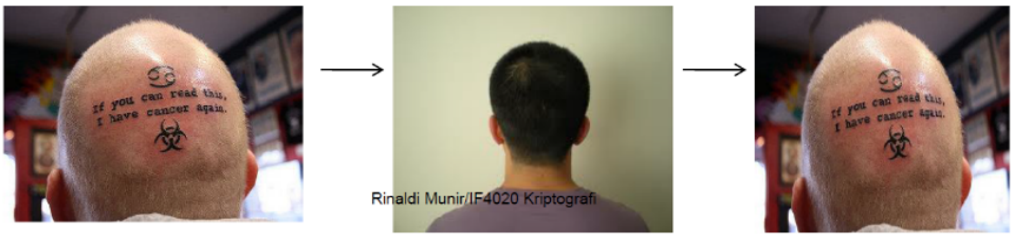
Seperti kriptografi, penggunaan steganografi sebetulnya telah digunakan berabad-abad yang lalu bahkan sebelum istilah steganografi itu sendiri muncul. Periode sejarah steganografi dapat dibagi menjadi:

1. Steganografi Kuno (*Ancient Steganography*)

   (a) Steganografi dengan media kepala budak

   Ditulis oleh **Herodatus** (485âĂŞ525 BC), sejarawan Yunani pada tahun 440 BC di dalam buku: *Histories of Herodatus*). Kisah perang antara kerajaan Persia dan rakyat Yunani. **Herodatus** menceritakan cara **Histaiaeus** mengirim pesan kepada **Aristagoras of Miletus** untuk melawan Persia.

   Caranya adalah dengan dipilih beberapa budak. Kemudian kepala budak tersebut digunduli dan ditulis pesan dengan cara ditato. Setelah pesan dituliskan, budak harus menunggu hingga rambutnya tumbuh kembali. Setelah rambut pada kepala budak tersebut tumbuh, budak dikirim ke tempat penerima. Di sana kepala budak digunduli agar pesan dapat dibaca.



Gambar 2.2: Steganografi dengan media kepala budak

(b) Penggunaan tablet *wax*

Orang-orang Yunani kuno menulis pesan rahasia di atas kayu yang kemudian ditutup dengan lilin (*wax*). Di dalam bukunya, **Heradatus** menceritakan **Demaratus** mengirim peringatan tentang serangan yang akan datang ke Yunani dengan menulis langsung pada tablet kayu yang kemudian dilapisi lilin dari lebah.

Gambar 2.3: Tablet *wax*

(c) Penggunaan tinta tak-tampak (*invisible ink*)

Pliny the Elder menjelaskan penggunaan tinta dari getah tanaman *thithymallus*. Jika dituliskan pada kertas maka tulisan dengan tinta terse-

but tidak kelihatan, tetapi bila kertas dipanaskan berubah menjadi gelap/coklat.

(d) Penggunaan kain sutra dan lilin

Orang Cina kuno menulis catatan pada potongan-potongan kecil sutra yang kemudian digumpalkan menjadi bola kecil dan dilapisi lilin. Selanjutnya bola kecil tersebut ditelan oleh si pembawa pesan. Pesan dibaca setelah bola kecil dikeluarkan dari perut si pembawa pesan.

2. Steganografi Zaman Renaisans (*Renaissance Steganography*)

Tahun 1499, **Johannes Trithemius** menulis buku *Steganographia*, yang menceritakan tentang metode steganografi berbasis karakter. Selanjutnya tahun 1518 dia menulis buku tentang steganografi dan kriptografi berjudul *Polygraphiae*. **Giovanni Battista Porta** menggambarkan cara menyembunyikan pesan di dalam telur rebus. Caranya, pesan ditulis pada kulit telur yang dibuat dari tinta khusus yang dibuat dengan satu ons tawas dan setengah liter cuka. Prinsipnya penyembunyiannya adalah tinta tersebut akan menembus kulit telur yang berpori, tanpa meninggalkan jejak yang terlihat. Tulisan dari tinta akan membekas pada permukaan isi telur yang telah mengeras (karena sudah direbus sebelumnya). Pesan dibaca dengan membuang kulit telur.

3. Steganografi Zaman Perang Dunia (*World War Steganography*)

Gambar 2.4: Steganografi zaman perang dunia

Selama terjadinya Perang Dunia ke-2, tinta yang tidak tampak (*invisible ink*) telah digunakan untuk menulis informasi pada lembaran kertas sehingga saat kertas tersebut jatuh di tangan pihak lain hanya akan tampak seperti lembaran kertas kosong biasa. Cairan seperti air kencing (*urine*), susu, vinegar, dan jus buah digunakan sebagai media penulisan sebab bila salah satu elemen tersebut dipanaskan, tulisan akan menggelap dan tampak melalui mata manusia [14].

4. Steganografi *Digital*

   Sejalan dengan perkembangan maka konsep awal steganografi diimplementasikan pula dalam dunia komputer, yang kemudian dikenal dengan istilah steganografi *digital*. Dalam hal ini, steganografi *digital* memiliki dua properti dasar yaitu media penampung (*cover data* atau *data carrier*) dan data *digital*

yang akan disisipkan (*secret data*), dimana media penampung dan data *digital* yang akan disisipkan dapat berupa *file* multimedia (teks/dokumen, citra, audio maupun video). Terdapat dua tahapan umum dalam steganografi *digital*, yaitu proses *embedding* atau *encoding* (penyisipan) dan proses *extracting* atau *decoding* (pemekaran atau pengungkapan kembali (*reveal*)). Hasil yang didapat setelah proses *embedding* atau *encoding* disebut *stego object* (apabila media penampung hanya berupa data citra maka disebut *stego image*) [18].

2.1.3 Metode Steganografi

Berdasarkan ranah operasinya, metode-metode steganografi dapat dibagi menjadi dua kelompok:

1. *Spatial (time) domain methods*

   Memodifikasi langsung nilai byte dari *cover-object* (nilai *byte* dapat merepresentasikan intensitas/warna *pixel* atau amplitudo). Contoh: Metode modifikasi LSB

2. *Tranform domain methods*

   Memodifikasi hasil transformasi sinyal dalam ranah transform (hasil transformasi dari ranah spasial ke ranah lain (misalnya ranah frekuensi). Contoh: Metode *Spread Spectrum* [14].

