

**ENKRIPSI CITRA DIGITAL MENGGUNAKAN TEKNIK
PENGGABUNGAN *HILL CIPHER* DAN SUBSTITUSI**

SKRIPSI

Oleh:
YOSSY YANUAR WIBOWO
0810963076-96



PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2012

UNIVERSITAS BRAWIJAYA



**ENKRIPSI CITRA DIGITAL MENGGUNAKAN TEKNIK
PENGGABUNGAN HILL CIPHER DAN SUBSTITUSI**

Skripsi

Sebagai salah satu syarat untuk memperoleh gelar
Sarjana Komputer dalam bidang Ilmu Komputer

Oleh :

YOSSY YANUAR WIBOWO
0810963076-96



PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2012

UNIVERSITAS BRAWIJAYA



LEMBAR PENGESAHAN SKRIPSI

Enkripsi Citra Digital Menggunakan Teknik
Penggabungan *Hill Cipher* dan Subtitusi

Oleh :

YOSSY YANUAR WIBOWO
0810963076-96

Setelah dipertahankan di depan Majelis Penguin
pada tanggal 10 Agustus 2012

dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Komputer dalam bidang Ilmu Komputer

Pembimbing I,

Pembimbing II,

Edy Santoso., SSi., M.Kom
NIP. 197404142003121004

Dian Eka R., SSi., M.Kom.
NIP. 197306192002122001

Mengetahui,
Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya

Dr. Abdul Rouf Alghofari, M.Sc
NIP.196709071992031001

UNIVERSITAS BRAWIJAYA



LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama

: Yossy yanuar Wibowo

NIM

: 0810963076-96

Jurusan

: Matematika

Program Studi

: Ilmu Komputer

Penulis skripsi berjudul : Enkripsi Citra Digital Menggunakan Teknik
Penggabungan Hill Cipher dan Subtitusi

Dengan ini menyatakan bahwa :

1. Isi dari Skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam Skripsi ini.
2. Apabila dikemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 10 Agustus 2012

Yang menyatakan,

Yossy Yanuar Wibowo

NIM. 0810963076-96

UNIVERSITAS BRAWIJAYA



ENKRIPSI CITRA DIGITAL MENGGUNAKAN TEKNIK PENGGABUNGAN *HILL CIPHER* DAN SUBSTITUSI

ABSTRAK

Seiring dengan perkembangan zaman, kemajuan teknologi dibidang komputer dan telekomunikasi berkembang sangat pesat. Lalu lintas pengiriman data dan informasi semakin tersebar luas, ketika mengirimkan data yang sangat rahasia ada kemungkinan ditengah jalan data tersebut dibajak atau diketahui artinya oleh orang lain. Sehingga menyebabkan data menjadi tidak aman lagi.

Salah satu solusi untuk menjaga agar data tersebut tetap aman saat dilakukan pengiriman data adalah dengan teknik kriptografi. Dalam skripsi ini akan dibahas bagaimana sebuah data dalam bentuk gambar atau citra digital dilakukan enkripsi dengan kunci tertentu sebelum dikirim ke tujuan, dimana kunci tersebut hanya diketahui oleh pengirim dan penerima sehingga walaupun data tersebut dapat diakses oleh orang banyak akan tetapi yang dapat mengetahui arti dari data tersebut hanya pengirim dan penerima.

Teknik kriptografi yang digunakan adalah *Hill Chiper* dan Subtitusi, kedua teknik ini akan dikombinasikan menjadi satu. Teknik *Hill Cipher* biasa digunakan untuk enkripsi data berupa teks, teknik ini memiliki tingkat keamanan yang tinggi, sedangkan teknik Subtitusi biasa digunakan untuk enkripsi data berupa citra digital, teknik ini memiliki kecepatan enkripsi atau dekripsi tinggi. Diharapkan dengan kombinasi dari kedua teknik ini akan menghasilkan suatu teknik enkripsi citra digital baru yang memiliki tingkat keamanan dan kecepatan enkripsi atau dekripsi yang tinggi.

UNIVERSITAS BRAWIJAYA



ENCRYPTION DIGITAL IMAGE USING COMBINATION HILL CIPHER TECHNIQUE AND SUBSTITUTION

ABSTRACT

Along the times, the progress of computer and telecommunications sector was growing rapidly. Data transfer and information traffic spreading widely, when want to transfer secret data there is a possibility in the middle of the transfer is hacked or the data that is known by others. Thus causing the data to be no longer safe.

