

BAB II TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian yang dilakukan oleh M. Zeghid, M. Machhout, L. Khriji, A. Baganne, dan R. Tourki pada tahun 2007 mengevaluasi enkripsi citra menggunakan algoritma AES yang disimulasikan pada perangkat lunak dan perangkat keras. Hasil yang diperoleh dari xilinx Virtex_II FPGA menunjukkan bahwa rata-rata waktu yang diperlukan oleh algoritma AES untuk menyelesaikan proses enkripsi paling sedikit dibandingkan dengan algoritma enkripsi citra sejenis lainnya. AES membutuhkan waktu sekitar 0,03175 detik untuk mengenkripsi file citra dengan ukuran 200×320 piksel. (Zeghid, 2007)

Di penelitian berikutnya, Manoj. B. dan Manjula. N. Harihar di tahun 2012 menjelaskan bahwa enkripsi dan dekripsi citra menggunakan AES didesain dan diimplementasikan untuk melindungi data citra pribadi dari akses yang tidak diijinkan. Kesuksesan algoritma AES menjadikannya standar enkripsi dan dekripsi yang terbaik yang ada di pasaran. Pada penelitiannya, mereka menggunakan kode VHDL yang disimulasikan menggunakan ISE 12.4 dalam *Xilinx Family Spartan-6(XC6SLX25)*. Frekuensi maksimum yang diperoleh adalah 165.462MHz sedangkan enkripsi dan dekripsi citra mencapai nilai 252.132Mbit/sec. (Manoj, 2012)

Pada buku yang ditulis oleh Mark Stamp menyatakan bahwa metode enkripsi AES menggunakan *block cipher mode*. Dalam bukunya, ia membandingkan 2 jenis mode blok, yaitu ECB(*Electronic Code Book*) dengan CBC(*Cipher Block Chaining*). Hasil penerapan kedua blok tersebut memberikan hasil bahwa CBC memiliki tingkat keamanan yang lebih tinggi dibandingkan dengan EBC. CBC lebih unggul daripada EBC karena citra digital hasil enkripsi AES dengan CBC, pengguna sudah tidak dapat menerka lagi citra apa yang ada di

dalamnya, sedangkan enkripsi citra digital AES menggunakan EBC, pengguna masih dapat menerka sketsa citra digital aslinya. (Stamp, 2006, hlm.51-52)

Penelitian yang dilakukan akan menggunakan algoritma AES dengan mode CBC untuk membantu pembangunan sistem enkripsi *multi-file*. Penggabungan file citra akan menghasilkan ukuran citra yang cukup besar, sehingga diperlukan metode enkripsi yang sudah terbukti aman dan cepat. Enkripsi akan berjalan secara dinamis sehingga waktu yang diperlukan bergantung pada dimensi citra itu sendiri. Untuk *pre-proses* dalam penelitian ini menggunakan *padding* pada saat proses penggabungan citra sebagai pembatas antar citra sekaligus menjadi pengisi kekosongan piksel karena adanya perbedaan ukuran dimensi citra. Hasil penggabungan tersebut kemudian di-enkripsi menggunakan AES dengan mode CBC yang telah dipilih sebelumnya.

2.2 Landasan Teori

2.2.1. Enkripsi Citra Digital

Enkripsi citra digital merupakan sebuah cara pengamanan data dengan cara memanipulasi susunan *bit* pada berkas citra digital sehingga informasi yang sebenarnya tidak dapat diperoleh oleh pihak-pihak yang tidak berkepentingan (Komal, 2011)

70% dari total pertukaran informasi di internet adalah data citra digital. Dapat dikatakan bahwa data ini merupakan bagian yang penting dari pertukaran data pada jaringan internet. Bagaimanapun, data citra digital berbeda dengan data teks. Data citra digital memiliki ukuran file, dan tingkat perulangan antar piksel yang tinggi. Metode enkripsi klasik seperti DES, IDES, yang berjalan baik pada data teks tidak cocok jika diterapkan pada enkripsi data citra digital. Salah satu metode yang cocok diterapkan pada data citra digital adalah AES. (Radhadevi, 2012)

2.2.2. Advanced Encryption Standard

National Institute of Standards and Technology(NIST) mengumumkan algoritma *Rijndael* sebagai *Advanced Encryption Standard*(AES) pada Oktober 2000(Manoj, 2012). Algoritma *Rijndael* merupakan algoritma dengan tingkat keamanan, performa, efisiensi, kemampuan dalam penerapan dan fleksibilitas yang paling baik . Algoritma ini ditemukan oleh Joan Daemen dari Proton World International dan Vincent Rijmen dari Universitas Katholieke di Leuven. (Karthigaikumar, 2011)

Metode AES adalah salah satu metode *Block Cipher* yang membagi data asli kedalam ukuran blok yang ditentukan dan menghasilkan blok *cipher* sesuai dengan ukuran blok yang sudah ditentukan. *Ciphertext* didapatkan dari *plaintext*/data asli dengan pengulangan fungsi F sampai beberapa kali putaran. Fungsi F bergantung pada hasil dari putaran sebelumnya dan kunci K(round function) diterapkan pada setiap putaran. (Stamp, 2006)

Algoritma AES :

Algoritma AES hanya mengijinkan panjang data 128 *byte* yang dapat dibagi menjadi empat operasi blok. Blok ini dioperasikan pada *array byte* dengan pengaturan matriks 4x4 yang disebut sebagai *state*. Untuk enkripsi secara keseluruhan, data harus melewati sebanyak Nr putaran(Nr=10, 12, 14). Putaran ini ditentukan oleh transformasi berikut (Manoj, 2012):

1.Transformasi SubBytes

Transformasi *SubBytes* memetakan setiap *byte* dari *array state* dengan menggunakan tabel substitusi S-box.(Munir, 2007)

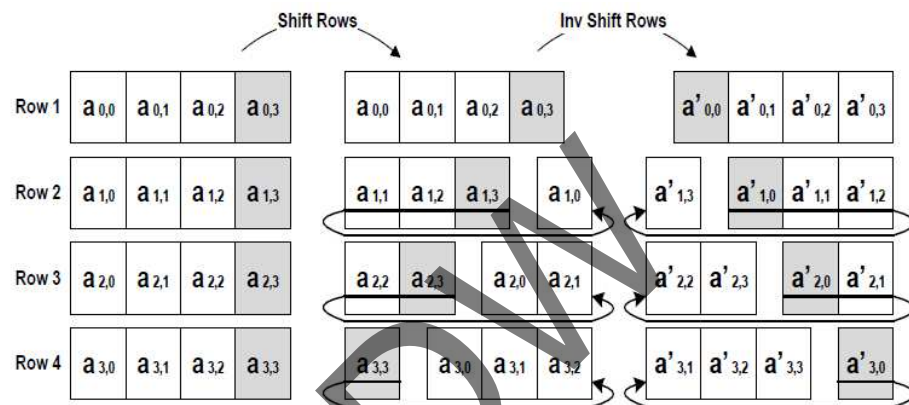
ByteSub(3c) = eb, dikarenakan eb muncul dalam tabel S-box pada baris 3 dan kolom c *Tabel 2.1*. (Stamp, 2006)

Tabel 2.1. Tabel AES Byte-Sub
(Stamp, 2006)

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

2. Transformasi ShiftRow

Transformasi ini melakukan pergeseran secara *wrapping* pada 3 baris terakhir dari *array state*. Jumlah pergeseran bergantung pada nilai baris(*r*). Baris *r*=1 digeser 1 *byte*, baris *r*=2 digeser 2 *byte*, dan baris *r*=3 digeser 3 *byte*. Baris *r*=0 tidak mengalami pergeseran. (Munir, 2007) Contoh *ShiftRow* dapat dilihat pada *Gambar 2.1*.



Gambar 2.1. Operasi Shift-Row
(Manoj, 2012)

3. Transformasi MixColumn

Transformasi *MixColumn* mengalikan setiap kolom dari *array state* dengan *polinomial* $a(x)$ yang ditetapkan dengan rumus :

$$a(x) = 03 * x^3 + 01 * x^2 + 01 * x + 02 \quad [1]$$

Transformasi ini dinyatakan sebagai perkalian matriks seperti pada *Gambar 2.2*. Perkalian matriks tersebut dijelaskan dengan *Gambar 2.3*.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Gambar 2.2. Perkalian Matriks.

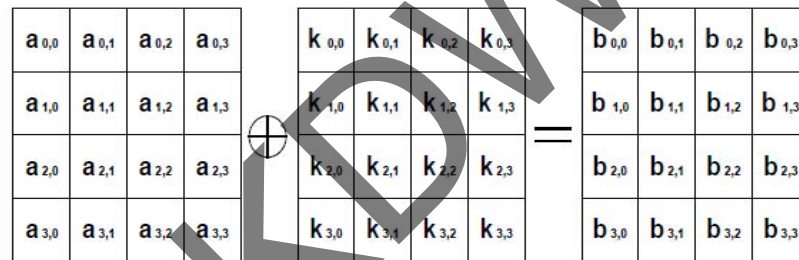
(Munir, 2007)

$$\begin{aligned}
S'(0,j) &= (2.s(0,j)) \text{ xor } (3.s(1,j)) \text{ xor } (s(2,j)) \text{ xor } (s(3,j)) \\
S'(1,j) &= (s(0,j)) \text{ xor } (2.s(1,j)) \text{ xor } (3.s(2,j)) \text{ xor } (s(3,j)) \\
S'(2,j) &= (s(0,j)) \text{ xor } (s(1,j)) \text{ xor } (2.s(2,j)) \text{ xor } (3.s(3,j)) \\
S'(3,j) &= (3.s(0,j)) \text{ xor } (s(1,j)) \text{ xor } (s(2,j)) \text{ xor } (2.s(3,j))
\end{aligned}$$

Gambar 2.3. The Mix Column Transformation on a Single Column $j(0 \leq j \leq 3)$
(Karthigaikumar, 2011)

4. Transformasi AddRoundKey

Add RoundKey adalah transformasi yang melakukan operasi XOR terhadap sebuah round key dengan array state, dan hasilnya disimpan di array state selanjutnya(Munir, 2007), seperti pada Gambar 2.4.



Gambar 2.4. Operasi Add Round Key
(Manoj, 2012)

Penggunaan Block Cipher cukup mudah, selama hanya ada satu blok untuk dienkripsi. Namun penggunaan block cipher akan mengalami kendala saat memproses jumlah blok yang banyak. Dengan adanya permasalahan ini, muncul beberapa model mode dalam pengerjaannya, beberapa diantaranya adalah ECB(Electronic Codebook) dan CBC(Cipher Block Chaining). (Stamp, 2006)

2.2.3. Cipher Block Chaining Mode

Cipher Block Chaining adalah salah satu mode dalam model block cipher, mode ini memiliki hasil akhir enkripsi yang sangat aman, informasi yang ada benar-benar hancur tidak beraturan. Sedangkan hasil enkripsi dari

Electronic Codebook informasi sebenarnya masih dapat ditebak. (Stamp, 2006)

Rumus Enkripsi *EBC*(*Electronic Codebook*)

$$C_i = E(P_i, K) \quad \text{for } i = 0, 1, 2, \dots \quad [2]$$

dan *CBC*(*Cipher Block Chaining*) dengan rumus enkripsi

$$C_i = E(P_i \oplus C_{i-1}, K) \quad \text{for } i = 0, 1, 2, \dots \quad [3]$$

Gambar 2.5.a adalah hasil Enkripsi dari EBC mode, sedangkan *Gambar 2.5.b* merupakan hasil Enkripsi dari CBC mode.



Gambar 2.5.a. Alice dan Mode EBC
(Stamp, 2006)

Gambar 2.5.b. Alice dan Mode CBC
(Stamp, 2006)

Dalam penelitian ini, penulis memutuskan untuk menggunakan mode *CBC* sebagai mode dalam algoritma AES.

2.2.4. Padding

Algoritma Blok Cipher seperti DES, mode Electronic Code Book(EBC), dan mode CBC memerlukan inputan dengan jumlah kelipatan dari ukuran blok nya. Jika citra asli yang akan dienkripsi tidak berjumlah kelipatan dari ukuran blok nya, maka diperlukan *padding* sebelum melakukan enkripsi, dengan cara menambahkan sebuah *padding string*. Ketika didekripsi, perlu menghilangkan *padding* tersebut agar kembali pada informasi asli nya. (Dworkin, 2001)

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Kebutuhan Perangkat Keras dan Perangkat Lunak

Sistem akan dibangun berbasis *desktop* dengan menggunakan .NET. Kebutuhan perangkat keras dan perangkat lunak yang dibutuhkan penulis untuk membangun sistem adalah:

- Kebutuhan perangkat keras, yakni:
 - *Processor* Intel® Core™ 2 Duo-T5750 CPU @ 2.00GHz
 - *RAM* 3072 MB
 - *Harddisk* 160GB
 - *Monitor* 14 inch
 - *Mouse*
 - *Keyboard*
- Kebutuhan perangkat lunak, yakni:
 - *Sistem operasi* : Windows 7 Professional 32-bit
 - *Text editor*: Visual Studio 2010
 - *Database*: Oracle XE 10g
 - *Framework*: .NET Framework 4.0

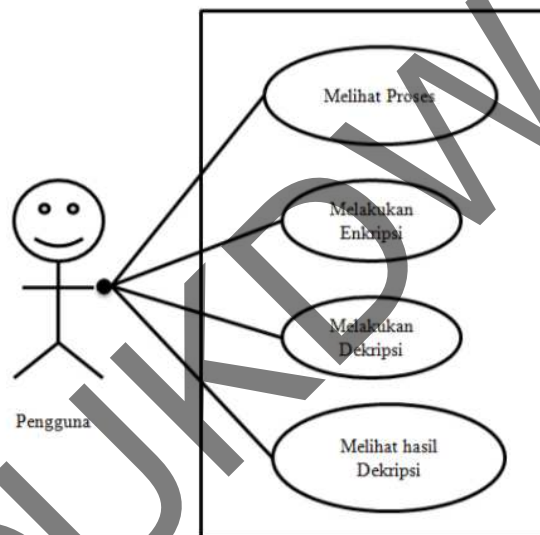
Kebutuhan perangkat keras dan perangkat lunak minimum yang diperlukan menggunakan untuk mengoperasikan sistem adalah:

- Kebutuhan perangkat keras, yakni:
 - *Processor* 1 GHz
 - *RAM* 512 MB
 - *Harddisk* 2 GB
 - *Monitor* 12 inch
 - *Mouse*
 - *Keyboard*

- Kebutuhan perangkat lunak, yakni:
 - Sistem operasi: Windows XP Professional SP1

3.2 Diagram Use Case

Sistem yang akan dibangun melibatkan dua aktor di dalamnya, yaitu pengguna dan sistem itu sendiri. Pengguna adalah aktor yang menggunakan sistem untuk meng-enkrip dan men-dekrip sedangkan sistem adalah program yang digunakan oleh aktor utama.