Ada empat jenis metode steganografi:

1. *Least Significant Bit Insertion* (LSB)

   Metode yang digunakan untuk menyembunyikan pesan pada media *digital* tersebut berbeda-beda. Contohnya, pada berkas *image* pesan dapat disembunyikan dengan menggunakan cara menyisipkannya pada bit rendah atau bit yang paling kanan (LSB) pada data *pixel* yang menyusun *file* tersebut. Pada berkas

bitmap 24 bit, setiap *pixel* (titik) pada gambar tersebut terdiri dari susunan tiga warna *Red*, *Green* dan *Blue* (RGB) yang masing-masing disusun oleh bilangan 8 bit (*byte*) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Dengan demikian, pada setiap *pixel* berkas *bitmap* 24 bit kita dapat menyisipkan 3 bit data.

2. *Algorithms and Transformation*

   Algoritma compression adalah metode steganografi dengan menyembunyikan data dalam fungsi matematika. Dua fungsi tersebut adalah *Discrete Cosine Transformation* (DCT) dan *Wavelet Transformation*. Fungsi DCT dan *Wavelet* yaitu mentransformasi data dari satu tempat (*domain*) ke tempat (*domain*) yang lain. Fungsi DCT yaitu mentransformasi data dari tempat *spatial* (*spatial domain*) ke tempat frekuensi (*frequency domain*).

3. *Redundant Pattern Encoding*

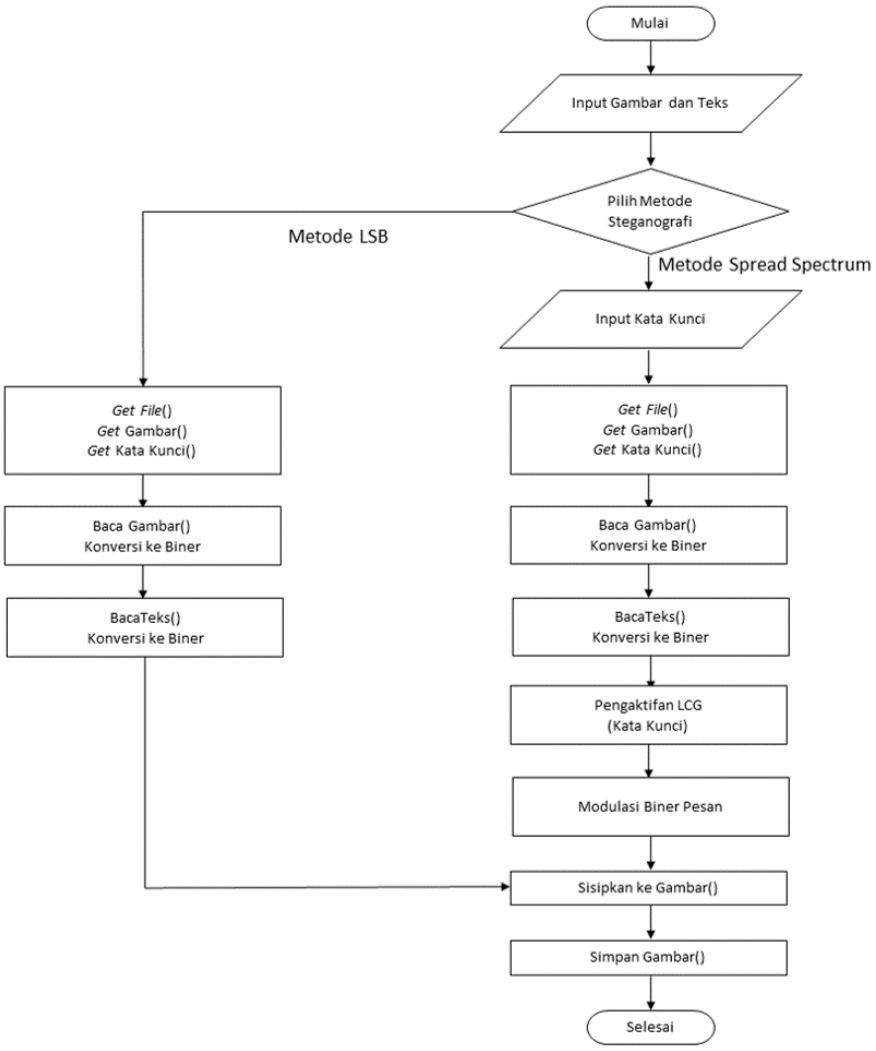
   Redundant Pattern Encoding adalah menggambar pesan kecil pada kebanyakan gambar. Keuntungan dari metode ini adalah dapat bertahan dari *cropping* (kegagalan). Kerugiannya yaitu tidak dapat menggambar pesan yang lebih besar.

4. *Spread Spectrum Method*

   Spread Spectrum steganografi terpencar-pencar sebagai pesan yang diacak (*encrypted*) melalui gambar (tidak seperti dalam LSB). Untuk membaca suatu pesan, penerima memerlukan algoritma yaitu *crypto-key* dan *stego-key*. Metode ini juga masih mudah diserang yaitu penghancuran atau pengrusakan dari kompresi dan proses *image* (gambar) [21].

Metode LSB dan *Spread Spectrum* adalah dua metode yang sering digunakan dalam melakukan steganografi. Selain karena metodenya yang sederhana, proses *encoding* dan *decoding* dari kedua metode tersebut juga *relative* cepat[16]. Tetapi

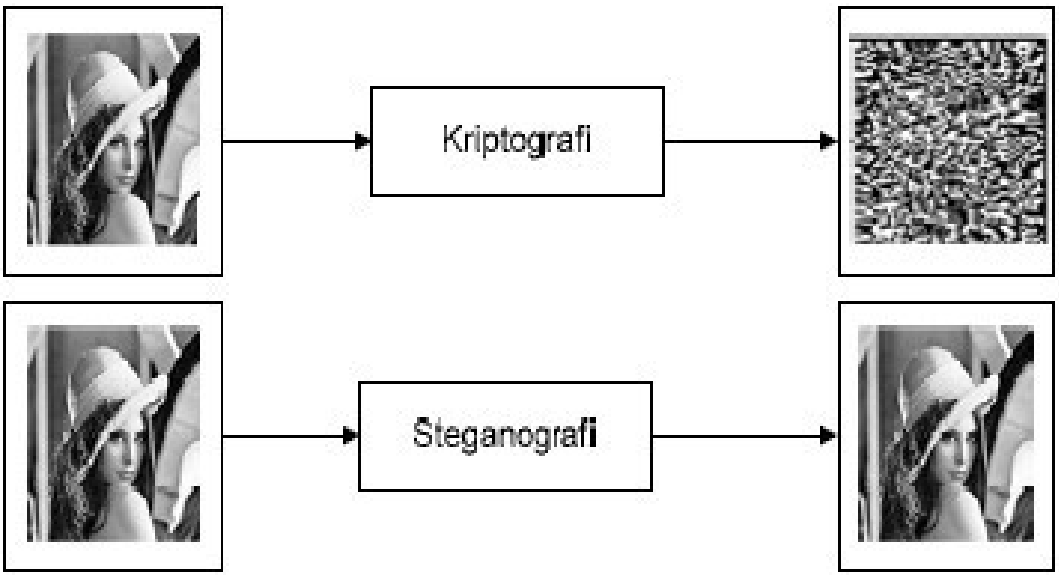
LSB memiliki proses *encoding* dan *decoding* yang lebih cepat dari metode *Spread Spectrum* karena proses metode *Spread Spectrum* harus melalui proses XOR antara pesan dan kata kunci, sedangkan LSB langsung menyisipkan pesan ke dalam gambar [15].