One solution to keep your data safe when data transfer is held by performe cryptographic techniques. In this paper we discuss how the data in the form of pictures or digital images encrypted with a specific key encryption before being sent to the destination, where the key is only known by the sender and recipient. Therefore, although the data can be accessed by many people but who can know the meaning of the data only the sender and receiver.

Cryptographic technique used is Hill cipher and substitution, both of these techniques will be combined into one. Hill Cipher technique commonly used in text encryption, this technique has a high level of security, while the substitution technique is used for digital image encryption, this technique has a high speed encryption or decryption. Expected with the combination of these techniques it will generate a new digital image encryption technique which have a high level of security and encryption or decryption speed.

UNIVERSITAS BRAWIJAYA



KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi. Skripsi yang berjudul **“ENKRIPSI CITRA DIGITAL MENGGUNAKAN TEKNIK PENGABUNGAN HILL CIPHER DAN SUBSTITUSI”** merupakan salah syarat memperoleh gelar Sarjana Komputer pada program studi Ilmu Komputer Jurusan Matematika Fakultas MIPA Universitas Brawijaya.

Tidak dapat dipungkiri bahwa tidak mungkin penulis dapat menyelesaikan skripsi ini tanpa bantuan dan dukungan dari banyak pihak. Untuk itu, dengan ketulusan dan kerendahan hati penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Edy Santoso., SSi., M.Kom., selaku dosen pembimbing utama yang telah meluangkan waktunya dengan sabar membimbing dan memberikan pengarahan kepada penulis.
2. Dian Eka R., SSi., M.Kom., selaku dosen pembimbing kedua yang telah membimbing dengan bijaksana dan selalu memberikan masukan kepada penulis.
3. Dr. Abdul Rouf Al-Ghofari, M.Sc., selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Brawijaya.
4. Drs. Marji, M.T., selaku Ketua Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Brawijaya.
5. Segenap Bapak dan Ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada penulis selama menempuh pendidikan di Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Brawijaya.
6. Segenap staf dan karyawan di Jurusan Matematika Fakultas MIPA Universitas Brawijaya yang telah banyak membantu penulis dalam pelaksanaan penyusunan proposal skripsi ini.
7. Kedua orang tua (bapak Sukari dan ibu lilik suprihatin) serta kakak (mas Erik), adik (dek inggit) dan seluruh keluarga yang selalu memberikan dukungan doa dan kasih sayang yang tulus kepada penulis dalam menyelesaikan skripsi ini.

8. Almarhum ibu sudarmi yang dengan sabar sudah mendidik penulis dari kecil sampai kelas 3 smp.
9. Putri oktiningsasi yang dengan sabar selalu setia menemani, mendengarkan keluhan dan selalu memberikan motivasi kepada penulis.
10. Semua teman-teman Program Studi Ilmu Komputer Fakultas MIPA Universitas Brawijaya angkatan 2008 yang telah banyak memberikan dukungan demi kelancaran pelaksanaan penyusunan skripsi ini.
11. Dan semua pihak yang telah terlibat baik secara langsung maupun tidak langsung yang tidak dapat penulis sebutkan satu per satu terima kasih atas semua bantuan yang telah diberikan.

Semoga skripsi ini bermanfaat bagi pembaca sekalian. Akhirnya, penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan dan mengandung banyak kekurangan, sehingga dengan segala kerendahan hati penulis mengharapkan kritik dan saran yang membangun dari pembaca.

Malang, 10 Agustus 2012

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN SKRIPSI	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Manfaat.....	3
1.6. Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA.....	5
2.1. Citra	5
2.1.1. Pengertian Citra.....	5
2.1.2. Warna Citra.....	6
2.1.3. Penyimpanan Citra	6
2.2. Kriptografi	7
2.2.1. Pengertian Kriptografi.....	7
2.2.2. Ciphers	8
2.2.2.1. Pengertian Ciphers	8
2.2.3. Hill cipher	9
2.2.3.1. Pengertian Hill cipher	9
2.2.3.2. Proses Enkripsi Hill Cipher.....	11
2.2.3.3. Proses Dekripsi Hill Cipher	13
2.2.3.4. Algoritma Enkripsi dan Dekripsi Hill Cipher	15
2.2.4. Subtitusi	18
2.2.4.1. Pengertian Metode Subtitusi	18
2.2.4.2. Enkripsi dan Dekripsi Metode Subtitusi	18
2.3. Metode uji coba	21
2.3.1. Pengukuran parameter waktu proses.....	21
2.3.2. Perbandingan Antara Size File Input dan File Output	21