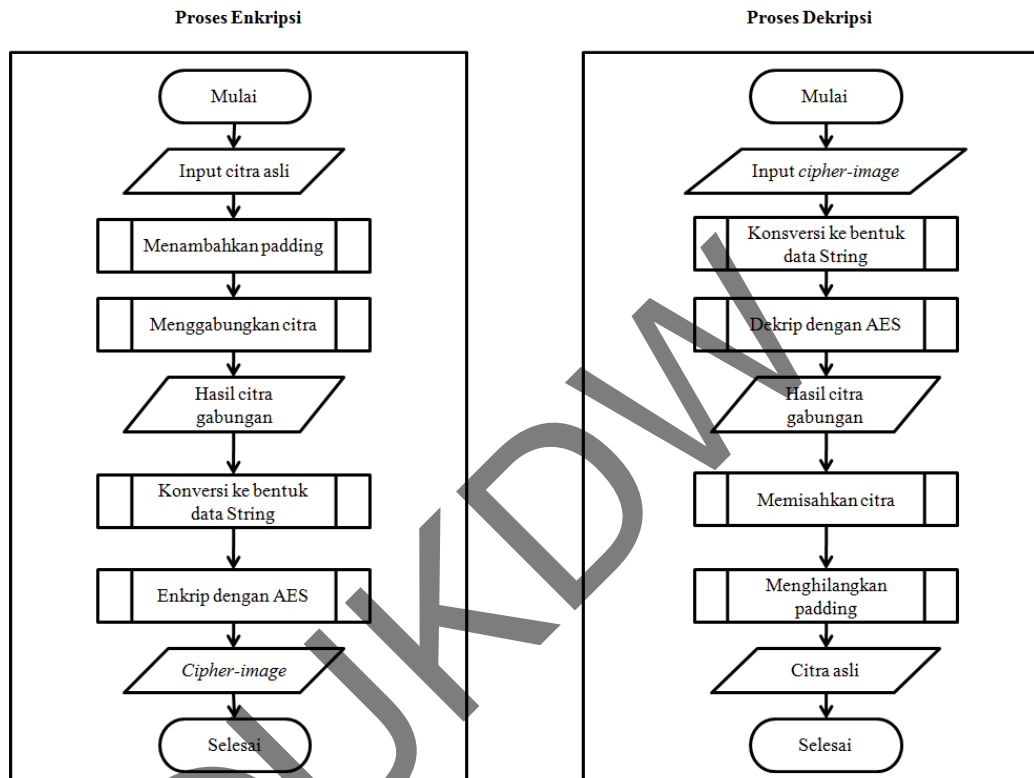
Gambar 3.1. Diagram Use Case

Fungsionalitas yang dapat diakses oleh aktor adalah:

- Melihat Proses adalah proses dimana pengguna dapat melihat proses konversi dari tipe data bitmap ke bentuk *string* yang kemudian diubah ke bentuk *cipher-text* ataupun sebaliknya.
- Melakukan enkripsi adalah proses mengubah citra asli menjadi *cipher-image*.
- Melakukan dekripsi adalah proses mengembalikan dari *cipher-image* menjadi citra asli .
- Melihat hasil dekripsi adalah proses di mana pengguna dapat melihat hasil dekripsi setiap citra.

3.3 Diagram Alir Sistem

Secara garis besar, sistem yang akan dibangun memiliki dua alir kerja, yaitu pada alir kerja Enkripsi dan alir kerja dekripsi seperti diagram alir pada Gambar 3.3.



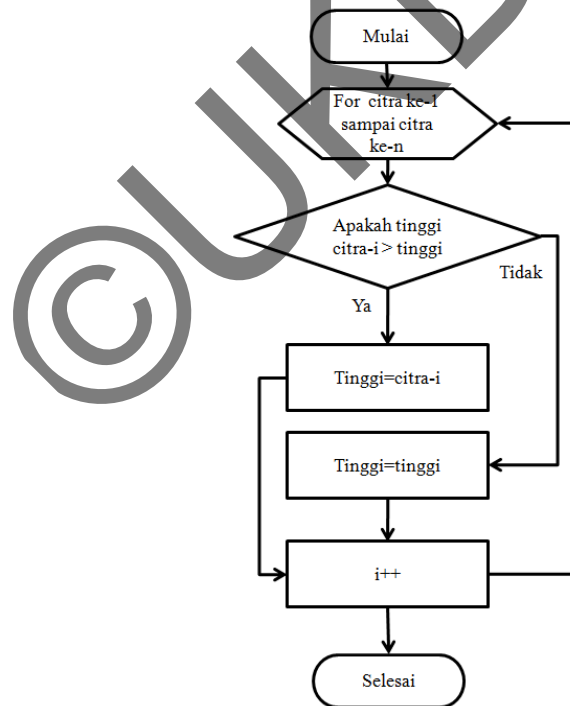
Gambar 3.2. Diagram Alir Sistem

Berdasarkan diagram alir di atas, sistem yang akan dibangun memiliki 2 alur kerja, yaitu alur kerja proses enkripsi dan alur kerja proses dekripsi. Setiap alur kerja memiliki 4 modul baik pada proses dekripsi maupun enkripsi yang perlu dikerjakan untuk mendapatkan hasil keluaran berupa citra hasil enkrip dan citra hasil dekrip, yaitu:

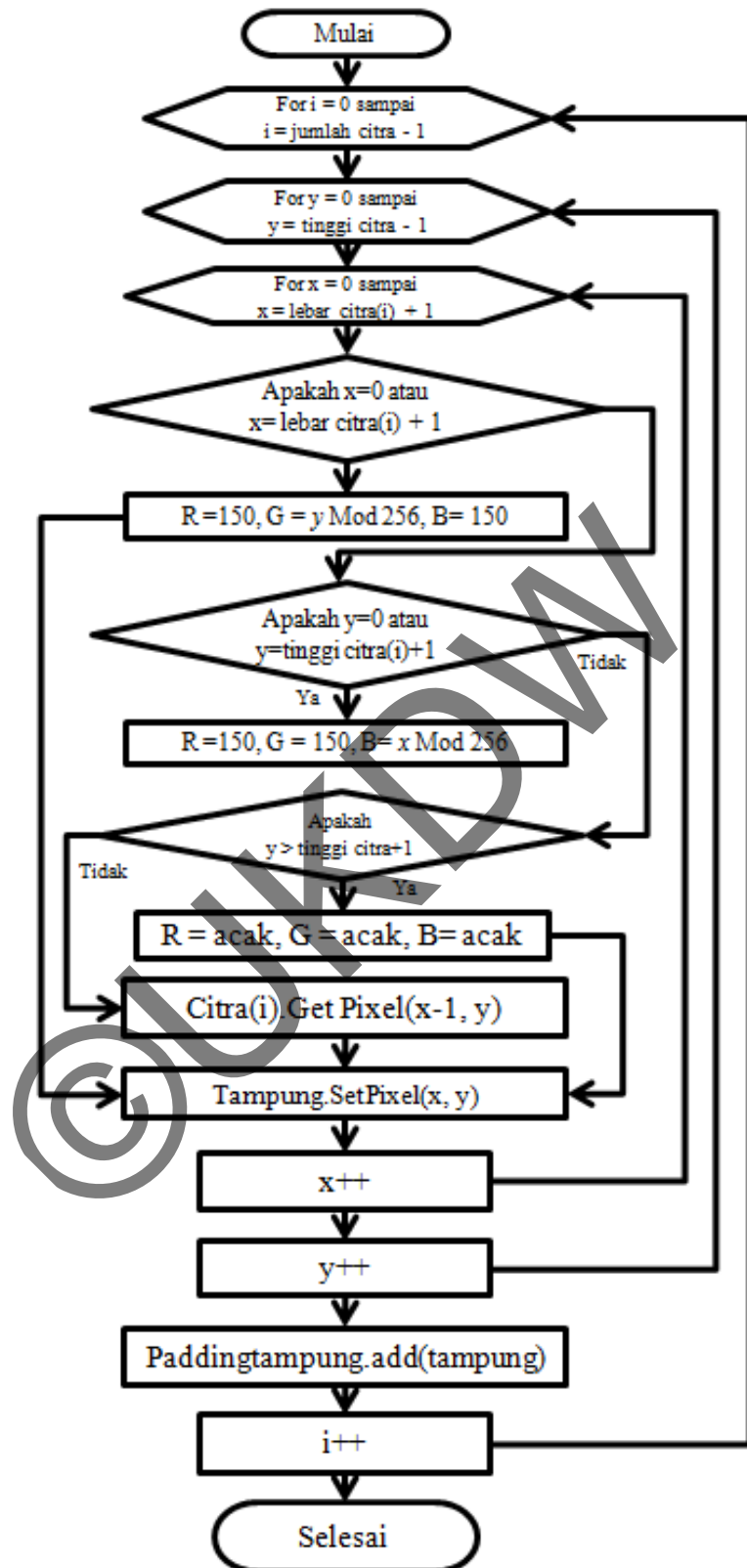
3.3.1 Menambahkan Padding Citra

Dalam proses ini diperlukan suatu variabel yang menampung nilai y yang paling besar. Variabel tersebut akan digunakan sebagai pembanding terhadap setiap nilai y citra. Diagram alirnya dapat dilihat pada *Gambar 3.3*, kemudian jika suatu citra nilai y nya kurang dari variabel tampungan tersebut, maka selisih nya akan diberi *padding* dengan warna acak yang tidak berpola, atau disebut juga sebagai *random-noise*, agar seluruh citra memiliki nilai y yang sama.

Setiap citra akan diberi penanda berupa garis vertikal berwarna degradasi warna ungu dengan rumus ($R = 150, G = y \text{ Mod } 256, B = 150$) dengan ketebalan 1 piksel pada sisi kiri dan kanan citra, sedangkan garis penanda horizontal berwarna degradasi warna kuning rumus ($R = 150, G = 150, B = x \text{ Mod } 256$) dengan ketebalan 1 piksel pada sisi atas dan bawah citra. Hasil dari menambahkan padding pada citra akan ditampung kedalam *list paddingtampung*. *Gambar 3.4* merupakan diagram alir untuk proses menambahkan padding Citra.



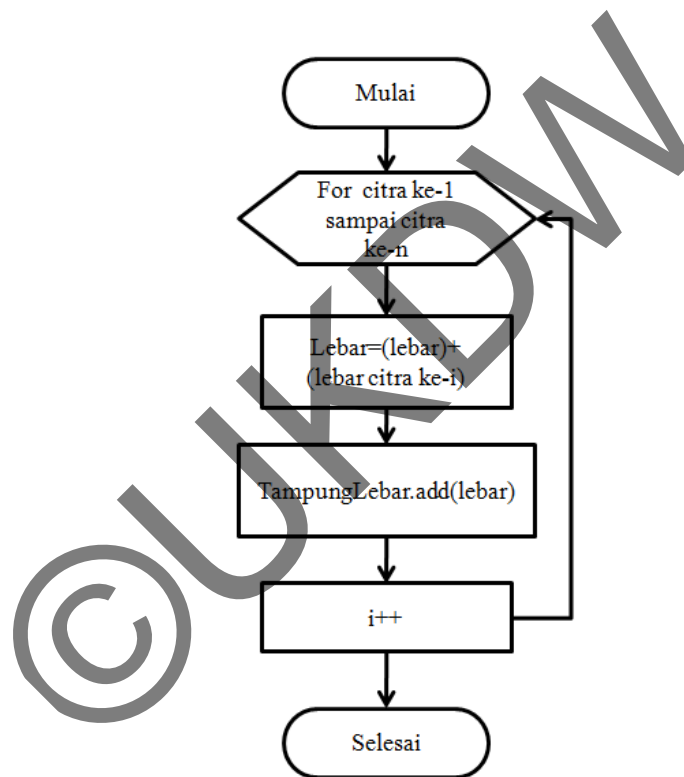
Gambar 3.3. Diagram Alir Menentukan Tinggi Citra Gabungan



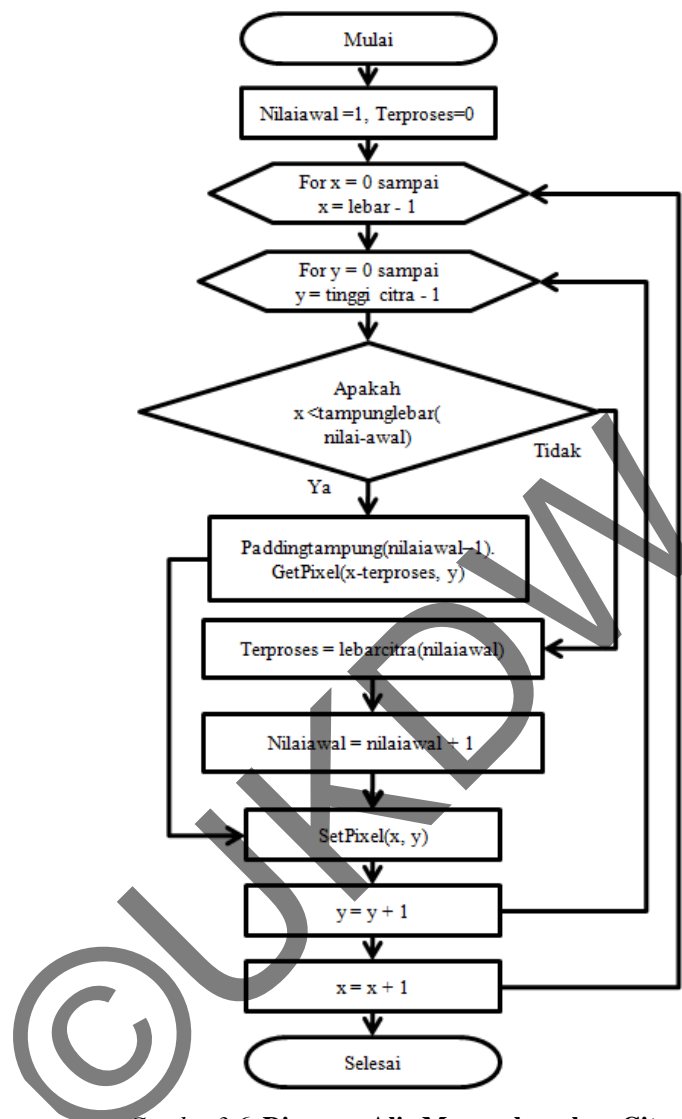
Gambar 3.4. Diagram Alir Menambahkan Padding Pada Citra

3.3.2 Menggabungkan Citra

Dalam proses penggabungan citra dibutuhkan sebuah pre-proses, yaitu menghitung lebar keseluruhan citra untuk kebutuhan dalam mendeklarasikan kanvas. Jumlah lebar juga ditampung dalam *list*. Diagram alir untuk menghitung lebar keseluruhan citra dapat dilihat pada *Gambar 3.5*. *List* jumlah lebar digunakan pada proses pengambilan nilai piksel dari citra hasil *padding*/variabel *padding*tampung. Diagram alir menggabungkan citra dapat dilihat pada *Gambar 3.6*.



Gambar 3.5. Diagram Alir Menghitung Lebar Citra Gabungan

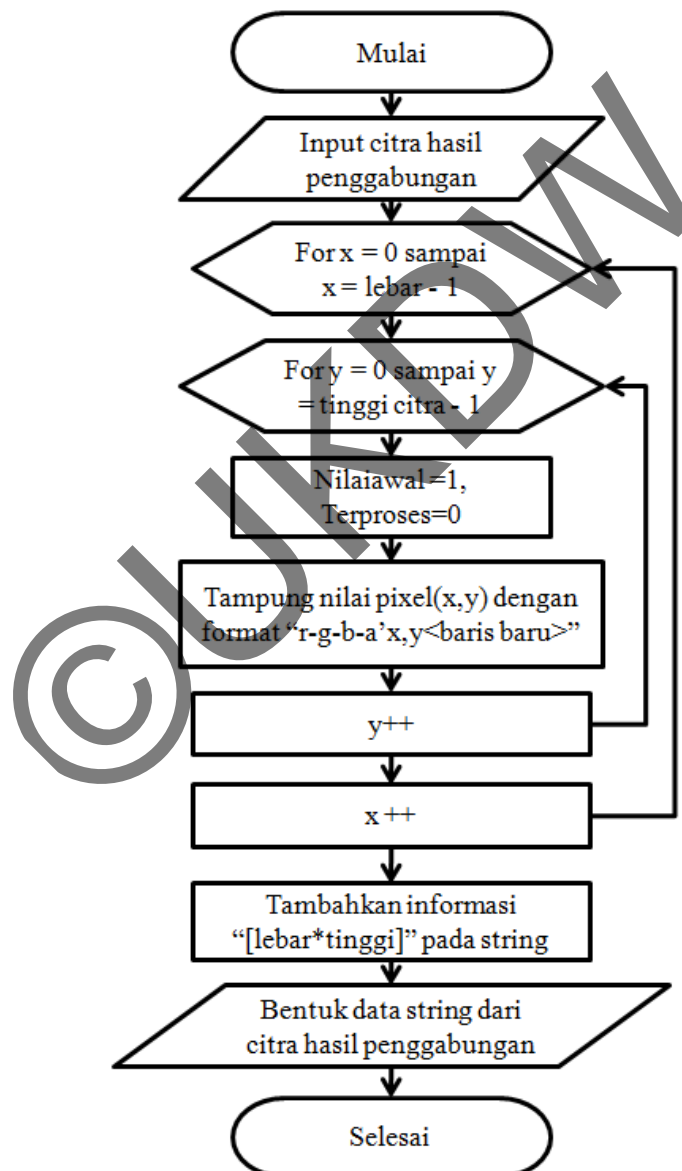


Gambar3.6. Diagram Alir Menggabungkan Citra

Setelah proses penambahan *padding* dan penggabungan citra selesai dijalankan, citra yang dimasukkan oleh pengguna akan menjadi satu bentuk citra yang sudah digabungkan. Hasil citra penggabungan ini akan menjadi suatu input dalam proses selanjutnya, yaitu proses konversi data ke bentuk data string agar kemudian dapat dilakukan proses enkripsi dengan metode AES yang menggunakan *library* dari Visual Studio 2010 dalam bahasa pemrograman VB.NET.