Gambar 2.5: *Flowchart Encoding* LSB dan *Spread Spectrum*

2.2 Perbedaan Steganografi dan Kriptografi

Steganografi dan kriptografi mempunyai prinsip kerja yang berbeda, meskipun keduanya mempunyai hubungan yang dekat dalam dunia keamanan data. Pada kriptografi menghasilkan sebuah *chipertext* dimana dengan itu seolah-olah dengan sengaja menunjukkan kepada orang lain bahwa ada sesuatu di dalamnya, namun tidak dapat diketahui maknanya. Namun dengan bentuk *chiper*-nya, justru akan membuat data tersebut terancam oleh usaha-usaha yang dilakukan oleh orang lain untuk dapat membongkarnya dengan tujuan dan atau alasan apapun.



Gambar 2.6: Perbedaan Kriptografi dan Steganografi

Steganografi dan kriptografi merupakan seni dan teknik yang dapat digunakan untuk melakukan pengamanan data *digital*. Namun keduanya tidaklah sama. Pada kriptografi, suatu data *digital* diamankan dengan cara mengenkripsi data tersebut dan menghasilkan sebuah data yang berupa sandi, secara visual data tersebut masih dapat terlihat atau diketahui, hanya saja data tersebut menjadi tidak dapat dimengerti. Berbeda dengan steganografi yang tujuannya adalah menyembunyikan data ke dalam

sebuah media yang lain, sehingga data tersebut tidak terlihat [20].

2.3 LSB (*Least Significant Bit*)

Penyembunyian data dilakukan dengan mengganti bit-bit data di dalam segmen citra dengan bit-bit rahasia. Pada susunan bit di dalam sebuah *byte* (1 *byte*= 8 bit), ada bit yang paling berarti (*most significant bit* atau MSB) dan bit yang paling kurang berarti (*least significant bit* atau LSB). LSB merupakan salah satu metode yang paling sederhanaa dalam steganografi. Bit yang cocok untuk diganti adalah bit LSB, sebab perubahan tersebut hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya [14].

Pada *file bitmap* 24 bit, setiap bit masing-masing memiliki komponen *Red, Green,* dan *Blue* (RGB), sehingga dapat menyimpan 3 bit pada setiap *pixel*-nya. Pada gambar 800x600 *pixel* dapat digunakan untuk menyembunyikan 1.440.000 bit (180.000 *Byte*) data rahasia. Sebagai contoh diambil 3 *pixel* dari *file bitmap* 24 bit yang akan disisipkan pesan atau data rahasia karakter "A":

(00001000 00101011 11011100)
(11100000 11000100 00010101)
(00010011 10101010 01100011)

Karakter "A" mempunyai nilai biner 01000001, maka bit hasil penyisipannya adalah:

(00001000 00101011 11011100)
(11100000 11000100 00010100)
(00010010 10101011 01100011

Bit-bit yang nilainya berganti ada 3 dalam 8 *Byte* yang digunakan. Contoh lainnya adalah diambil 8 pixel dari sebuah gambar, maka data rahasia yang dapat

dimasukkan adalah 1 kata, contohnya adalah "ADA"

(10011011 01100100 01010000)

(10010011 01010101 01001000)

(10011010 01010111 01001110)

(10011010 01010101 01010000)

(10001000 01000001 00111111)

(01101001 00100010 00110100)

(01101101 00100111 00110010)

(01111001 00110011 00110101)

Kata "ADA" mempunyai biner A = 01000001, D = 01000100, maka bit hasil penyisipannya adalah:

(1001101**0** 0110010**1** 01010000)

(1001001**0** 0101010**0** 01001000)

(10011010 01010111 01001110)

(1001101**1** 0101010**0** 01010000)

(10001000 01000001 0011111**0**)

(0110100**0** 00100010 0011010**1**)

(0110110**0** 0010011**0** 00110010)

(0111100**0** 0011001**0** 00110101)

Bit-bit yang nilainya berganti ada 13 dalam 24 *Byte* yang digunakan. Secara rata-rata, LSB hanya menggunakan setengah dari bit dalam gambar yang perlu dimodifikasi untuk menyembunyikan pesan rahasia. Perubahan ini tidak dapat dirasakan oleh mata manusia, dan pesan berhasil disembunyikan [6].

2.4 ASCII

ASCII adalah singkatan dari *American Standard Code for Information Interchange*. Komputer hanya dapat memahami angka, jadi kode ASCII adalah representasi numerik dari karakter seperti 'a' atau '@' atau karakter lainnya. Kode ASCII memiliki komposisi bilangan biner sebanyak 8 bit. Dimulai dari 00000000 hingga 11111111. Total kombinasi yang dihasilkan ASCII sebanyak 256, dimulai dari kode 0 hingga 255 dalam sistem bilangan desimal. [3]

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Tabel 2.1: Tabel ASCII

2.5 Citra *Digital*

2.5.1 Pengertian Citra *Digital*

Citra atau gambar dapat didefinisikan sebagai sebuah fungsi dua dimensi, f(x,y), x dan y adalah koordinat bidang datar; dan harga fungsi f di setiap pasang-

an koordinat (x,y) disebut intensitas atau level keabuan (*grey level*) dari gambar di titik itu [8]. Citra ada 2 macam, yaitu:

1. Citra kontinu, yaitu citra yang dihasilkan dari sistem optik yang menerima sinyal analog, misal: mata manusia dan kamera analog.