2.3.3. Analisa Keamanan	21
2.3.4. Ketahanan Image Cipherteks	22
BAB III METODOLOGI DAN PERANCANGAN.....	23
3.1 Analisis dan Perancangan Sistem.....	24
3.1.1 Deskripsi Sistem.....	24
3.1.2 Rancangan Sistem.....	24
3.1.2.1 Enkripsi Hill cipher.....	26
3.1.2.2 Dekripsi hill cipher	27
3.1.2.3 Subtitusi	29
3.2. Perhitungan Manual	30
3.2.1. Perhitungan Parameter.....	30
3.2.2. Perhitungan Proses Enkripsi	30
3.2.3. Perhitungan Proses Dekripsi	33
3.3. Perancangan Uji Coba dan Analisis	38
3.3.1. Pengukuran parameter waktu proses	38
3.3.2. Perbandingan Antara Ukuran File Input dan File Output	39
3.3.3. Analisa Keamanan	40
3.3.4. Ketahanan Image Cipher teks	40
3.4. Perancangan Antar Muka	42
BAB IV IMPLEMENTASI DAN PEMBAHASAN.....	45
4.1. Lingkungan Implementasi.....	45
4.1.1. Lingkungan Perangkat Keras.....	45
4.1.2. Lingkungan Perangkat Lunak	45
4.2. Implementasi Program	45
4.2.1. Implementasi Enkripsi	46
4.2.1.1. Implementasi Enkripsi Hill Cipher	46
4.2.2. Implementasi Dekripsi	47
4.2.2.1. Implementasi Dekripsi <i>Hill Cipher</i>	48
4.2.3. Implementasi Subtitusi	49
4.2.4. Implementasi Menghitung Determinan	52
4.2.5. Implementasi Menghitung Koofaktor	53
4.2.6. Implementasi Inverse	53
4.2.7. Implementasi Minnor	54
4.2.8. Implementasi Tranpose	54
4.3. Implementasi Antarmuka	55
4.4. Uji Coba	59
4.4.1. Uji Coba Pengukuran Parameter Waktu Proses	59

4.4.2. Uji Coba Perbandingan Ukuran <i>File Input</i> dan <i>File Output</i>	61
4.4.3. Uji Coba Analisa Keamanan	62
4.4.4. Uji Coba Ketahanan <i>Image Cipher</i>	64
4.5. Analisa Hasil	66
4.5.1. Analisa Hasil Pengukuran Parameter Waktu Proses	67
4.5.2. Analisa Hasil Perbandingan Ukuran <i>File Input</i> dan <i>File Output</i>	67
4.5.3. Analisa Hasil Kekuatan Keamanan	68
4.5.4. Analisa Hasil Ketahanan <i>Image Cipher</i>	68
4.5.5. Analisa Hasil Keseluruhan	68
BAB V KESIMPULAN DAN SARAN	71
5.1. Kesimpulan	71
5.2. Saran	71
DAFTAR PUSTAKA	73



UNIVERSITAS BRAWIJAYA



DAFTAR GAMBAR

Gambar 2.1 Citra Digital.....	5
Gambar 2.2 Syarat Perkalian Matrik.....	9
Gambar 2.3 Nilai piksel dalam matrik	18
Gambar 2.4 Pembacaan pixel dengan spiral	19
Gambar 2.5 Deretan pixel dalam garis lurus.....	19
Gambar 2.6 Subtitusi dengan urutan pixel baru.....	20
Gambar 2.7 Hasil Subtitusi korespondensi satu-satu.....	20
Gambar 3.1 Langkah – Langkah Penelitian.....	23
Gambar 3.2 Flowchart Proses Enkripsi Citra.....	25
Gambar 3.3 Flowchart Proses Dekripsi Citra	26
Gambar 3.4 <i>Enkripsi Hill cipher</i>	27
Gambar 3.5 <i>Dekripsi hill cipher</i>	28
Gambar 3.6 Subtitusi	29
Gambar 3.7 Perancangan Antar Muka Menu Utama	42
Gambar 4.1 Antarmuka Enkripsi Citra Digital	55
Gambar 4.2 Antarmuka Open File	56
Gambar 4.3 Antarmuka Input Key	57
Gambar 4.4 Antarmuka Enkripsi dan Dekripsi	57
Gambar 4.5 Antarmuka Histogram	58
Gambar 4.6 Antarmuka Save File	58
Gambar 4.7 Antarmuka <i>File Info</i>	59