3.3.3 Konversi Tipe Data Bitmap ke Tipe Data String

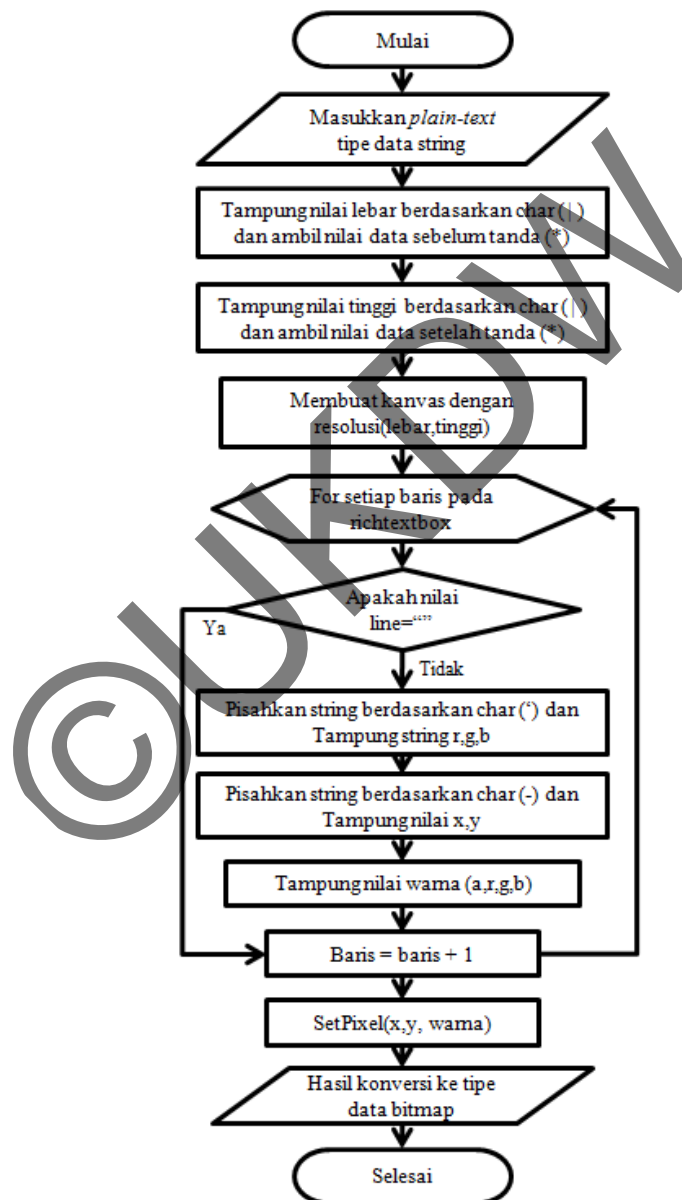
Proses konversi data menjadi tipe data string harus dilakukan dengan alasan library *kriptografi* yang disediakan oleh Visual Studio 2010 dalam bahasa pemrograman VB.NET memiliki suatu kondisi pada tipe data sebagai inputannya, yaitu tipe data *string*. Diagram alir proses konversi data ke tipe data string dapat dilihat pada *Gambar 3.7*.



Gambar 3.7. Diagram Alir Konversi Tipe Data Bitmap ke Tipe Data String

3.3.4 Konversi Tipe Data String ke Tipe Data Bitmap

Proses konversi data *string* menjadi tipe data *bitmap* harus dilakukan sebelum proses dekripsi, karena *library* kriptografi yang disediakan oleh Visual Studio 2010 dalam bahasa pemrograman VB.NET memiliki hasil keluaran dengan tipe data *string*. Diagram alir proses konversi data ke tipe data string dapat dilihat pada *Gambar 3.8*.



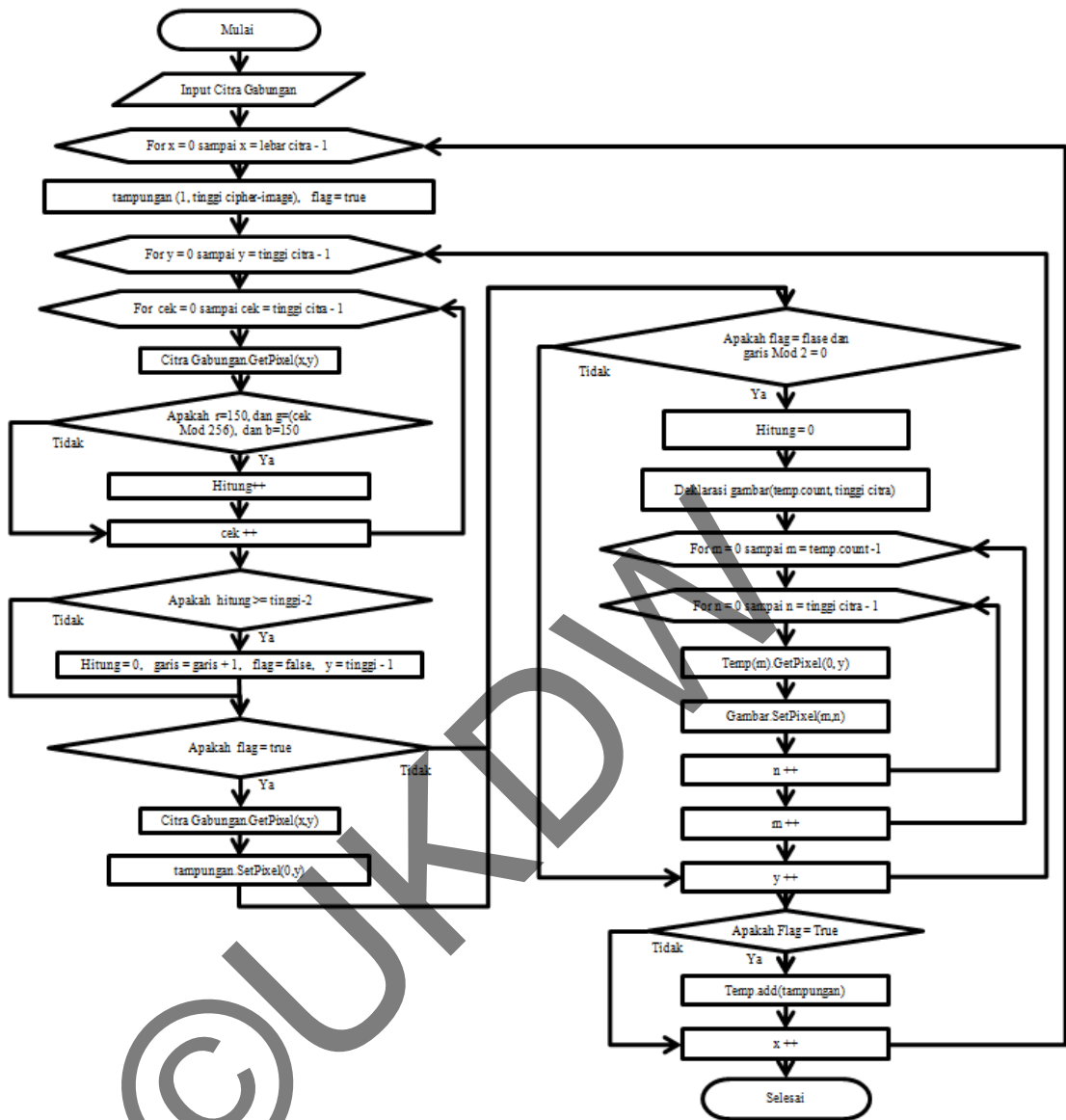
Gambar 3.8. Diagram Alir Konversi Tipe Data String ke Tipe Data Bitmap

3.3.5 Memisahkan Citra

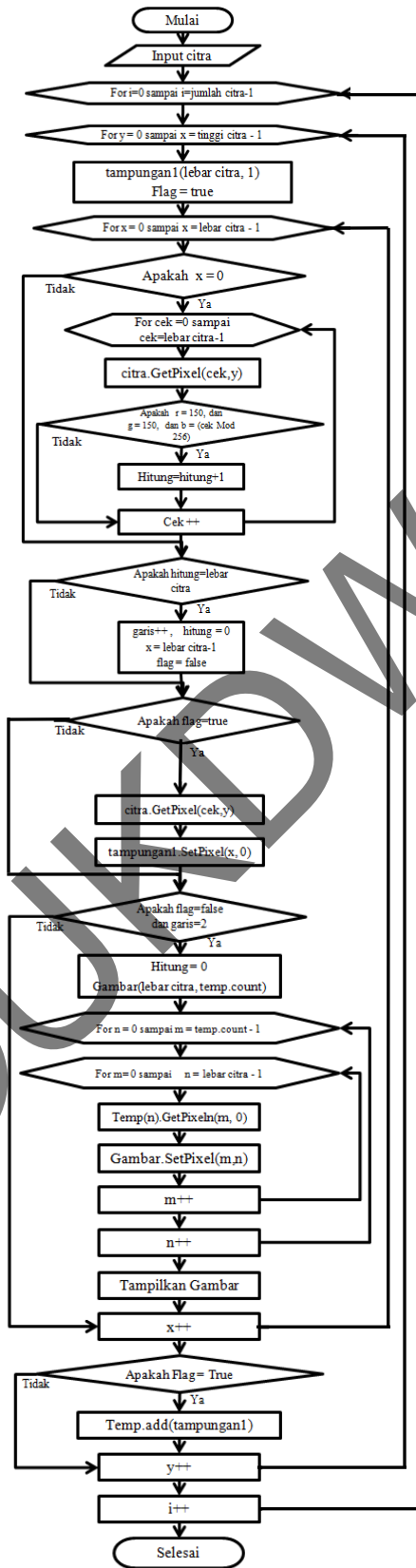
Proses ini digunakan untuk memisahkan citra hasil dekripsi agar jumlah citra kembali menjadi sebanyak jumlah citra yang dimasukkan oleh pengguna pada saat mengenkrip. Dalam proses pemisahan citra, sistem akan mendeteksi jumlah citra berdasarkan garis pembatas secara garis vertikal berwarna degradasi warna ungu dengan rumus ($R = 150$, $G = y \text{ Mod } 256$, $B = 150$) dengan ketebalan 1 piksel pada sisi kiri dan kanan citra. Jika sistem menemukan 4 garis pembatas, maka citra gabungan itu terdiri dari 2 citra, karena setiap citra memiliki garis pembatas berwarna pada sisi kanan dan kiri nya. Diagram alir memisahkan citra dapat dilihat pada *Gambar.3.9*.

3.3.6 Menghilangkan *Padding*

Proses ini merupakan proses akhir dari alir sistem dekripsi citra multi-file. Pada proses ini, sistem akan mendeteksi apakah terdapat *padding* pada citra yang diproses, jika terdapat *padding* pada citra tersebut, maka sistem akan menghilangkannya dan citra akan kembali ke bentuk aslinya seperti sebelum dienkrip. Diagram alir menghilangkan *padding* dapat dilihat pada *Gambar 3.10*.



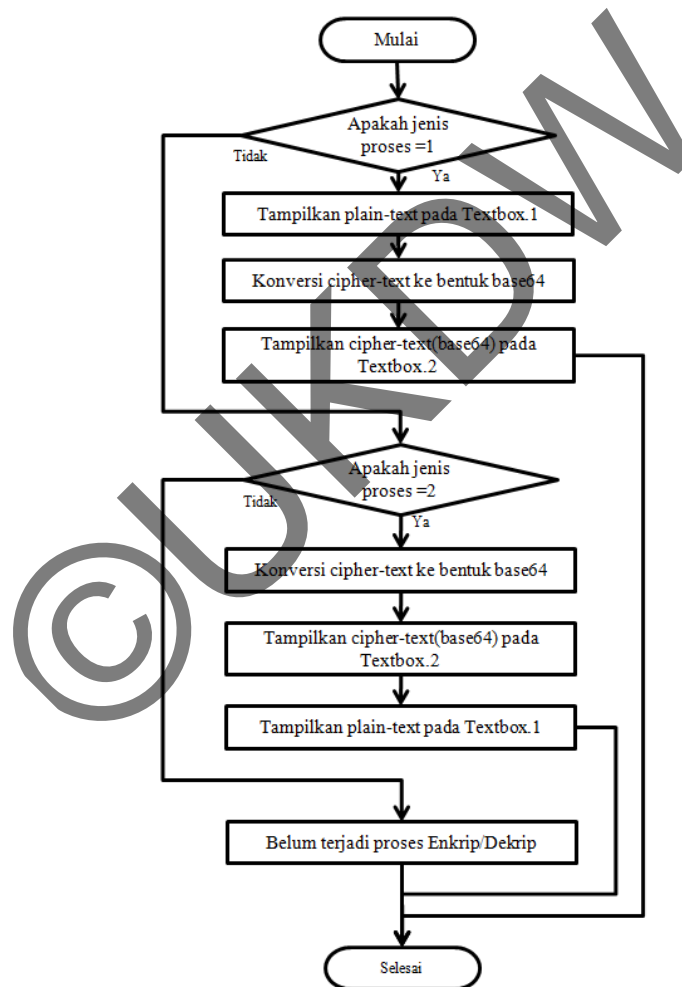
Gambar 3.9. Diagram Alir Memisahkan Citra



Gambar 3.10. Diagram Alir Menghilangkan Padding

3.3.7 Melihat Proses

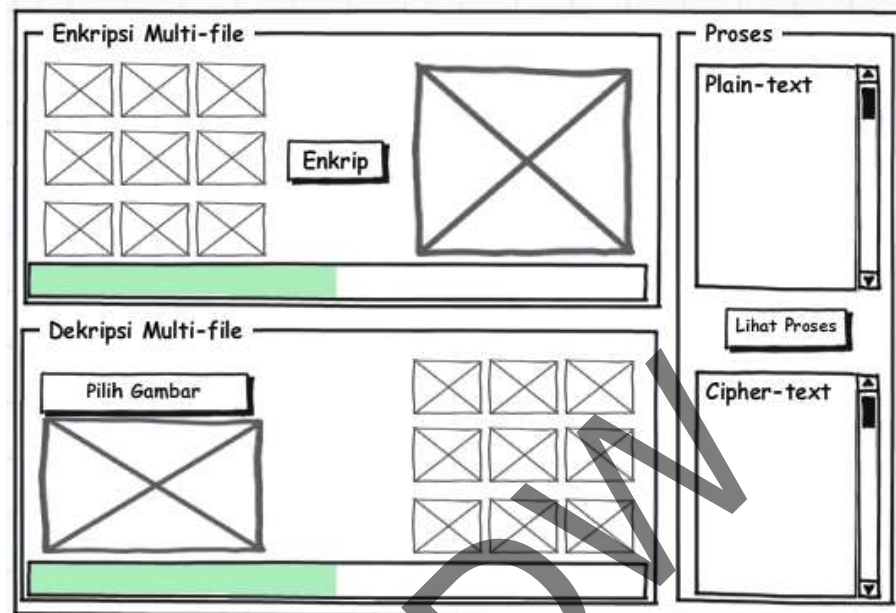
Melihat proses adalah proses yang terpisah dari alur sistem utama, proses ini bersifat pilihan. Jika proses ini tidak dijalankan maka alur sistem utama tidak akan terganggu sama hal nya jika proses ini dijalankan. Modul ini berfungsi untuk melihat proses baik enkripsi maupun dekripsi pada data *string*, data *string* ini merupakan hasil konversi dari citra penggabungan. Diagram alur proses melihat proses dapat dilihat pada *Gambar 3.11*.