2. Citra diskrit, yaitu citra yang dihasilkan melalui proses digitalisasi terhadap citra kontinu.

Agar dapat diolah dengan komputer, maka suatu citra harus direpresentasikan secara *numeric* dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinyu menjadi nilai-nilai diskrit disebut digitalisasi, dan citra yang dihasilkan disebut citra *digital*.

Ada 3 bidang studi utama yang menangani pengolahan data atau informasi berbentuk gambar atau citra, yaitu:

1. Grafika Komputer (*Computer Graphics*)

2. Pengolahan Citra (*Image Processing*)

3. Pengenalan Pola (*Pattern Recognition*)

2.5.2 Pengolahan Citra (*Image Processing*)

Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik. Pengolahan Citra bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin (dalam hal ini komputer). Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain. Jadi, masukannya adalah citra dan keluarannya juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan [13]

2.5.3 Format *File* pada Citra *Digital*

1. BMP

   BMP adalah singkatan dari *Bitmap* yang dahulu dikembangkan oleh MICROSOFT. *Bitmap* dapat menyimpan data warna untuk masing-masing *pixel* dalam gambar tanpa kompresi apapun. Format ini dapat digunakan untuk menyembunyikan data tanpa menaikkan kecurigaan pada mata manusia. Gambar yang dihasilkan tanpa kompresi dan format *lossless* yang merupakan salah satu faktor penting. Ekstensi yang digunakan dalam *file* ini adalah .bmp [7].

2. JPEG

   Istilah JPEG sebenarnya adalah singkatan dari pengembangnya, yaitu *Joint Photographic Experts Group*. Gambar JPEG tidak terbatas pada sejumlah warna tertentu. Oleh karena itu, format JPEG paling baik untuk mengompresi gambar foto. Gambar dengan format JPEG dapat berisi data gambar beresolusi tinggi berwarna-warni, itu adalah format *lossy*, yang berarti beberapa kualitas hilang ketika gambar dikompresi. Jika gambar terlalu banyak dikompres, grafiknya menjadi seperti "tidak berwarna" dan sebagian detailnya hilang. Ekstensi yang digunakan dalam *file* ini adalah .jpeg [5].

3. GIF

   GIF adalah singkatan dari *Graphics Interchange Format* yang dikembangkan oleh COMPUSERVICE. GIF digunakan untuk tujuan menyimpan beberapa gambar *bitmap* dalam satu *file* gambar. GIF sering digunakan untuk menyimpan grafik multi-bit dan data gambar. GIF tidak terkait dengan aplikasi perangkat lunak tertentu tetapi dirancang untuk memudahkan pertukaran dan tampilan data gambar yang tersimpan di lokal atau sistem komputer jarak jauh. GIF digunakan juga karena menerapkan metode kompresi *lossless*. Ekstensi yang

digunakan dalam *file* ini adalah .gif [6].

4. TIFF

   TIFF adalah singkatan dari *Taged Image Format File*. TIFF dikembangkan oleh ADOBE dan digunakan untuk grafis berkualitas tinggi dengan kompresi *loss-less*. Format *file* ini memiliki transparansi dan pilihan warna terindeks untuk menanamkan pesan rahasia di atasnya. TIFF mendukung properti RGB dan *GRAYSCALE* dan digunakan untuk HD *Imaging*. Ini adalah salah satu format *file* paling serbaguna di antara semua format yang tersedia. Ekstensi yang digunakan dalam format *file* ini adalah .tiff [7].

5. PNG

   PNG adalah singkatan dari *Portable Network Graphics* yang dikembangkan oleh PNG *Development Group*. PNG mampu menyembunyikan pesan yang besar di dalamnya. Format *file* ini diciptakan untuk meningkatkan format *file* gambar GIF menghilangkan batasan 256 warna tetapi tidak mendukung animasi. Dan PNG menggunakan kompresi data *lossless*. Ekstensi yang digunakan dalam format *file* ini adalah .png [7].

Perbedaan komponen antara masing-masing format *file* citra *digital* dapat dilihat pada Tabel 2.2.

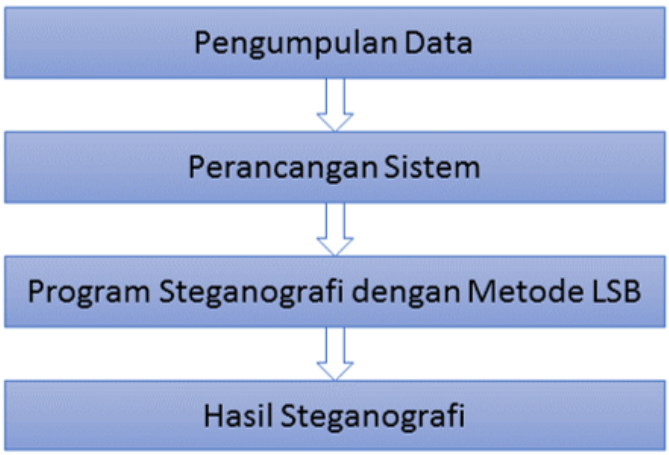
Komponen	BMP	JPEG	GIF	TIFF	PNG
Kompresi *Lossless*	Ya	Tidak	Ya	Ya	Ya
Grayscale	Ya	Ya	Ya	Ya	Ya
RGB	Terbatas	Ya	Ya	Ya	Ya
Index Pilihan Warna	Ya	Tidak	Ya	Ya	Ya
Transparansi	Tidak	Tidak	Ya	Tidak	Ya
Pilihan Animasi	Tidak	Tidak	Ya	Tidak	Tidak
Color bits	32	24	24	24, 48	24, 48

Tabel 2.2: Perbedaan *file* citra *digital*

BAB III

HASIL DAN PEMBAHASAN

Dalam penelitian ini tahapan yang akan dilakukan adalah seperti gambar di bawah ini



Gambar 3.1: Alur Penelitian

3.1 Pengumpulan Data

1. Studi Pustaka

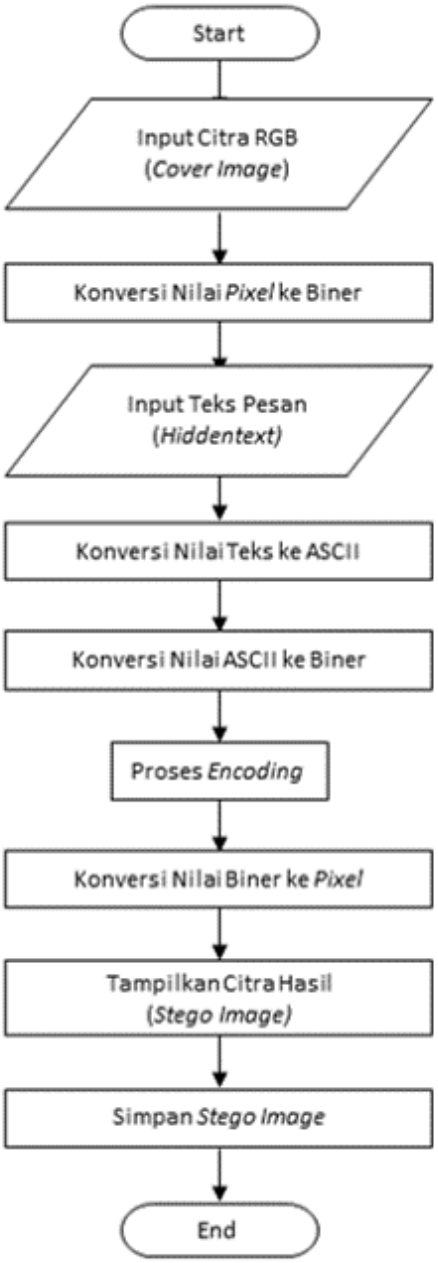
   Penulis mendapatkan informasi yang berkaitan dengan steganografi melalui buku referensi dan juga dalam bentuk *e-book*. Penulis juga mencari informasi melalui berbagai situs di internet yang sesuai dengan topik.

2. Studi Literatur

   Penulis mencoba mencari perbandingan dengan studi sejenis dari beberapa karya ilmiah lokal maupun internasional, seperti jurnal dan skripsi.

3.2 Perancangan Sistem

3.2.1 Proses Penyisipan (*Encoding*) pesan ke Citra *Digital*

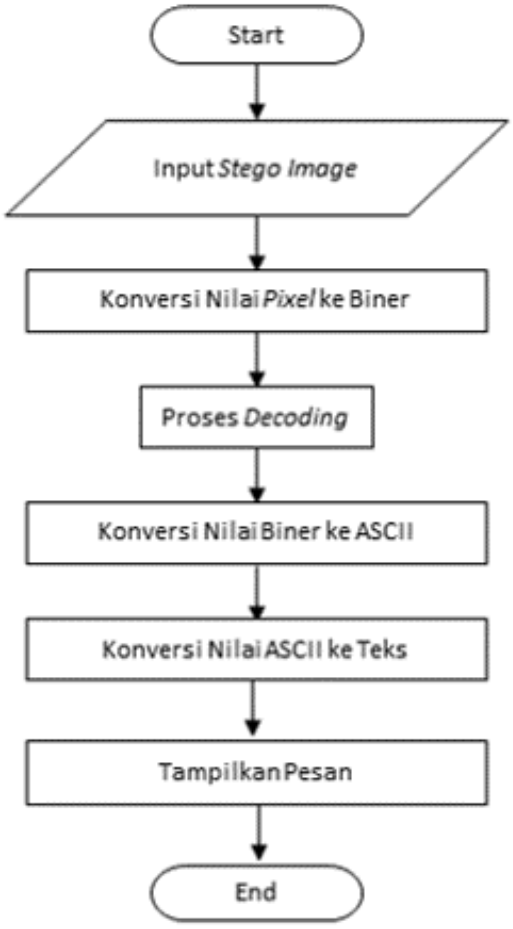


Gambar 3.2: *Flowchart* Penyisipan Pesan Rahasia

Pada gambar di atas adalah *flowchart* proses penyisipan pesan ke dalam *file* citra (*Cover Image*). Dimulai dengan membaca *file* citra RGB. 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 (1 *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 Image*.

Selanjutnya adalah menyisipkan pesan (*Hiddentext*) yang akan disembunyikan ke dalam *Cover Image*. Pesan tersebut dikonversi terlebih dahulu menjadi nilai ASCII dan kemudian dikonversi kembali menjadi nilai Biner. Setelah itu terjadilah proses penyisipan (*Encoding*). Selanjutnya biner yang telah disisipkan akan dikonversikan kembali ke dalam *pixel*. Dan menyimpan citra yang telah disisipkan pesan ke dalam *Cover Image* sehingga diperoleh atau dapat ditampilkan sebuah gambar baru (*Stego Image*).