Gambar 3.11. Diagram Alir Melihat Proses

3.4 Rancangan Antarmuka Sistem

3.4.1 Rancangan Antarmuka *Form* Main Menu

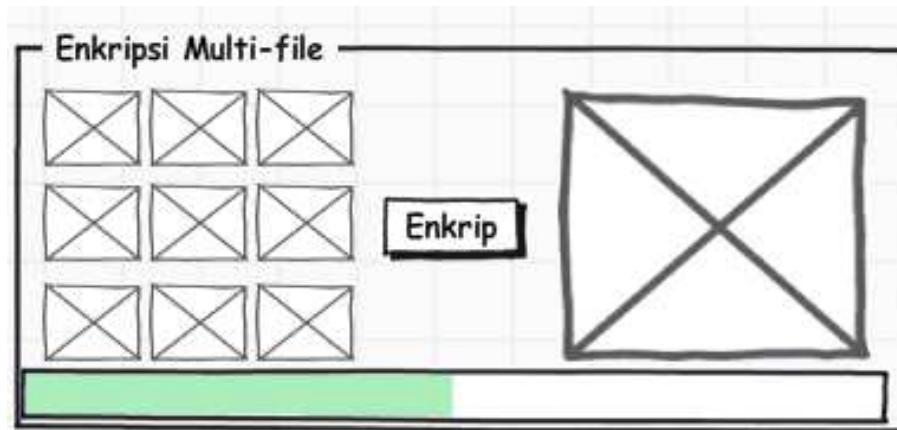


Gambar 3.12. Rancangan Antarmuka *Form* Main Menu

Pada *form* Main Menu, Gambar 3.12, terdapat 3 bagian, yaitu bagian enkrip *multi-file*, bagian dekrip *multi-file*, dan bagian proses. Untuk enkripsi dilakukan pada bagian pertama, bagian kedua untuk melakukan dekripsi, dan bagian ketiga untuk melihat proses perubahan *plain-text* menjadi *cipher-text* atau sebaliknya.

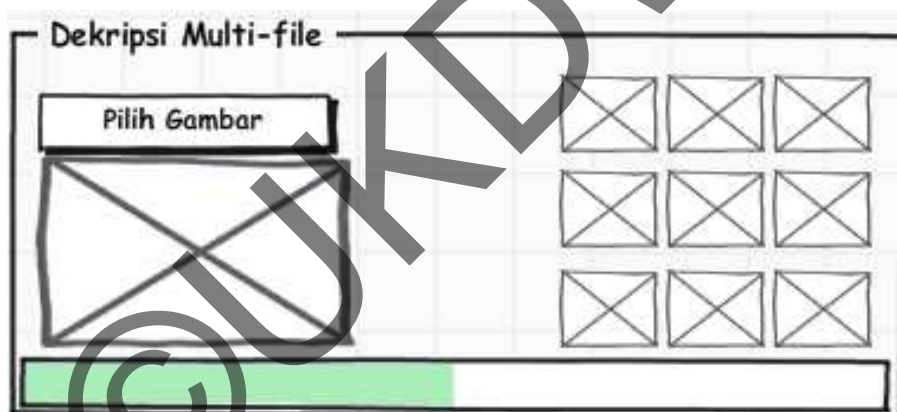
3.4.1.1 Rancangan Antarmuka Bagian Enkripsi

Di dalam *form* Main Menu-Enkripsi, Gambar 3.13, pengguna dapat memasukkan citra pada 9 *picture-box* yang berukuran kecil kemudian menekan tombol “Enkrip” untuk menjalankan proses enkripsi. Hasil dari proses ini akan ditampilkan pada *picture-box* yang berukuran besar.



Gambar 3.13. Rancangan Antarmuka Form Main Menu Bagian Enkripsi

3.4.1.2 Rancangan Antarmukra Bagian Dekripsi

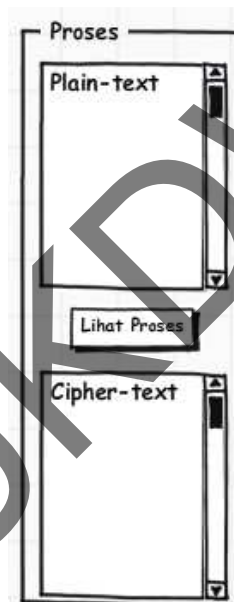


Gambar 3.14. Rancangan Antarmuka Form Main Menu bagian dekripsi

Di dalam form Main Menu-Dekripsi, Gambar 3.14, pengguna dapat memasukkan citra dengan menekan tombol “Pilih gambar”, setelah pengguna memasukkan citra, sistem secara otomatis akan melakukan proses dekripsi dan dilanjutkan dengan proses memisahkan citra dan menghilangkan *padding*-nya. Hasil dari proses dekripsi akan ditampilkan pada *picture-box* yang berukuran besar, sedangkan hasil dari memisahkan dan menghilangkan *padding* citra akan ditampung pada *picture-box* yang berukuran kecil sebanyak maksimal 9 citra.

3.4.1.3 Rancangan Antarmukra Bagian Melihat Proses

Pada *form* Main Menu-Proses, *Gambar 3.15*, pengguna dapat melihat proses sebelum dan sesudah dienkrip ataupun didekrip. Pengguna dapat melihat proses tersebut dengan menekan tombol “Lihat Proses”, secara otomatis, sistem akan menampilkan hasil tersebut pada *text-box plain-text* dan *text-box cipher-text*. Untuk melihat hasil dari proses tersebut, pengguna harus terlebih dahulu melakukan proses enkripsi atau proses dekripsi. Jika pengguna secara tidak sengaja menekan tombol tersebut, maka sistem akan memberitahukan kepada pengguna bahwa pengguna belum melakukan proses apapun.



Gambar 3.15. Rancangan Antarmuka Form Main Menu Bagian Melihat Proses

3.5 Contoh Perhitungan Manual Sistem

3.5.1 Perhitungan Dalam Proses Enkripsi

3.5.1.1 Perhitungan Nilai Terbesar dari Tinggi Citra

Sebagai contoh, pengguna x memasukkan 3 file citra dengan dimensi masing-masing citra $A(30,24)$, $B(40,40)$ dan $C(37,55)$. Sistem akan menampilkan nilai y yang terbesar diantara seluruh citra yang diinputkan dengan cara

membandingkan tampungan dengan nilai y dari citra yang sedang diproses dengan rumus :

$$\text{Tampung} = \max_{1 < i \leq 3} y \quad [4]$$

Dengan rumus diatas, maka nilai tampungan y dari citra A(30,24), B(40,40) dan C(37,55) adalah 55.

3.5.1.2 Perhitungan Dimensi Citra Setelah Diberi *Padding*

Pemberian *padding* dilakukan agar seluruh citra memiliki dimensi nilai y yang sama agar dapat digabungkan secara *horizontal* menjadi sebuah citra yang baru. Citra diberikan garis tepi di setiap sisinya sebesar 1 piksel sebagai garis pembatas yang menjadi informasi bagi sistem bahwa di dalam garis tepi tersebut merupakan citra asli. Dengan menambahkan garis pembatas sebesar 1 piksel dan mengetahui nilai terbesar dari citra yang ditampung dalam variabel *tampung*, maka nilai dimensi x menjadi $x+2$ dan nilai dimensi y menjadi $tampung+2$. Dengan mengetahui nilai tampung yang diperoleh dari nilai y terbesar dari citra A, B, dan C, maka seluruh dimensi citra akan berubah mejadi A($x_a+2,tampung+2$), B($x_b+2,tampung+2$), C($x_c+2,tampung+2$). Pada perhitungan nilai terbesar dari tinggi citra, diperoleh nilai tampung $y=55$, maka diperoleh dimensi baru untuk setiap citra A, B dan C yang dapat dilihat pada *Tabel 3.1*.

Tabel 3.1. Tabel Matrix Dimensi Pemberian Padding

Citra	Sebelum Padding(x,y)	Setelah Padding(x,y)
A	(30,24)	(32,57)
B	(40,40)	(42,57)
C	(37,55)	(39,57)

Pemberian garis tepi/pembatas citra pada tepi atas dan bawah citra dilakukan dengan memberikan garis berwarna merah(255,0,0) pada seluruh koordinat($x_i,0$) dan seluruh koordinat($x,tinggi_citra_i+1$). Sedangkan pemberian garis tepi/pembatas citra pada tepi kiri dan kanan citra dilakukan dengan memberikan garis berwarna merah(255,0,0) pada seluruh koordinat($0,y_i$) dan seluruh koordinat($(lebar_citra_i+1),y_i$). Untuk pemberian *padding* dilakukan

dengan memberikan warna hitam(0,0,0) pada koordinat((x_i ,($tinggi_citra_i+1$)) sampai dengan (x_i ,($tampung+1$)).

3.5.1.3 Perhitungan Lebar Citra Gabungan

Perhitungan lebar citra adalah proses perhitungan yang dilakukan setelah mendapat nilai dimensi dari citra A,B, dan C setelah diberikan *padding*. Penghitungan lebar citra dilakukan dengan cara menjumlahkan seluruh nilai x dari setiap citra. Cara ini dapat dituliskan dengan rumus :

$$x = \sum_{i=0}^n x_i$$

Contoh perhitungan nilai x pada citra gabungan dapat diaplikasikan pada citra hasil *padding* A,B, dan C yang dapat dilihat pada *Tabel 3.1*

Berdasarkan perhitungan *padding*, dimensi citra setelah *padding* dari masing-masing citra A,B, dan C adalah :

- A(32,57)
- B(42,57)
- C(39,57)

Lebar citra gabungan(x) :

- $x = \sum_{i=0}^2 x_i$
- $x = x_0 + x_1 + x_2$
- $x = 32 + 42 + 39$
- $x = 113$

3.5.1.4 Perhitungan Dimensi Citra Gabungan

Dimensi citra gabungan didapatkan dari jumlah keseluruhan lebar citra setelah mengalami *padding* sebagai nilai x dan tinggi citra setelah mengalami *padding* sebagai nilai y . Pada sub-bab sebelumnya telah membahas cara mendapatkan nilai-nilai tersebut. Untuk kasus citra A, B, dan C, telah diperoleh nilai x gabungan adalah 113 dan nilai y setelah mengalami *padding* adalah 57, maka nilai dimensi citra gabungan dapat diketahui, yaitu ABC(113,57).

3.5.2 Perhitungan Dalam Proses Dekripsi

3.5.2.1 Perhitungan Jumlah Citra Pada Citra Gabungan

Untuk mengetahui jumlah citra yang terdapat dalam citra gabungan digunakan perhitungan berdasarkan jumlah dari garis pembatas *vertical* pada citra gabungan dibagi dua. Mengapa jumlah garis pembatas *vertical* dibagi dua? Nilai pembagi didapatkan karena setiap citra hasil padding pasti memiliki 2 garis pembatas di sisi kanan dan sisi kiri citra. Cara ini dapat dituliskan dalam rumus :

$$n = \frac{\sum v}{2}$$

n = jumlah citra, v = Garis pembatas *vertical*

Jika menggunakan kasus citra gabungan ABC(113,57), saat proses dijalankan akan ditemukan sebanyak enam garis pembatas *vertical*. Maka jumlah citra asli yang terdapat dalam citra ABC dapat dihitung dengan rumus diatas sebagai berikut :

- $n = \frac{\sum v}{2}$
- $n = \frac{6}{2}$
- $n = 3$

3.5.2.2 Perhitungan Menghilangkan Padding

Setelah menemukan jumlah citra asli dalam citra gabungan, maka sistem akan memulai proses identifikasi garis pembatas yang berwarna (255,0,0), saat menemukan garis pembatas yang pertama kali, citra asli tersebut akan langsung ditampung secara *vertical* ke dalam *list* dengan dimensi sebesar (1,y). Setiap ada kenaikan nilai x maka nilai citra akan ditampung ke dalam *list* selanjutnya. Proses penampungan ini akan berhenti jika sistem menemukan adanya garis *vertical* yang kedua. Maka dimensi citra dalam proses ini akan berubah menjadi ($x-2$, y). Contoh perhitungannya dimensi sementara setelah menghilangkan garis pembatas

vertical dapat menggunakan kasus citra A,B, dan C dengan rumus sebagai berikut :

$$y = y^1$$

y = tinggi citra asli

y^1 = tinggi citra sebelum proses menghilangkan *padding vertical*

$$x = x^1 - 2$$

x^1 = lebar citra sebelum proses menghilangkan *padding vertical*

x = lebar citra asli

Citra hasil pemisahan tersebut masih memiliki *padding* dan garis pembatas *horizontal*. Agar citra tersebut kembali ke bentuk aslinya, maka perhitungan menghilangkan *padding* menggunakan indikator garis pembatas *horizontal*, jika diketahui bahwa nilai piksel(x,0) dengan warna (255,0,0) sejumlah lebar dari citra itu sendiri, maka sistem akan mendeteksi bahwa piksel(x,0) merupakan garis pembatas *horizontal* yang dibuang, kemudian dilanjutkan dengan mengambil dan menampung setiap nilai piksel berikutnya dan memasukkan kedalam tampungan berupa *list* dengan dimensi (x,1), saat nilai y bertambah, maka tampungan berikutnya dimasukkan ke dalam *list* yang selanjutnya. Tampungan akan dihentikan ketika sistem menemukan garis pembatas *horizontal* yang kedua pada citra tersebut.

Proses selanjutnya adalah menuliskan kembali data dari hasil tampungan tersebut kedalam kanvas baru sebagai citra aslinya dengan ukuran lebar sama dengan lebar citra aslinya dan tinggi nya sejumlah *list* yang ada. Contoh perhitungannya dapat menggunakan kasus citra A,B, dan C pada dengan rumus sebagai berikut :

$$x = x^1$$

x = lebar citra asli

x^1 = lebar citra dengan *padding*

$$y = \sum t$$

t = tampungan dalam list

y = tinggi citra asli

Tabel 3.2. Tabel Hasil Menghilangkan Padding citra

Citra	Citra Gabungan	Setelah Dipisahkan
A	(32,57)	(30,24)
B	(42,57)	(40,40)
C	(39,57)	(37,55)

3.6 Rancangan Pengujian Sistem

3.6.1 Pengujian Terhadap Prosentase Keberhasilan Sistem

1. Tujuan
 - Mengetahui prosentase keberhasilan sistem
2. Metode
 - Menggunakan citra digital
 - Menggunakan model *padding* dengan garis tepi tiap citra
3. Alat dan Bahan
 - Citra digital dengan jumlah sampel sebanyak 40 citra dengan karakter yang berbeda-beda.
4. Hasil
 - Persentase keberhasilan proses penggabungan(enkripsi multi-file) dan proses pemisahan(dekripsi multi-file) dibandingkan dengan jumlah sampel pengujian.

3.6.2 Pengujian Terhadap Perbandingan Properti Citra Digital

1. Tujuan
 - Mengetahui perbedaan citra asli dengan citra hasil dekripsi beserta penyebabnya
2. Metode

- Menggunakan citra digital
 - Menggunakan model *padding* dengan garis tepi tiap citra
3. Alat dan Bahan
- Citra digital dengan jumlah sampel sebanyak 40 citra dengan karakter yang berbeda-beda.
4. Hasil
- Membandingkan histogram citra digital asli dengan citra digital hasil dekripsi.
 - Membandingkan dimensi citra digital asli dengan citra digital hasil dekripsi
 - Membandingkan ukuran file citra digital asli dengan citra digital hasil dekripsi

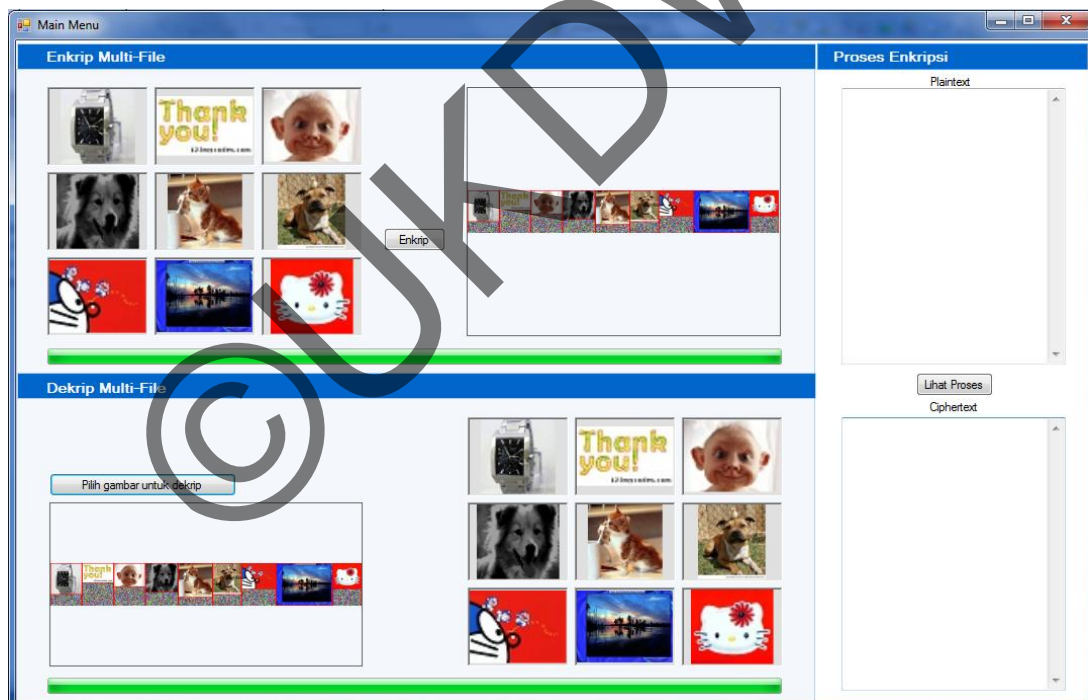
©UKDW

BAB IV IMPLEMENTASI DAN ANALISIS SISTEM

4.1 Implementasi Sistem

4.1.1 Form Main Menu

Gambar 4.1 menunjukkan implementasi *form* Main Menu. Pada *form* ini pengguna dapat melakukan enkrip *multi-file* atau melakukan dekrip *multi-file*, enkrip dan dekrip terletak pada kolom bagian kiri, pengguna juga dapat melihat proses yang terjadi dalam proses enkrip maupun proses dekrip pada kolom bagian kanan.



Gambar 4.1. Form Main Menu

4.1.1.1 Form Main Menu Bagian Enkripsi

Pada bagian enkrip *multi-file*, pengguna diminta untuk memberikan inputan berupa citra digital dengan jumlah 1-9 citra digital pada kotak-kotak kecil di sisi kiri. Yang kemudian tombol “enkrip” digunakan untuk memulai proses enkripsi *multi-file* tersebut, di sisi kanan ada kotak citra yang berukuran besar untuk menampilkan hasil dari penggabungan dan penambahan *padding* citra yang diinputkan oleh pengguna. Di bagian bawah terdapat *progress-bar* yang menunjukkan proses dari enkrip *multi-file* itu sendiri. *Gambar 4.2* adalah kolom enkrip *multi-file* dalam form main menu.



Gambar 4.2. Form Main Menu Bagian Enkripsi

4.1.1.2 Form Main Menu Bagian Dekripsi

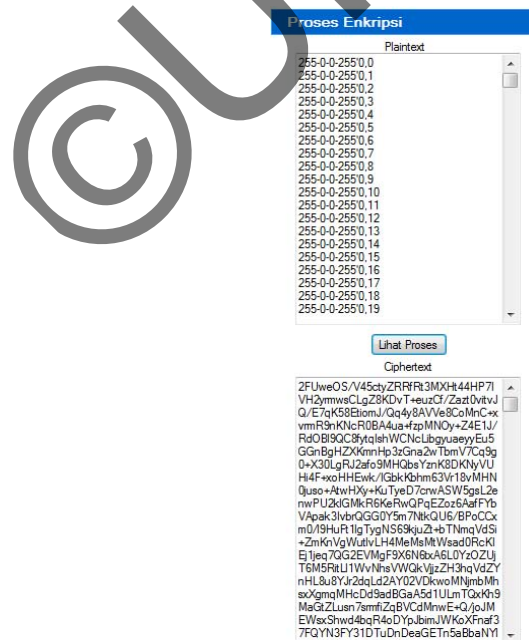
Untuk dekripsi *multi-file*, pengguna diminta untuk memberikan inputan berupa sebuah *cipher-image* dengan cara menekan tombol “pilih gambar untuk dekrip”. Citra hasil dekripsi akan ditampilkan pada kotak citra yang berukuran besar pada bagian kiri, kemudian hasil pemisahan dan penghilangan *padding* citra akan ditampilkan pada kotak-kotak citra yang berukuran kecil pada bagian kanan. *Gambar 4.3* merupakan kolom dekrip *multi-file* dalam form main menu.



Gambar 4.3. Form Main Menu Bagian Dekripsi

4.1.1.3 Form Main Menu Bagian Melihat Proses

Gambar 4.4 adalah tampilan *form* main menu-proses. Pada bagian ini pengguna dapat melihat proses enkripsi *multi-file* maupun proses dekripsi *multi-file*, bagian ini akan menampilkan data string sebelum proses enkripsi dan data string setelah proses enkripsi, pengguna hanya. Bagian ini bersifat pilihan, pengguna dapat melihat proses tersebut atau mengabaikannya, untuk melihat proses tersebut pengguna hanya tinggal menekan tombol “lihat proses”.



Gambar 4.4. Form Main Menu Bagian Melihat Proses

4.2 Implementasi Proses

4.2.1 Memberikan Padding

Implementasi proses memberikan *padding* dapat dilihat pada *Gambar 4.5*. Pada *pseudocode* memberikan *padding*, baris 5 – 8 merupakan implementasi proses memberikan garis tepi citra secara vertikal, garis tepi horizontal pada baris 9-12, sedangkan pemberian *padding* ada pada baris 13-16. Baris 17-25 digunakan untuk mengambil nilai dalam citra ke-*i* dengan koordinat (x,y), baris 27 untuk memberikan nilai *pixel* pada tampungan *Paddingimage*. Setelah proses tersebut selesai dijalankan, maka hasilnya akan ditampung ke dalam *list Paddingtampung* pada baris ke-30 yang nantinya akan digunakan pada proses selanjutnya, yaitu proses menggabungkan citra.

4.2.2 Menggabungkan Citra

Implementasi proses menggabungkan citra dapat dilihat pada *Gambar 4.6*. Baris 6-17 digunakan untuk mengambil nilai *pixel* dari *list* tampungan *Paddingtampung*, kemudian nilai *pixel* tersebut akan dituliskan pada variabel *Tampung* di baris ke-19. Variabel *Tampung* akan digunakan pada proses selanjutnya, yaitu proses konversi tipe data *bitmap* ke tipe data *string*.

```

1. FOR i = 0 sampai (jumlah citra)-1
2.  paddimage ← citra yang mempunyai dengan ukuran lebar
   citra baru yang didapat dari lebar citra ke-i + 2 dan
   ukuran tinggi dari citra
3.  FOR y = 0 sampai tinggi-1
4.    FOR x = 0 sampai lebar citra ke-i + 1
5.      IF x = 0 atau x = lebar citra ke-i + 1 THEN
6.        r ← 150
7.        g ← y Mod 256
8.        b ← 150
9.        a ← 255
10.     ELSEIF y = 0 atau y = tinggi citra ke-i + 1 THEN
11.       r ← 150
12.       g ← 150
13.       b ← x Mod 256
14.       a ← 255
15.     ELSEIF y > tinggi citra ke-i + 1 THEN
16.       r = bilangan acak dari 0 sampai 255
17.       g = bilangan acak dari 0 sampai 255
18.       b = bilangan acak dari 0 sampai 255
19.       a = 255
20.     ELSE
21.       TRY
22.         WITH mengambil nilai pixel dari citra ke-i di
           koordinat (x,y)
23.         r = konversi warna merah menjadi integer
24.         g = konversi warna hijau menjadi integer
25.         b = konversi warna biru menjadi integer
26.         a = konversi warna opacity menjadi integer
27.       END WITH
28.       CATCH ex berupa Exception
29.     END TRY
30.   END IF
31.   Memberi nilai pixel dari citra ke paddimage(x,y)
32. NEXT x
33. NEXT y
34. Menambahkan nilai paddimage pada padingtampung
35. NEXT i

```

Gambar 4.5. Pseudocode Menambahkan Padding Citra

```

1. Nilai_awal ← 1
2. Terproses ← 0
3. Tampung ← citra dengan lebar dan tinggi
4. FOR x = 0 sampai lebar - 1
5.   FOR y = 0 sampai tinggi - 1
6.     IF x < lebarcitra(nilai_awal) THEN
7.       TRY
8.         WITH paddingtampung(nilai_awal - 1).GetPixel(x-
           terproses, y)
9.           r = konversi warna merah menjadi integer
10.          g = konversi warna hijau menjadi integer
11.          b = konversi warna biru menjadi integer
12.        END WITH
13.       CATCH ex berupa Exception
14.     END TRY
15.   ELSE
16.     terproses ← lebarcitra(nilai_awal)
17.     nilai_awal ← nilai_awal + 1
18.   END IF
19.   Memberi nilai pixel (a,r,g,b) pada koordinat (x,y) ke dalam
       variabel tampung
20.   NEXT y
21. NEXT x

```

Gambar 4.6. Pseudocode Menggabungkan Citra

4.2.3 Konversi Tipe Data Bitmap ke Tipe Data String

Gambar 4.7 adalah *pseudocode* implementasi proses konversi tipe data bitmap ke tipe data string. Proses ini akan mengubah tipe data bitmap dari tampungan variabel Tampung ke tipe data string dengan mencatat setiap informasi nilai warna (r-g,b) beserta koordinat (x,y) setiap pixel secara berurutan dari (0,0), (1,0), (2,0), ..., (x,y). Setelah semua informasi tercatat, di akhir baris akan ditambahkan informasi nilai lebar dan tinggi citra dari variabel tampung.

```

1. FOR x = 0 sampai lebar citra pada variabel tampung -1
2.   FOR y = 0 sampai tinggi citra pada variabel tampung - 1
3.     r,g,b,a ← Nilai pixel dalam Integer
4.     Plaintext &= (r & "-" & g & "-" & b & "-" & a & "'" & x
       & "," & y & vbNewLine)
5.   NEXT y
6. NEXT x
7. plaintext &= "|" & lebar citra dari variabel tampung &
   "*" & tinggi citra di variabel tampung

```

Gambar 4.7. Pseudocode Konversi Tipe Data Bitmap ke Tipe Data String

4.2.4 Konversi Tipe Data String ke Tipe Data Bitmap

Pseudocode konversi tipe data *string* ke tipe data *bitmap* dapat dilihat pada *Gambar 4.8*. Konversi ini dilakukan dengan memisah-misahkan informasi dalam *string*. Pemisahan pertama dilakukan dengan mengidentifikasi karakter (`()`), pecahan pertama sebelum tanda tersebut merupakan informasi nilai setiap pixel beserta dengan koordinatnya, sedangkan pecahan kedua setelah tanda tersebut merupakan nilai lebar dan tinggi citra pada baris ke 1-2. Pecahan pertama kemudian dipecah lagi menjadi 2 bagian berdasarkan tanda (`(')`), `a` sebelum tanda tersebut dan `b` setelah tanda tersebut, `a` menyimpan nilai warna, sedangkan `b` menyimpan nilai koordinatnya, pemisahan ini terdapat pada baris 11-23.

4.2.5 Memisahkan Citra

Pseudocode memisahkan citra dapat dilihat pada *Gambar 4.9*. *Picture box* yang terdapat pada *pseudocode* tersebut berisi citra digital hasil konversi dari tipe data *string* ke tipe data *bitmap*. Proses melakukan pemeriksaan garis pembatas citra secara *vertical* terdapat pada baris 14 – 31. Setelah garis pembatas ditemukan, sistem akan menampung setiap nilai dalam pixel citra tersebut pada variabel *Tampungan* hingga sistem menemukan adanya garis pembatas berikutnya, penerapannya ada pada baris 33-42. Setelah menemukan 2 garis pembatas, sistem akan menganggapnya sebagai garis tepi citra asli, maka sistem akan menuliskan variabel *Tampungan* ke dalam variabel *Gambar*, penerapan ini terdapat pada baris 43-57. Baris 63-65 digunakan untuk menambahkan variabel *tampungan* ke dalam *list Temp*.

4.2.6 Menghilangkan Padding

Pseudocode menghilangkan *padding* terdapat pada *Gambar 4.10*. Variabel *Pisahtampung* merupakan variabel *tampungan* yang dihasilkan dari proses Memisahkan citra. Sebelum melakukan proses ini, sistem terlebih dulu akan

membersihkan *list* Temp pada baris 4. Dalam mengidentifikasi garis tepi *horizontal* pada citra digital dilakukan pada baris 12-31. Jika garis tersebut telah ditemukan, maka sistem akan menampung seluruh nilai *pixel* dalam variabel *Tampungan1* hingga sistem menemukan garis tepi yang kedua kalinya, implementasi terdapat pada baris 32-41. Selanjutnya sistem akan menuliskan informasi nilai *pixel* yang telah ditampung dalam variabel *Tampungan1* kedalam variabel *Imag*, baris 42- 61. Baris 62-65 digunakan untuk menambahkan variabel *Tampungan1* kedalam *list* Temp.

```

1. Set width dengan memisahkan string dengan
   Cint(stringplaintext.Split(CChar("|"))(1).Split
   (CChar("*"))(0))
2. Set hight dengan memisahkan string dengan
   Cint(stringplaintext.Split(CChar("|"))(1).Split
   (CChar("*"))(1))
3. Bild ← citra dengan ukuran panjang dan tinggi
4. stringplaintext =
   stringplaintext.Split(CChar("|"))(0)
5. richbox ← mengisi richtextbox pada form
6. richbox.Text = stringplaintext
7. FOR EACH line yang ada di richbox
8.   IF line tidak mempunyai nilai THEN
9.     Continue For
10.  END IF
11. farbe ← line.Split(CChar(" "))(0)
12. r,g,b,a ← nilai pixel dalam integer
13. geteilt() ← nilai farbe yang dipisah menggunakan
   tanda '-'
14. TRY
15.   r = pecahan pertama
16.   g = pecahan kedua
17.   b = pecahan ketiga
18.   a = pecahan keempat
19. CATCH ex berupa Exception
20. END TRY
21. X ← mengambil nilai dari karakter yang ditentukan
   yang terdapat pada baris
22. Y ← mengambil nilai dari karakter yang ditentukan
   yang terdapat pada baris
23. c ← nilai pixel yang merupakan gabungan dari a,
   r, g, b
24. NEXT
25. Memberi nilai pixel pada koordinat(x,y) dengan
   warna c pada variabel bild