3.2.2 Proses Ekstraksi (*Decoding*) pesan dari Citra *Digital*

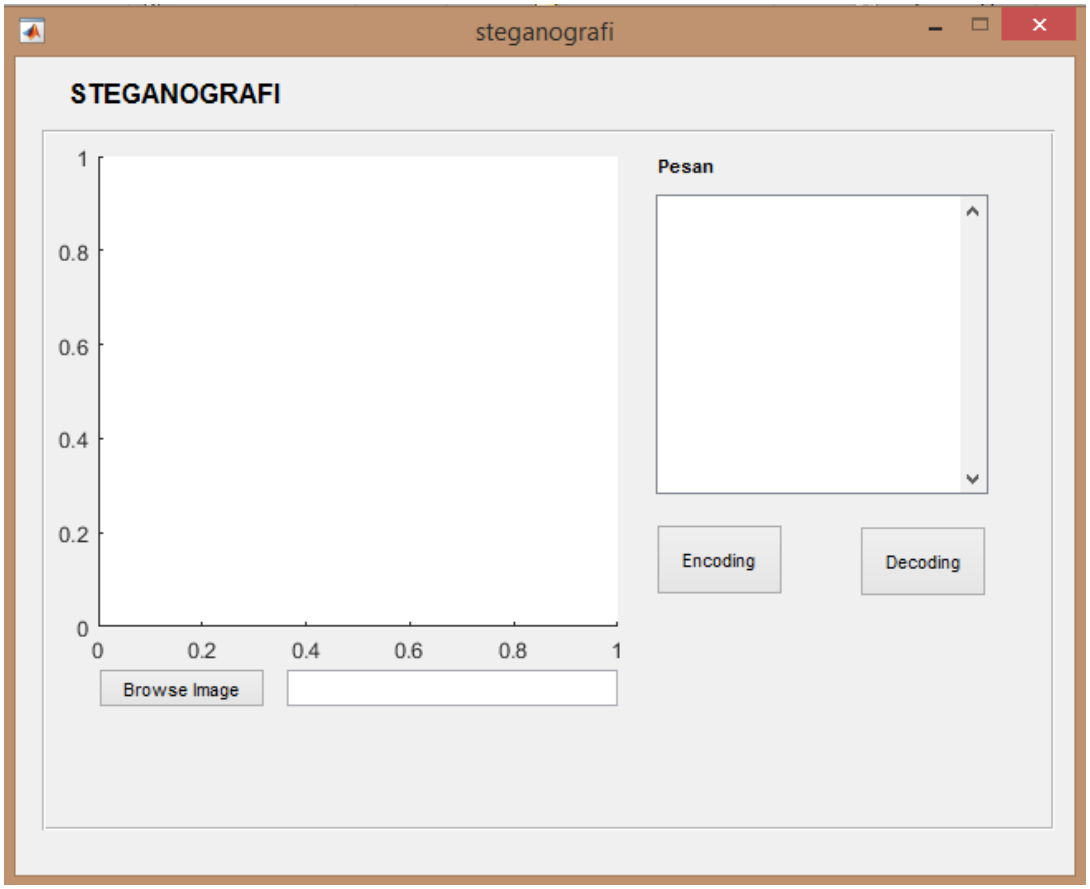


Gambar 3.3: *Flowchart* Ekstraksi Pesan Rahasia

Pada gambar di atas adalah *flowchart* proses ekstraksi pesan dari *Stego Image* yang menghasilkan *Hiddentext* yang terdapat di dalamnya. Prosesnya dimulai dengan membaca *file* citra, dan mengubah *pixel* ke dalam nilai biner. Kemudian proses ekstraksi (*Decoding*). Setelah diperoleh bit-bit yang tersembunyi pada *Cover Image* maka proses berikutnya adalah mengkonversi kembali pesan yang tersembunyi (*Hiddentext*), sehingga pesan dapat ditampilkan kembali.

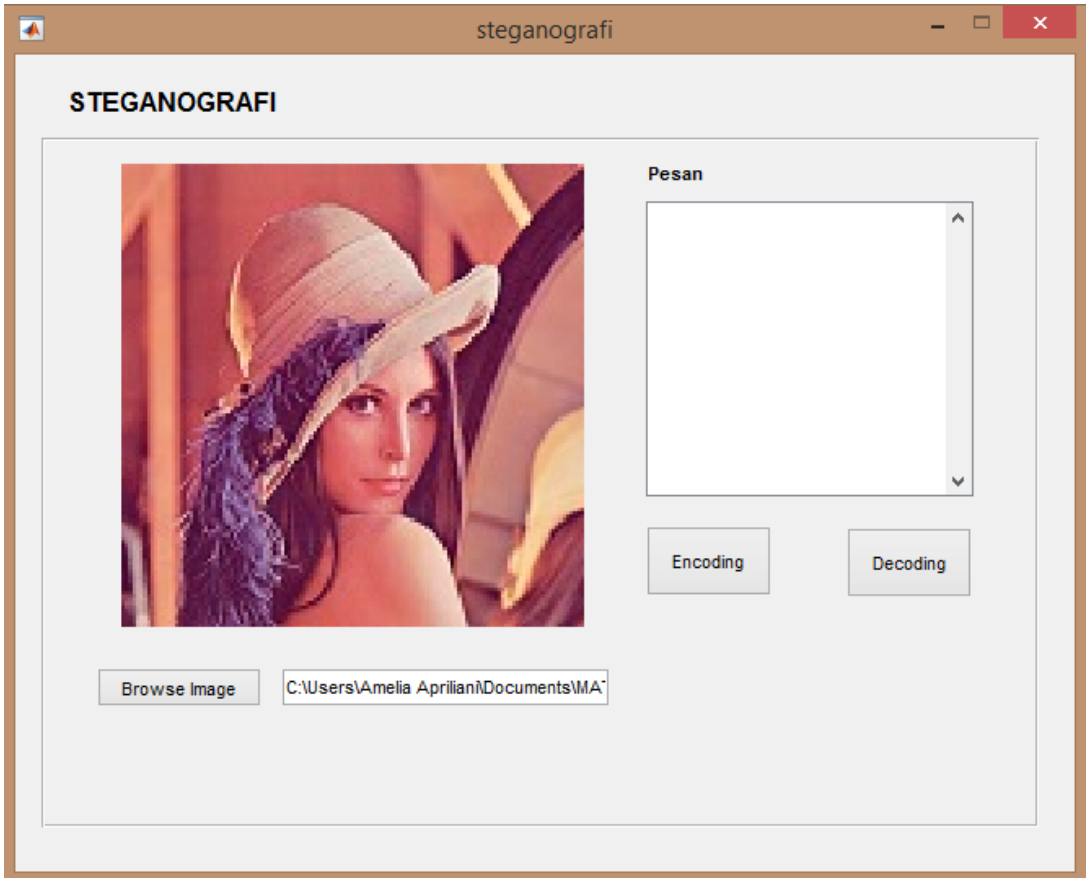
3.2.3 Desain Antar Muka Program

Berikut adalah desain antar muka dari program steganografi yang dibangun dengan menggunakan Matlab.