```

Gambar 4.8. Pseudocode Konversi Tipe Data String ke Tipe Data Bitmap

```

1. Tinggi ← nilai tinggi sebuah citra yang terdapat pada PictureBox
2. Lebar ← nilai lebar sebuah citra yang terdapat pada PictureBox
3. Hitung ← mempunyai nilai awal 0
4. Picture ← citra yang diambil dari PictureBox dengan lebar dan tinggi sebagai
   parameternya
5. jumlah ← mempunyai nilai awal 0
6. garis ← mempunyai nilai awal 0
7. flag ← mempunyai nilai awal false
8. FOR x = 0 sampai lebar -1
9.     Tampungan ← citra dengan parameter (1, tinggi)
10.    Flag = True
11.    FOR y = 0 sampai tinggi - 1
12.        Hitung = 0
13.        IF y = 0 THEN
14.            FOR cek = 0 sampai tinggi - 1
15.                TRY
16.                    WITH pixel dari citra dengan parameter (x,cek)
17.                        r ← nilai pixel warna merah dikonversi menjadi Integer
18.                        g ← nilai pixel warna hijau dikonversi menjadi Integer
19.                        b ← nilai pixel warna biru dikonversi menjadi Integer
20.                    END WITH
21.                    CATCH ex berupa Exception
22.                    IF r = 150 dan g = cek Mod 256 dan b = 150 THEN
23.                        hitung += 1
24.                    END IF
25.                NEXT cek
26.            IF hitung = tinggi THEN
27.                hitung = 0
28.                Garis = garis + 1
29.                flag = false
30.                y = tinggi - 1
31.            END IF
32.        END IF
33.    IF flag = True THEN
34.        TRY
35.            WITH mengambil nilai pixel dari variabel picture
36.                r ← nilai pixel warna merah dikonversi menjadi Integer
37.                g ← nilai pixel warna hijau dikonversi menjadi Integer
38.                b ← nilai pixel warna biru dikonversi menjadi Integer
39.            END WITH
40.            CATCH ex sebagai Exception
41.        END TRY
42.        SET pixel variabel tampungan (0, y, Color.FromArgb(255, r, g, b))
43.    ELSEIF flag = False AND garis MOD 2 = 0 THEN
44.        gambar ← citra dengan parameter (temp.Count, tinggi)
45.        FOR m = 0 sampai temp.Count - 1
46.            FOR n = 0 sampai tinggi - 1
47.                TRY
48.                    WITH pixel dari variabel temp ke-m dengan parameter (0,n)
49.                        r ← nilai pixel warna merah dikonversi menjadi Integer
50.                        g ← nilai pixel warna hijau dikonversi menjadi Integer
51.                        b ← nilai pixel warna biru dikonversi menjadi Integer
52.                    END WITH
53.                    CATCH ex sebagai Exception
54.                END TRY
55.                SET pixel variabel gambar dengan parameter (m, n, Color.FromArgb(255, r,
g, b))
56.            NEXT n
57.        NEXT m
58.        jumlah += 1
59.        PISAHTAMPUNG.Add(gambar)
60.        Temp.Clear()
61.    END IF
62. NEXT y
63. IF flag = True THEN
64.     Temp.Add(tampungan)
65. END IF
66. NEXT x

```

Gambar 4.9. Pseudocode Memisahkan Citra

```

1. hitung ← nilai awal 0
2. jumlah ← nilai awal 0
3. garis ← nilai awal 0
4. temp.Clear()
5. FOR i = 0 sampai pisahtampung.Count - 1
6.   garis = 0
7.   FOR y = 0 sampai pisahtampung(i).Height - 1
8.     Tampungannya1 ← citra dengan parameter (pisahtampung(i).Width, 1)
9.     flag = True
10.    FOR x = 0 sampai pisahtampung(i).Width - 1
11.      hitung = 0
12.      IF x = 0 THEN
13.        FOR cek sampai pisahtampung(i).Width - 1
14.          TRY
15.            WITH pixel dari variabel pisahtampung ke-i dengan parameter (cek, y)
16.              r ← nilai pixel warna merah dikonversi menjadi Integer
17.              g ← nilai pixel warna hijau dikonversi menjadi Integer
18.              b ← nilai pixel warna biru dikonversi menjadi Integer
19.            END WITH
20.            CATCH ex sebagai Exception
21.          END TRY
22.          IF r = 150 And g = 150 And b = cek Mod 256 THEN
23.            hitung += 1
24.          END IF
25.        NEXT cek
26.      IF hitung = pisahtampung(i).Width THEN
27.        garis += 1
28.        x = pisahtampung(i).Width - 1
29.        flag = False
30.      END IF
31.    END IF
32.  IF flag = True THEN
33.    TRY
34.      WITH pixel dari variabel pisahtampung ke-i dengan parameter (x, y)
35.        r ← nilai pixel warna merah dikonversi menjadi Integer
36.        g ← nilai pixel warna hijau dikonversi menjadi Integer
37.        b ← nilai pixel warna biru dikonversi menjadi Integer
38.      END WITH
39.      CATCH ex sebagai Exception
40.    END TRY
41.    SET pixel variabel tampungannya1 dengan parameter (x, 0, Color.FromArgb(255,
r, g, b))
42.  ELSEIF flag = False dan garis = 2 THEN
43.    hitung = 0
44.    imag ← citra dengan parameter (pisahtampung(i).Width, temp.Count)
45.    FOR n = 0 sampai temp.Count - 1
46.      FOR m = 0 sampai pisahtampung(i).Width - 1
47.        TRY
48.          WITH pixel dari variabel temp ke-n dengan parameter (x, y)
49.            r ← nilai pixel warna merah dikonversi menjadi Integer
50.            g ← nilai pixel warna hijau dikonversi menjadi Integer
51.            b ← nilai pixel warna biru dikonversi menjadi Integer
52.          END WITH
53.          CATCH ex sebagai Exception
54.        END TRY
55.        SET pixel variabel imag dengan parameter (m, n, Color.FromArgb(255, r,
g, b))
56.      NEXT m
57.    NEXT n
58.    jumlah = jumlah + 1
59.    y = pisahtampung(i).Height - 1
60.    temp.Clear()
61.  ENDIF
62. NEXT x
63. IF flag = True Then
64.   temp.Add(tampungannya1)
65. END IF
66. NEXT y
67. NEXT i
68. jenisproses = 2

```

Gambar 4.10. Pseudocode Menghilangkan Padding

4.2.7 Melihat Proses

Gambar 4.11 adalah pseudocode melihat proses, baik enkripsi maupun dekripsi. Baris 1-5 digunakan untuk menampilkan proses enkripsi dengan menampilkan *plain-text* dan dilanjutkan *cipher-text*, baris 6-10 untuk menampilkan proses dekripsi dengan menampilkan *cipher-text* terlebih dulu kemudian *plain-text*, sedangkan baris 11-12 digunakan ketika pengguna belum melakukan satu proses pun.









```
1. IF jenisproses = 1 THEN
2.   tampilkan kotak pesan dengan konten "Proses Enkripsi"
3.   textbox1.Text = plaintext
4.   cipher ← konversi ciphertext ke bentuk base64
5.   Textbox2.Text = cipher
6. ELSEIF jenisproses = 2 THEN
7.   tampilkan kotak pesan dengan konten "Proses Dekripsi"
8.   cipher ← konversi ciphertext ke bentuk base64
9.   textbox2.Text = cipher
10.  textbox1.Text = plaintext
11. ELSE
12.  Tampilkan kotak pesan dengan konten "Belum terjadi
    proses Enkripsi ataupun Dekripsi"
13. ENDIF
```








Gambar 4.11. Pseudocode Melihat Proses

4.3 Analisis dan Pengujian Sistem











Tabel 4.1 merupakan tabel perbandingan citra sebelum di-*enkripsi* dan setelah di-*dekripsi*. Tabel ini menampilkan seluruh hasil percobaan dengan jumlah sebanyak 40 sampel citra digital.









Tabel 4.1. Tabel Perbandingan Citra Asli dan Citra Hasil Dekripsi









No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
				
1	Citra			0
	Dimensi	55 x 55	55 x 55	0
	Ukuran File	7,70 KB	7,70 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
2	Citra			0
	Dimensi	55 x 34	55 x 34	0
	Ukuran File	4,46 KB	4,46 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
3	Citra			0
	Dimensi	55 x 41	55 x 41	0
	Ukuran File	6,74 KB	6,74 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
4	Citra			0
	Dimensi	57 x 51	57 x 51	0
	Ukuran File	4,42 KB	4,42 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0










No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
5	Citra			0
	Dimensi	60 x 60	60 x 60	0
	Ukuran File	11,5 KB	11,5 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
6	Citra			0
	Dimensi	50 x 55	50 x 55	0
	Ukuran File	8,90 KB	8,90 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
7	Citra			0
	Dimensi	60 x 45	60 x 45	0
	Ukuran File	6,30 KB	6,30 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
8	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	22,2 KB	22,2 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
9	Citra			0
	Dimensi	50 x 45	50 x 45	0
	Ukuran File	5,77 KB	5,77 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
				
10	Citra			0
	Dimensi	45 x 40	45 x 40	0
	Ukuran File	5,85 KB	5,85 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
11	Citra			0
	Dimensi	55 x 65	55 x 65	0
	Ukuran File	9,85 KB	9,85 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
12	Citra			0
	Dimensi	75 x 86	75 x 86	0
	Ukuran File	7,92 KB	7,92 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
13	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	23,2 KB	23,2 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
14	Citra			0
	Dimensi	39 x 55	39 x 55	0
	Ukuran File	5,86 KB	5,86 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
15	Citra			0
	Dimensi	40 x 35	40 x 35	0
	Ukuran File	3,68 KB	3,68 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
16	Citra			0
	Dimensi	45 x 60	45 x 60	0
	Ukuran File	8,04 KB	8,04 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
17	Citra			0
	Dimensi	80 x 53	80 x 53	0
	Ukuran File	13,8 KB	13,8 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
18	Citra			0
	Dimensi	60 x 60	60 x 60	0
	Ukuran File	2,69 KB	2,69 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
				
19	Citra			0
	Dimensi	70 x 56	70 x 56	0
	Ukuran File	9,47 KB	9,47 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
20	Citra			0
	Dimensi	80 x 76	80 x 76	0
	Ukuran File	17,1 KB	17,1 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
21	Citra			0
	Dimensi	60 x 43	60 x 43	0
	Ukuran File	7,29 KB	7,29 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
22	Citra			0
	Dimensi	60 x 46	60 x 46	0
	Ukuran File	7,63 KB	7,63 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
23	Citra			0
	Dimensi	80 x 60	80 x 60	0
	Ukuran File	13,6 KB	13,6 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
24	Citra			0
	Dimensi	80 x 60	80 x 60	0
	Ukuran File	15 KB	15 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
25	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	23 KB	23 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
26	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	18,6 KB	18,6 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
27	Citra			0
	Dimensi	80 x 60	80 x 60	0
	Ukuran File	15,1 KB	15,1 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
				
28	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	13,6 KB	13,6 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
29	Citra			0
	Dimensi	50 x 37	50 x 37	0
	Ukuran File	6 KB	6 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0
30	Citra			0
	Dimensi	80 x 60	80 x 60	0
	Ukuran File	10,7 KB	10,7 KB	0
	Resolution	96 dpi	96 dpi	0
	Bit Depth	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
31	Citra			0
	Dimensi	80 x 60	80 x 60	0
	Ukuran File	14 KB	14 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
32	Citra			0
	Dimensi	100 x 75	100 x 75	0
	Ukuran File	24 KB	24 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
33	Citra			0
	Dimensi	60 x 39	60 x 39	0
	Ukuran File	7,22 KB	7,22 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
34	Citra			0
	Dimensi	70 x 46	70 x 46	0
	Ukuran File	7,68 KB	7,68 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
35	Citra			0
	Dimensi	60 x 45	60 x 45	0
	Ukuran File	8,09 KB	8,09 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
36	Citra			0
	Dimensi	60 x 41	60 x 41	0
	Ukuran File	5,04 KB	5,04 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
				
37	Citra			0
	Dimensi	56 x 56	56 x 56	0
	Ukuran File	0,5 KB	0,5 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
38	Citra			0
	Dimensi	39 x 55	39 x 55	0
	Ukuran File	6,1 KB	6,1 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0
39	Citra			0
	Dimensi	147 x 97	147 x 97	0
	Ukuran File	38,5 KB	38,5 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

No.	Keterangan	Citra Asli	Citra Hasil Dekripsi	Perbedaan
40	Citra			0
	Dimensi	60 x 45	60 x 45	0
	Ukuran File	7,16 KB	7,16 KB	0
	Resolution	96 dpi	96 dpi	0
	<i>Bit Depth</i>	32	32	0

4.3.1 Pengujian Terhadap Prosentase Keberhasilan Sistem

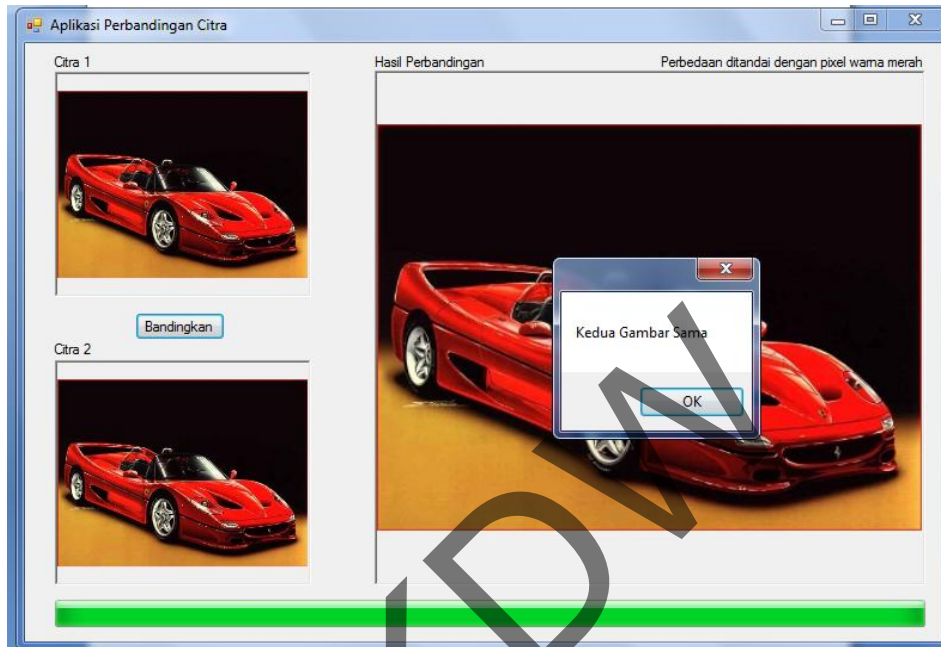
Pengujian ini bertujuan untuk mengetahui seberapa efektif model *padding* dan penggabungan citra baik pada proses enkripsi maupun dalam proses dekripsi, pengujian dilakukan dengan menggunakan sampel sebanyak 40 citra digital. Dari hasil percobaan pada *Tabel 4.1* diperoleh nilai prosentase keberhasilan sistem sebesar 100%. Angka ini diperoleh dari jumlah keberhasilan dibagi dengan total citra yang diuji kemudian dikalikan dengan 100.

4.3.2 Pengujian Terhadap Perbandingan Properti Citra Digital

Pengujian ini dilakukan dengan melakukan perbandingan dari sisi dimensi citra, ukuran file citra, dan citra digital itu sendiri. Membandingkan dimensi citra dilakukan dengan cara nilai x citra sesudah proses dekripsi dikurangi dengan nilai x citra asli dan nilai y citra sesudah proses dekripsi dikurangi nilai y citra asli. Nilai ukuran file (dalam satuan KB) citra sesudah proses dekripsi dikurangi dengan nilai ukuran file citra asli merupakan cara untuk menghitung perbedaan ukuran memori file citra.

Membandingkan citra digital itu sendiri dilakukan dengan cara membandingkan nilai warna *R*, *G*, *B*, dan *A* pada setiap piksel di koordinat yang sama dari cira asli dengan citra hasil dekripsi. Program yang digunakan adalah

“Aplikasi Perbandingan Citra”, aplikasi ini merupakan sebuah program sederhana buatan penulis. Tampilan hasil perbandingan ini dapat dilihat pada *Gambar 4.12*.



Gambar 4.12. Membandingkan Citra Digital dengan Aplikasi Perbandingan Citra

Dari hasil percobaan pada *Tabel 4.1* dapat dilihat dari sisi citra digital antara citra asli dengan citra hasil dekripsi, dimensi citra, ukuran file citra digital, *resolution*, dan *bit-depth* tidak terdapat adanya perbedaan.

LISTING PROGRAM

- Form Main Menu Bagian Enkripsi

```

Imports System.IO
Imports System.Drawing
Imports System.Math
Imports System.Windows.Forms
Imports System.Drawing.Imaging
Imports System.Drawing.Drawing2D
Imports System.Security.Cryptography

Public Class MainMenu
    Dim box, temp, paddingtampung, pisahampung As List(Of Bitmap)
    Dim lebarcitra As List(Of Integer)
    Dim r, g, b, a As Integer
    Dim kpic1, kpic2, kpic3, kpic4, kpic5, kpic6, kpic7, kpic8, kpic9 As Integer
    Dim ciphertext As Byte()
    Dim plaintext, cipher, txtSourceCode As String

    Dim jenisproses As Integer = 0

    Dim key() As Byte = System.Text.Encoding.ASCII.GetBytes("asdfghjklzxcvbnm")
    Dim IV() As Byte = System.Text.Encoding.ASCII.GetBytes("asdfghjklzxcvbnm")
    Private Property picImage As Object
    Dim acak As New Random

    Dim encodeType As ImageFormat = ImageFormat.Jpeg
    Dim decodingString As String = String.Empty

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
        Handles MyBase.Load

        box = New List(Of Bitmap)
        lebarcitra = New List(Of Integer)
        temp = New List(Of Bitmap)
        paddingtampung = New List(Of Bitmap)
        pisahampung = New List(Of Bitmap)
        kpic1 = kpic2 = kpic3 = kpic4 = kpic5 = kpic6 = kpic7 = kpic8 = kpic9 = 0

    End Sub

    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click
        kpic1 = kpic1 + 1
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                PictureBox1.Image = Image.FromFile(.FileName)
                PictureBox1.SizeMode = PictureBoxSizeMode.Zoom
            End If
        End With
        If kpic1 > 1 Then
            box.ElementAt(0)
            box(0) = PictureBox1.Image

```

```

Else
    box.Add(PictureBox1.Image)
End If
'MessageBox.Show(box.Count)

End Sub

Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
    kpic2 = kpic2 + 1
    If PictureBox1.Image Is Nothing Then
        MessageBox.Show("Box 1 harus diisi")
    Else
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                PictureBox2.Image = Image.FromFile(.FileName)
                PictureBox2.SizeMode = PictureBoxSizeMode.Zoom

                End If
            End With
        End If
        If kpic2 > 1 Then
            box.ElementAt(1)
            box(1) = PictureBox2.Image
        Else
            box.Add(PictureBox2.Image)
        End If
        'MessageBox.Show(box.Count)
    End Sub

Private Sub PictureBox3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox3.Click
    kpic3 = kpic3 + 1

    If PictureBox2.Image Is Nothing Then
        MessageBox.Show("Box 2 harus diisi")
    Else
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                PictureBox3.Image = Image.FromFile(.FileName)
                PictureBox3.SizeMode = PictureBoxSizeMode.Zoom

                End If
            End With
        End If
        If kpic3 > 1 Then
            box.ElementAt(2)
            box(2) = PictureBox3.Image
        Else
            box.Add(PictureBox3.Image)
        End If
    End Sub

```

```

        'MessageBox.Show(box.Count)
    End If

    End Sub

    Private Sub PictureBox4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox4.Click
        kpic4 = kpic4 + 1
        If PictureBox3.Image Is Nothing Then
            MessageBox.Show("Box 3 harus diisi")
        Else
            Dim open As New OpenFileDialog

            With open
                .CheckFileExists = True
                .ShowReadOnly = False
                .Filter = "All files|*.*| bitmap files(*)|.bmp;*.png;*.jpg"
                .FilterIndex = 2
                If .ShowDialog = DialogResult.OK Then
                    PictureBox4.Image = Image.FromFile(.FileName)
                    PictureBox4.SizeMode = PictureBoxSizeMode.Zoom

                End If
            End With
        End If

        If kpic4 > 1 Then
            box.ElementAt(3)
            box(3) = PictureBox4.Image
        Else
            box.Add(PictureBox4.Image)
        End If
        ' MessageBox.Show(box.Count)

    End Sub

    Private Sub PictureBox5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox5.Click
        kpic5 = kpic5 + 1
        If PictureBox4.Image Is Nothing Then
            MessageBox.Show("Box 4 harus diisi")
        Else
            Dim open As New OpenFileDialog

            With open
                .CheckFileExists = True
                .ShowReadOnly = False
                .Filter = "All files|*.*| bitmap files(*)|.bmp;*.png;*.jpg"
                .FilterIndex = 2
                If .ShowDialog = DialogResult.OK Then
                    PictureBox5.Image = Image.FromFile(.FileName)
                    PictureBox5.SizeMode = PictureBoxSizeMode.Zoom

                End If
            End With
        End If

        If kpic5 > 1 Then
            box.ElementAt(4)
            box(4) = PictureBox5.Image
        Else
            box.Add(PictureBox5.Image)
        End If
        'MessageBox.Show(box.Count)

    End Sub

    Private Sub PictureBox6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox6.Click
        kpic6 = kpic6 + 1

```

```

If PictureBox5.Image Is Nothing Then
    MessageBox.Show("Box 5 harus diisi")
Else
    Dim open As New OpenFileDialog

    With open
        .CheckFileExists = True
        .ShowReadOnly = False
        .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
        .FilterIndex = 2
        If .ShowDialog = DialogResult.OK Then
            PictureBox6.Image = Image.FromFile(.FileName)
            PictureBox6.SizeMode = PictureBoxSizeMode.Zoom

            End If
        End With
    End If
    If kpic6 > 1 Then
        box.ElementAt(5)
        box(5) = PictureBox6.Image
    Else
        box.Add(PictureBox6.Image)
    End If
    'MessageBox.Show(box.Count)
End Sub

Private Sub PictureBox7_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox7.Click
    kpic7 = kpic7 + 1
    If PictureBox6.Image Is Nothing Then
        MessageBox.Show("Box 6 harus diisi")
    Else
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                PictureBox7.Image = Image.FromFile(.FileName)
                PictureBox7.SizeMode = PictureBoxSizeMode.Zoom

                End If
            End With
        End If
        If kpic7 > 1 Then
            box.ElementAt(6)
            box(6) = PictureBox7.Image
        Else
            box.Add(PictureBox7.Image)
        End If
        'MessageBox.Show(box.Count)
    End Sub

Private Sub PictureBox8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox8.Click
    kpic8 = kpic8 + 1
    If PictureBox7.Image Is Nothing Then
        MessageBox.Show("Box 7 harus diisi")
    Else
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2

```

```

        If .ShowDialog = DialogResult.OK Then
            PictureBox8.Image = Image.FromFile(.FileName)
            PictureBox8.SizeMode = PictureBoxSizeMode.Zoom

        End If
    End With
End If
If kpic8 > 1 Then
    box.ElementAt(7)
    box(7) = PictureBox8.Image
Else
    box.Add(PictureBox8.Image)
End If
'MessageBox.Show(box.Count)

End Sub

Private Sub PictureBox9_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox9.Click
    kpic9 = kpic9 + 1
    If PictureBox8.Image Is Nothing Then
        MessageBox.Show("Box 8 harus diisi")
    Else
        Dim open As New OpenFileDialog

        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                PictureBox9.Image = Image.FromFile(.FileName)
                PictureBox9.SizeMode = PictureBoxSizeMode.Zoom

            End If
        End With
    End If
    If kpic9 > 1 Then
        box.ElementAt(8)
        box(8) = PictureBox9.Image
    Else
        box.Add(PictureBox9.Image)
    End If
    'MessageBox.Show(box.Count)
End Sub

Private Sub TombolEnkripsi_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TombolEnkripsi.Click
    TextBox1.Text = ""
    TextBox2.Text = ""

    If PictureBox1.Image Is Nothing Then
        MessageBox.Show("file kosong, tidak ada yang dapat diproses")

    Else

        '##### Menghitung jumlah pixel untuk membuat kanvas yang baru
        #####
        Dim lebar, tinggi As Integer

        '##### Menghitung tinggi kanvas #####
        tinggi = 0
        For i = 0 To box.Count - 1
            If (box(i).Height > tinggi) Then

```



```

        tinggi = box(i).Height
    Else
        tinggi = tinggi
    End If
Next i

tinggi = tinggi + 2

'##### Memberikan padding #####

For i = 0 To box.Count - 1
    Dim paddimage As New Bitmap(box(i).Width + 2, tinggi)
    For y = 0 To tinggi - 1
        For x = 0 To box(i).Width + 1
            If (x = 0 Or x = box(i).Width + 1) Then
                r = 150
                g = y Mod 256
                b = 150
                a = 255

            ElseIf (y = 0 Or y = box(i).Height + 1) Then
                r = 150
                g = 150
                b = x Mod 256
                a = 255

            ElseIf (y > box(i).Height + 1) Then
                r = acak.Next(0, 255)
                g = acak.Next(0, 255)
                b = acak.Next(0, 255)
                a = 255

            Else
                Try
                    With box(i).GetPixel(x - 1, y - 1)
                        r = Convert.ToInt16(.R)
                        g = Convert.ToInt16(.G)
                        b = Convert.ToInt16(.B)
                        a = Convert.ToInt16(.A)
                    End With
                Catch ex As Exception
                End Try
            End If
            paddimage.SetPixel(x, y, Color.FromArgb(a, r, g, b))
        Next x
    Next y
    paddingtampung.Add(paddimage)
Next i

'##### Menghitung lebar kanfas #####
lebar = 0
lebarcitra.Add(lebar)

For i = 0 To paddingtampung.Count - 1
    lebar = (lebar) + (paddingtampung(i).Width)
    lebarcitra.Add(lebar)
Next i

pbe.Maximum = 2 * (lebar * tinggi)

'#####
#####
Penggabungan image

```

```

Dim nilai_awal As Integer = 1

Dim terproses As Integer = 0

Dim tampung As New Bitmap(lebar, tinggi)

For x = 0 To lebar - 1
    For y = 0 To tinggi - 1
        If (x < lebarcitra(nilai_awal)) Then
            Try
                With paddingtampung(nilai_awal - 1).GetPixel(x - terproses, y)
                    r = Convert.ToInt16(.R)
                    g = Convert.ToInt16(.G)
                    b = Convert.ToInt16(.B)
                    a = Convert.ToInt16(.A)
                End With
                Catch ex As Exception
                End Try
            Else
                terproses = lebarcitra(nilai_awal)
                nilai_awal = nilai_awal + 1
            End If
            tampung.SetPixel(x, y, Color.FromArgb(a, r, g, b))
            pbe.Value += 1
        Next y
    Next x

    '##### Menampilkan image hasil di box10 #####
    PictureBox10.Image = tampung

    PictureBox10.SizeMode = PictureBoxSizeMode.Zoom

    MessageBox.Show("Proses Penggabungan selesai, dilanjutkan proses enkrip citra
dengan AES")

    '##### Konversi dari nilai bitmap ke nilai string #####
    For x = 0 To tampung.Width - 1
        For y = 0 To tampung.Height - 1
            Dim r, g, b, a As Integer
            r = tampung.GetPixel(x, y).R
            g = tampung.GetPixel(x, y).G
            b = tampung.GetPixel(x, y).B
            a = tampung.GetPixel(x, y).A

            plaintext &= (r & "-" & g & "-" & b & "-" & a & "" & x & "," & y &
vbNewLine)
            pbe.Value += 1
        Next y
    Next x

    plaintext &= "|" & tampung.Width & "*" & tampung.Height

    '##### Enkripsi dengan librari AES #####
    Using myaes As Security.Cryptography.Aes = Security.Cryptography.Aes.Create()
        ' encrypt the string to an array of bytes.
        Dim encrypted As Byte() = EncryptStringToBytes_Aes(plaintext, key, IV)

        MessageBox.Show("proses enkripsi selesai")
    End Using

```

```

##### Menyimpan hasil enkripsi ke dalam drive penyimpanan
#####
Dim fileStream As IO.FileStream = New IO.FileStream("encrypt.jpg",
IO.FileMode.Create)
Try
    For i As Integer = 0 To encrypted.Length - 1
        fileStream.WriteByte(encrypted(i))
    Next i
Catch ex As Exception
Finally
    fileStream.Close()
End Try
End Using
MessageBox.Show("Proses enkripsi telah selesai, file telah tersimpan
dengan nama 'encrypt.jpg'")

End If

jenisproses = 1
End Sub

Private Sub TombolDekripsi_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TombolDekripsi.Click
    TextBox1.Text = ""
    TextBox2.Text = ""
    Dim open As New OpenFileDialog

    Using myAes As Security.Cryptography.Aes = Security.Cryptography.Aes.Create()
        With open
            .CheckFileExists = True
            .ShowReadOnly = False
            .Filter = "All files|*.*| bitmap files(*)|*.bmp;*.png;*.jpg"
            .FilterIndex = 2
            If .ShowDialog = DialogResult.OK Then
                Dim encrypted() As Byte = File.ReadAllBytes(open.FileName)

                ciphertext = encrypted

                ##### Dekripsi dari tipe data string
                #####
                Dim stringplaintext As String = DecryptStringFromBytes_Aes(encrypted,
key, IV)

                plaintext = stringplaintext

                ##### Konversi dari string ke bitmap #####
                Dim width As Integer =
CInt(stringplaintext.Split(CChar("|"))(1).Split(CChar("*"))(0))
                Dim height As Integer =
CInt(stringplaintext.Split(CChar("|"))(1).Split(CChar("*"))(1))
                Dim bild As Bitmap
                bild = New Bitmap(width, height)

                stringplaintext = stringplaintext.Split(CChar("|"))(0)
                Dim richbox As New RichTextBox()
                richbox.Text = stringplaintext

                For Each line In richbox.Lines

                    If line = "" Then
                        Continue For
                    End If

                    Dim farbe As String = line.Split(CChar(" "))(0)
                    Dim r, g, b, a As Integer

```

```

        Dim geteilt() As String = farbe.Split(CChar("-"))

        Try
            r = Convert.ToInt16(geteilt(0).Trim)
            g = Convert.ToInt16(geteilt(1).Trim)
            b = Convert.ToInt16(geteilt(2).Trim)
            a = Convert.ToInt16(geteilt(3).Trim)
        Catch ex As Exception
        End Try

        Dim x As Integer =
CInt(CDb1(line.Split(CChar(" "))(1).Split(CChar(","))(0)))
        Dim y As Integer =
CInt(CDb1(line.Split(CChar(" "))(1).Split(CChar(","))(1)))
        Dim c As Color = Color.FromArgb(a, r, g, b)
        bild.SetPixel(CInt(x), CInt(y), c)
    Next

    PictureBox11.Image = bild
    PictureBox11.SizeMode = PictureBoxSizeMode.Zoom

    Dim fileStream As IO.FileStream = New IO.FileStream("decrypt.jpg",
IO.FileMode.Create)
    Dim plainimagejoin() As Byte
    plainimagejoin = ImageToByte(bild)
    Try
        For i As Integer = 0 To plainimagejoin.Length - 1
            fileStream.WriteByte(plainimagejoin(i))
        Next i
    Catch ex As Exception
    Finally
        fileStream.Close()
    End Try
End If
End With
End Using
MessageBox.Show("Proses dekripsi selesai, dilanjutkan pemisahan dan unpadding pada
citra")

'##### Memisahkan Citra
#####
Dim tinggi, lebar, hitung As Integer
Dim jumlah As Integer = 0
Dim garis As Integer = 0
Dim flag As Boolean = False

tinggi = PictureBox11.Image.Height
lebar = PictureBox11.Image.Width
Dim picture As New Bitmap(lebar, tinggi)
hitung = 0
picture = PictureBox11.Image
pbd.Value = 0
pbd.Maximum = lebar * tinggi

For x = 0 To lebar - 1
    Dim tampungan As New Bitmap(1, tinggi)
    flag = True

    For y = 0 To tinggi - 1
        hitung = 0
        If (y = 0) Then
            For cek = 0 To tinggi - 1
                Try
                    With picture.GetPixel(x, cek)
                        r = Convert.ToInt16(.R)
                        g = Convert.ToInt16(.G)
                        b = Convert.ToInt16(.B)