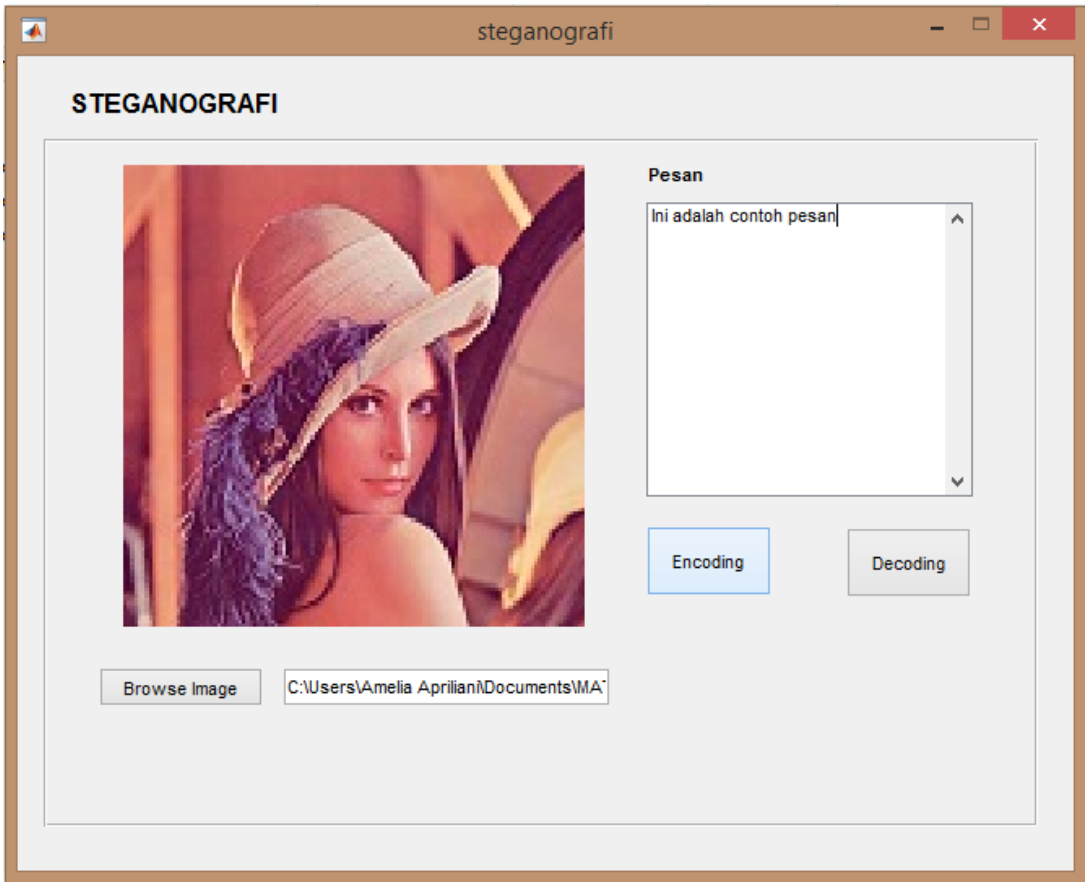
Gambar 3.4: Desain *Form* Steganografi

Dari tampilan tersebut, pengambilan gambar yang akan dijadikan sebagai *Cover Image* dilakukan dengan menekan tombol "*Browse Image*" dan gambar akan tampil.



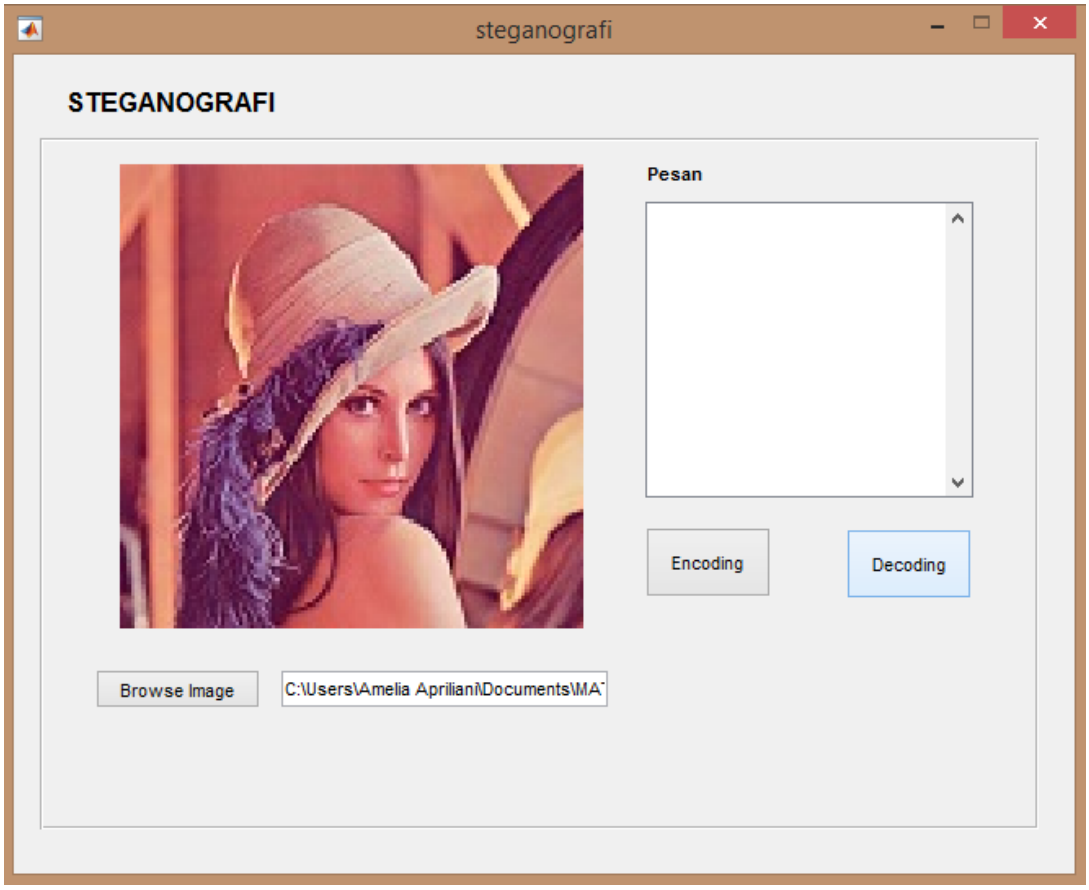
Gambar 3.5: Desain *Form - Cover Image*

Setelah *Cover Image* tampil maka pesan yang akan disisipkan atau *Hidden-text* dapat dituliskan pada kolom pesan. Dan kemudian klik tombol *Encoding* untuk melakukan proses *Encoding*. Setelah proses *Encoding*, maka akan didapatkan *Stego Image* dan disimpan di folder yang sama.