```

```

        End With
    Catch ex As Exception
    End Try
    If (r = 150 And g = (cek) Mod 256 And b = 150) Then
        hitung = hitung + 1
    End If
Next cek

If (hitung >= tinggi - 2) Then
    hitung = 0
    garis = garis + 1
    flag = False
    y = tinggi - 1
End If
End If

If flag = True Then
    Try
        With picture.GetPixel(x, y)
            r = Convert.ToInt16(.R)
            g = Convert.ToInt16(.G)
            b = Convert.ToInt16(.B)
        End With
        Catch ex As Exception
        End Try

        tampungan.SetPixel(0, y, Color.FromArgb(255, r, g, b))

    ElseIf flag = False And garis Mod 2 = 0 Then
        hitung = 0
        Dim gambar As New Bitmap(temp.Count, tinggi)

        For m = 0 To temp.Count - 1
            For n = 0 To tinggi - 1
                Try
                    With temp(m).GetPixel(0, n)
                        r = Convert.ToInt16(.R)
                        g = Convert.ToInt16(.G)
                        b = Convert.ToInt16(.B)
                    End With
                    Catch ex As Exception
                    End Try
                    gambar.SetPixel(m, n, Color.FromArgb(255, r, g, b))
                Next n
            Next m

            jumlah = jumlah + 1
            pisahtampung.Add(gambar)
            temp.Clear()
        End If
        pbd.Value = pbd.Value + 1
    Next y

    If flag = True Then
        temp.Add(tampungan)
    End If
Next x

'##### menghilangkan padding #####
hitung = 0
jumlah = 0
garis = 0
temp.Clear()

For i = 0 To pisahtampung.Count - 1
    garis = 0
    For y = 0 To pisahtampung(i).Height - 1
        Dim tampungan1 As New Bitmap(pisahtampung(i).Width, 1)

```

```

flag = True
For x = 0 To pisahtampung(i).Width - 1
    hitung = 0

    If (x = 0) Then
        For cek = 0 To pisahtampung(i).Width - 1

            Try
                With pisahtampung(i).GetPixel(cek, y)
                    r = Convert.ToInt16(.R)
                    g = Convert.ToInt16(.G)
                    b = Convert.ToInt16(.B)
                End With
                Catch ex As Exception
                End Try
                If (r = 150 And g = 150 And b = (cek + 1) Mod 256) Then
                    hitung = hitung + 1
                End If
            Next cek
            If (hitung = pisahtampung(i).Width) Then
                garis = garis + 1
                x = pisahtampung(i).Width - 1
                flag = False
            End If
        End If

        If flag = True Then
            Try
                With pisahtampung(i).GetPixel(x, y)
                    r = Convert.ToInt16(.R)
                    g = Convert.ToInt16(.G)
                    b = Convert.ToInt16(.B)
                End With
                Catch ex As Exception
                End Try
                tampungan1.SetPixel(x, 0, Color.FromArgb(255, r, g, b))

            ElseIf (flag = False And garis = 2) Then
                hitung = 0
                Dim imag As New Bitmap(pisahtampung(i).Width, temp.Count)
                For n = 0 To temp.Count - 1
                    For m = 0 To pisahtampung(i).Width - 1
                        Try
                            With temp(n).GetPixel(m, 0)
                                r = Convert.ToInt16(.R)
                                g = Convert.ToInt16(.G)
                                b = Convert.ToInt16(.B)
                            End With
                            Catch ex As Exception
                            End Try
                            imag.SetPixel(m, n, Color.FromArgb(255, r, g, b))

                        Next m
                    Next n
                jumlah = jumlah + 1
                y = pisahtampung(i).Height - 1
                temp.Clear()
                If (jumlah = 1) Then
                    PictureBox12.Image = imag
                    PictureBox12.SizeMode = PictureBoxSizeMode.Zoom
                ElseIf (jumlah = 2) Then
                    PictureBox13.Image = imag
                    PictureBox13.SizeMode = PictureBoxSizeMode.Zoom
                ElseIf (jumlah = 3) Then
                    PictureBox14.Image = imag
                    PictureBox14.SizeMode = PictureBoxSizeMode.Zoom
                ElseIf (jumlah = 4) Then

```

```

        PictureBox15.Image = imag
        PictureBox15.SizeMode = PictureBoxSizeMode.Zoom
    ElseIf (jumlah = 5) Then
        PictureBox16.Image = imag
        PictureBox16.SizeMode = PictureBoxSizeMode.Zoom
    ElseIf (jumlah = 6) Then
        PictureBox17.Image = imag
        PictureBox17.SizeMode = PictureBoxSizeMode.Zoom
    ElseIf (jumlah = 7) Then
        PictureBox18.Image = imag
        PictureBox18.SizeMode = PictureBoxSizeMode.Zoom
    ElseIf (jumlah = 8) Then
        PictureBox19.Image = imag
        PictureBox19.SizeMode = PictureBoxSizeMode.Zoom
    ElseIf (jumlah = 9) Then
        PictureBox20.Image = imag
        PictureBox20.SizeMode = PictureBoxSizeMode.Zoom
    End If
End If

Next x
If flag = True Then
    temp.Add(tampungan1)
End If
Next y
Next i
pbd.Value = tinggi * lebar
jenisproses = 2

End Sub
Private Sub PictureBox12_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox12.Click
    sfdImage.ShowDialog()
    If sfdImage.FileName > "" Then
        PictureBox12.Image.Save(sfdImage.FileName)

    End If
End Sub
Private Sub PictureBox13_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox13.Click
    sfdImage.ShowDialog()
    If sfdImage.FileName > "" Then
        PictureBox13.Image.Save(sfdImage.FileName)

    End If
End Sub

Private Sub PictureBox14_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox14.Click
    sfdImage.ShowDialog()
    If sfdImage.FileName > "" Then
        PictureBox14.Image.Save(sfdImage.FileName)

    End If
End Sub

Private Sub PictureBox15_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox15.Click
    sfdImage.ShowDialog()
    If sfdImage.FileName > "" Then
        PictureBox15.Image.Save(sfdImage.FileName)

    End If
End Sub

Private Sub PictureBox16_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox16.Click
    sfdImage.ShowDialog()

```

```

        If sfdImage.FileName > "" Then
            PictureBox16.Image.Save(sfdImage.FileName)

        End If
    End Sub

    Private Sub PictureBox17_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox17.Click
        sfdImage.ShowDialog()
        If sfdImage.FileName > "" Then
            PictureBox17.Image.Save(sfdImage.FileName)

        End If
    End Sub

    Private Sub PictureBox18_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox18.Click
        sfdImage.ShowDialog()
        If sfdImage.FileName > "" Then
            PictureBox18.Image.Save(sfdImage.FileName)

        End If
    End Sub

    Private Sub PictureBox19_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox19.Click
        sfdImage.ShowDialog()
        If sfdImage.FileName > "" Then
            PictureBox19.Image.Save(sfdImage.FileName)

        End If
    End Sub

    Private Sub PictureBox20_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox20.Click
        sfdImage.ShowDialog()
        If sfdImage.FileName > "" Then
            PictureBox20.Image.Save(sfdImage.FileName)

        End If
    End Sub

    Public Shared Function ImageToByte(ByVal img As Image) As Byte()
        Dim imgStream As MemoryStream = New MemoryStream()

        img.Save(imgStream, System.Drawing.Imaging.ImageFormat.Png)

        imgStream.Close()
        Dim byteArray As Byte() = imgStream.ToArray()
        imgStream.Dispose()

        Return byteArray
    End Function

    Private Function UnicodeBytesToString(
ByVal bytes() As Byte) As String

        Return System.Text.Encoding.Unicode.GetString(bytes)
    End Function

    Shared Function EncryptStringToBytes_Aes(ByVal plainText As String, ByVal Key() As
Byte, ByVal IV() As Byte) As Byte()
        ' Check arguments.
        If Key Is Nothing OrElse Key.Length <= 0 Then
            Throw New ArgumentNullException("Key")
        End If
        If IV Is Nothing OrElse IV.Length <= 0 Then
            Throw New ArgumentNullException("Key")
        End If
    End Function

```



```

Dim encrypted() As Byte
' Create an Aes object
' with the specified key and IV.
Using aesAlg As Security.Cryptography.Aes = Security.Cryptography.Aes.Create()

    aesAlg.Key = Key
    aesAlg.IV = IV

    ' Create a decryptor to perform the stream transform.
    Dim encryptor As ICryptoTransform = aesAlg.CreateEncryptor(aesAlg.Key,
aesAlg.IV)
    ' Create the streams used for encryption.
    Using msEncrypt As New MemoryStream()
        Using csEncrypt As New CryptoStream(msEncrypt, encryptor,
CryptoStreamMode.Write)
            Using swEncrypt As New StreamWriter(csEncrypt)

                'Write all data to the stream.
                swEncrypt.Write(plainText)
            End Using
            encrypted = msEncrypt.ToArray()
        End Using
    End Using
End Using

' Return the encrypted bytes from the memory stream.
Return encrypted

End Function

Shared Function DecryptStringFromBytes_Aes(ByVal cipherText() As Byte, ByVal Key() As
Byte, ByVal IV() As Byte) As String
    ' Check arguments.

    If cipherText Is Nothing OrElse cipherText.Length <= 0 Then
        Throw New ArgumentNullException("cipherText")
    End If
    If Key Is Nothing OrElse Key.Length <= 0 Then
        Throw New ArgumentNullException("Key")
    End If
    If IV Is Nothing OrElse IV.Length <= 0 Then
        Throw New ArgumentNullException("Key")
    End If
    ' Declare the string used to hold
    ' the decrypted text.
    Dim plaintext As String = Nothing

    ' Create an Aes object
    ' with the specified key and IV.
    Using aesAlg As Security.Cryptography.Aes = Security.Cryptography.Aes.Create()

        aesAlg.Key = Key
        aesAlg.IV = IV
        ' Create a decryptor to perform the stream transform.
        Dim decryptor As ICryptoTransform = aesAlg.CreateDecryptor(aesAlg.Key,
aesAlg.IV)
        ' Create the streams used for decryption.
        Using msDecrypt As New MemoryStream(cipherText)

            Using csDecrypt As New CryptoStream(msDecrypt, decryptor,
CryptoStreamMode.Read)

                Using srDecrypt As New StreamReader(csDecrypt)

                    ' Read the decrypted bytes from the decrypting stream
                    ' and place them in a string.

```

```

        plaintext = srDecrypt.ReadToEnd()
    End Using
    End Using
    End Using
    End Using

    Return plaintext

End Function

Private Sub BtnLihatProses_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnLihatProses.Click
    If (jenisproses = 1) Then
        MessageBox.Show("Proses Enkripsi")
        TextBox1.Text = plaintext
        MessageBox.Show("Plaintext dalam tipe data string")

        cipher = System.Convert.ToBase64String(ciphertext)

        TextBox2.Text = cipher
        MessageBox.Show("Ciphertext dalam tipe data string")
    ElseIf (jenisproses = 2) Then
        MessageBox.Show("Proses Dekripsi")
        cipher = System.Convert.ToBase64String(ciphertext)

        TextBox2.Text = cipher
        MessageBox.Show("Ciphertext dalam tipe data string")

        TextBox1.Text = plaintext
        MessageBox.Show("Plaintext dalam tipe data string")
    Else
        MessageBox.Show("Belum terjadi adanya proses Dekripsi atau Enkripsi")
    End If
End Sub

Public Function ImageToBase64(ByVal image As Image, ByVal format As ImageFormat)
    Using ms As New MemoryStream()
        image.Save(ms, format)
        Dim imageBytes As Byte() = ms.ToArray()
        Dim base64String As String = Convert.ToBase64String(imageBytes)
        Return base64String
    End Using
End Function

Public Function Base64ToImage(ByVal base64code As String) As Image
    Dim imageBytes As Byte() = Convert.FromBase64String(base64code)
    Dim ms As New MemoryStream(imageBytes, 0, imageBytes.Length)
    Dim tmpImage As Image = Image.FromStream(ms, True)
    Return tmpImage
End Function

End Class

```

LISTING PROGRAM

- Aplikasi Perbandingan Citra Digital

```

Public Class Form1
    Dim kpic1, kpic2 As Integer
    Dim box As New List(Of Bitmap)
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
        Handles MyBase.Load
            kpic1 = kpic2 = 0
            box = New List(Of Bitmap)
        End Sub
    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
        System.EventArgs) Handles PictureBox1.Click
            kpic1 = kpic1 + 1
            Dim open As New OpenFileDialog

            With open
                .CheckFileExists = True
                .ShowReadOnly = False
                .Filter = "All files|*.*| bitmap files(*)|.bmp;*.png;*.jpg"
                .FilterIndex = 2
                If .ShowDialog = DialogResult.OK Then
                    PictureBox1.Image = Image.FromFile(.FileName)
                    PictureBox1.SizeMode = PictureBoxSizeMode.Zoom
                End If
            End With
            If kpic1 > 1 Then
                box.ElementAt(0)
                box(0) = PictureBox1.Image
            Else
                box.Add(PictureBox1.Image)
            End If
        End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
        System.EventArgs) Handles PictureBox2.Click
            kpic2 = kpic2 + 1
            Dim open As New OpenFileDialog

            With open
                .CheckFileExists = True
                .ShowReadOnly = False
                .Filter = "All files|*.*| bitmap files(*)|.bmp;*.png;*.jpg"
                .FilterIndex = 2
                If .ShowDialog = DialogResult.OK Then
                    PictureBox2.Image = Image.FromFile(.FileName)
                    PictureBox2.SizeMode = PictureBoxSizeMode.Zoom
                End If
            End With
            If kpic2 > 1 Then
                box.ElementAt(0)
                box(0) = PictureBox1.Image
            Else
                box.Add(PictureBox1.Image)
            End If
        End Sub

    Private Sub Bandingkan_Click(ByVal sender As System.Object, ByVal e As
        System.EventArgs) Handles Bandingkan.Click
            If (box(0).Width = box(1).Width And box(0).Height = box(1).Height) Then
                Dim r, g, b, a As Integer
                Dim r1, r2, g1, g2, b1, b2, a1, a2 As Integer
            End If
        End Sub
    End Class

```

```

Dim perbedaan As Integer = 0
Dim gambar As New Bitmap(box(0).Width, box(0).Height)

pb.Maximum = box(0).Width * box(0).Height

For x = 0 To box(0).Width - 1
    For y = 0 To box(0).Height - 1
        Try
            With box(0).GetPixel(x, y)
                r1 = Convert.ToInt16(.R)
                g1 = Convert.ToInt16(.G)
                b1 = Convert.ToInt16(.B)
                a1 = Convert.ToInt16(.A)
            End With
            Catch ex As Exception
            End Try

            Try
                With box(1).GetPixel(x, y)
                    r2 = Convert.ToInt16(.R)
                    g2 = Convert.ToInt16(.G)
                    b2 = Convert.ToInt16(.B)
                    a2 = Convert.ToInt16(.A)
                End With
                Catch ex As Exception
                End Try

            If (r1 = r2 And g1 = g2 And b1 = b2 And a1 = a2) Then
                r = r1
                g = g1
                b = b1
                a = a1
            Else
                perbedaan = perbedaan + 1
                r = 255
                g = 0
                b = 0
                a = 255
            End If

            gambar.SetPixel(x, y, Color.FromArgb(a, r, g, b))
            pb.Value = pb.Value + 1
        Next y
    Next x

PictureBox3.Image = gambar
PictureBox3.SizeMode = PictureBoxSizeMode.Zoom

If (perbedaan > 0) Then
    MessageBox.Show("Kedua Gambar Ada Perbedaan")
Else
    MessageBox.Show("Kedua Gambar Sama")
End If
Else
    MessageBox.Show("proses tidak dapat dijalankan karena adanya perbedaan
dimensi")
End If
box.Clear()
End Sub
End Class

```