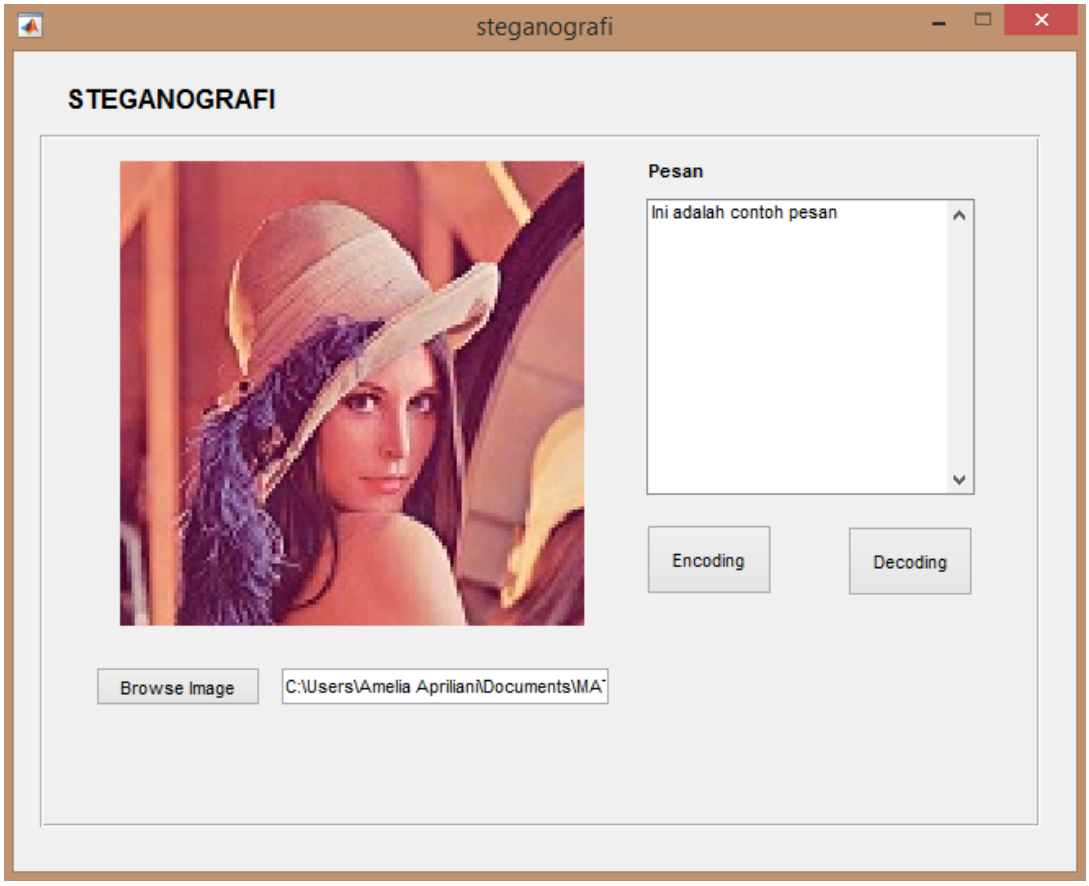


Gambar 3.6: Desain *Form* - Proses *Encoding*

Jika ingin melakukan proses *Decoding*, maka buka *Stego Image* yang telah disimpan. Kemudian klik tombol *Decoding* dan pesan akan didapatkan.



Gambar 3.7: Desain *Form* - Proses *Decoding*



Gambar 3.8: Desain *Form* - Pesan Hasil *Decoding*

DAFTAR PUSTAKA

[1] Adiria. (2010). "Analisis dan Perancangan Aplikasi Steganografi pada Citra Digital dengan Menggunakan Metode LSB (Least Significant Bit)". Skripsi Sarjana pada Universitas Islam Negeri Jakarta

[2] Arymurthy, A. M., dan Setiawan, S. (1992). "Pengantar Pengolahan Citra. Jakarta: PT Elex Media Komputindo".

[3] ASCII Table. 2010. "ASCII Table and Description". ASCII Table [Online]. Tersedia: `https://www.asciitable.com`. [17 April 2018].

[4] Bunyamin, H., dan Adrian. (2009). "Aplikasi Steganography pada File dengan Menggunakan Teknik Low Bit Encoding dan Least Significant Bit". Jurnal Informatika UKM, Vol. 5, No. 2, pp. 107âĂŞ117.

[5] Elgabar, Eltyeb E. A bed. (2013). "Comparison of LSB Steganography in BMP and JPEG Images". International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Vol.3, Issue-5.

[6] Elgabar, Eltyeb E. A bed dan Mohammed, Fakhreldeen A. (2013). "JPEG versus GIF Images in forms of LSB Steganography". International Journal of Computer Science and Network (IJCSN), Vol. 2, Issue 6.

[7] Gautama, Prakriti dan Sharma, Deepak. (2015). "A Survey on Digital Image Steganography Techniques". International Journal of Electronics, Electrical and Computational System (IJEECS), ISSN 2348-117X, Vol. 4, Issue 11.

[8] Hermawati, F. A. (2013). "Pengolahan Citra Digital". Yogyakarta: ANDI.

[9] Irfan. (2013). "Penyembunyian Informasi (steganography) Gambar Menggunakan Metode LSB (Least Significant Bit)". Rekayasa Teknologi Vol. 5, No. 1.

[10] Joshi, K., dan Yadav, R. (2015). "A New LSB-S Image Steganography Method Blend with Cryptography for Secret Communication". Third International Conference on Image Infomation Processing.

[11] Kessler, G. C. (2001). "Steganography Hiding Data Within Data".

[12] M. K., Kadam, K., Koshti, A., dan Dunghav, P. (2012). "Steganography Using Least Signicant Bit Algorithm". International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 3, pp. 338-341.

[13] Munir, R. (2004). "Pengolahan Citra Digital". Bandung: Informatika.

[14] Munir, R. (2006). "Kriptografi". Bandung: Informatika.

[15] Pakereng, M.A Ineke, Beeh, Yos Richard, dan Endrawan, Sonny. (2010). "Perbandingan Steganografi Metode Spread Spectrum dan Least Significant Bit (LSB) Antara Waktu Proses dan Ukuran File Gambar". JURNAL INFORMATIKA, VOL. 6 NO. 1.

[16] M., Pavani, S., Naganjaneyulu, dan C., Nagaraju. (2013). "A Survey on LSB Based Steganography Methods". International Journal Of Engineering And Computer Science, Vol. 2, Issue pp. 2464-2467.

[17] Prasetyo, F. P. (2010). "Steganografi Menggunakan Metode LSB denga Software Matlab". Skripsi Sarjana pada Universitas Islam Negeri Jakarta

[18] Prayudi, Y., dan Kuncoro, P. S. (2005). "Implementasi Steganografi Menggunakan Teknik Adaptive Minimum Error Least Significant Bit Replacement (AMELSBR)". Seminar Nasional Aplikasi Teknologi Informasi.

[19] Rakhmat, B., dan Fairuzabadi, M. (2010). "Steganografi Menggunakan Metode Least Significant Bit dengan Kombinasi Algoritma Kriptografi Vigenere dan RC4". Jurnal Dinamika Informatika, Volume 5, Nomor 2.

[20] Setiana, dan Mahmudy, W. F. (2006). "Steganografi Pada File Citra Bitmap 24 Bit Untuk Pengamanan Data Menggunakan Metode Least Significant Bit (LSB) Insertion". Kursor, vol. 2, no. 2, pp. 38-44.

[21] Wikipedia. (n.d.). Retrieved from https://id.wikipedia.org/wiki/Steganografi