UNIVERSITAS BRAWIJAYA



DAFTAR TABEL

Tabel 3.1 Nilai RGB Image	30
Tabel 3.2 Tranpose Matrik Pixel.....	31
Tabel 3.3 Perancangan Tabel Uji Coba waktu proses.....	38
Tabel 3.4 Perancangan Tabel Uji coba Perbandingan Ukuran File Input dan output	39
Tabel 3.5 Perancangan Tabel Uji Coba Analisa Keamanan	40
Tabel 3.6 Perancangan Tabel Uji Coba Ketahanan Image Cipherteks .	41
Tabel 4.1 Tabel Uji Coba Parameter Waktu Proses.....	60
Tabel 4.2 Tabel Uji Coba Perbandingan Ukuran <i>File Input</i> dan <i>File Output</i>	61
Tabel 4.3 Tabel uji coba Analisa Keamanan	63
Tabel 4.4 Tabel uji coba Ketahanan <i>Image Cipher</i>	64



UNIVERSITAS BRAWIJAYA



DAFTAR SOURCE CODE

Source Code 4.1 Prosedur <i>Enkripsi</i>	46
Source Code 4.2 Prosedur <i>Enkripsi Hill Cipher</i>	47
Source Code 4.3 Prosedur <i>Dekripsi</i>	47
Source Code 4.4 Prosedur <i>Dekripsi Hill Cipher</i>	48
Source Code 4.5 Prosedur Subtitusi	52
Source Code 4.6 Prosedur Determinan	52
Source Code 4.7 Prosedur Koofaktor	53
Source Code 4.8 Prosedur Inverse	53
Source Code 4.9 Prosedur Minnor	54
Source Code 4.10 Prosedur Tranpose	55



UNIVERSITAS BRAWIJAYA



BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi dan informasi sekarang ini telah berkembang dengan sangat pesat, semakin banyak komputer atau peralatan telekomunikasi yang terhubung dalam dunia maya atau internet. Banyak manfaat yang dapat dilakukan melalui internet, salah satunya adalah untuk mempermudah hubungan antar lembaga. Hubungan tersebut dapat berupa penyampaian data dari pihak satu ke pihak lainnya, data tersebut bisa berupa teks, video, gambar dan lain-lain. Saat suatu data dikirim dari pihak satu ke pihak lainnya, ada kemungkinan bahwa data tersebut dapat diketahui artinya oleh pihak-pihak yang tidak diinginkan. Kriptografi dapat mengubah data pesan tersebut menjadi tidak dapat dimengerti oleh pihak-pihak yang tidak diinginkan sehingga data yang dikirimkan akan tetap aman walaupun setiap orang dapat mengaksesnya secara bebas.

Salah satu metode enkripsi untuk citra digital yaitu teknik Subtitusi, subtitusi adalah suatu teknik enkripsi simetri dimana dilakukan penggantian setiap objek *plaintext* dengan obyek lain, teknik ini menerapkan korespondensi satu-satu untuk tiap-tiap objek *plaintext* yang akan disandikan. Kelemahan teknik ini adalah tidak dapat diterapkan apabila gambar hanya memiliki satu warna saja karena metode ini tidak dapat merubah komposisi warna awal dan kelebihannya adalah apabila gambar memiliki warna yang bervariasi maka hasil enkripsinya akan menjadi semakin tidak dapat dikenali atau baik (Aryus,2008).

Salah satu teknik kriptografi adalah *Hill Cipher*, *Hill Cipher* merupakan algoritma kriptografi kunci simetris, sebenarnya metode ini lebih umum digunakan pada data yang berbentuk teks akan tetapi pada skripsi ini akan dicoba diterapkan pada gambar. Algoritma *Hill Chiper* menggunakan matrik berukuran 3×3 sebagai kunci untuk melakukan enkripsi dan dekripsi. Alasan mengapa menggunakan teknik *hill cipher* ini adalah karena bentuk image atau citra digital yang memiliki jumlah pixel banyak sehingga diperlukan metode yang sangat cepat ketika melakukan enkripsi pada setiap pixel, sehingga metode *hill cipher* sesuai dengan enkripsi yang cepat namun juga memiliki tingkat keamanan yang tinggi karena warna disetiap pixel akan dirubah atau dienkripsi berdasarkan key.Metode

Hill cipher bisa melakukan enkripsi atau dekripsi dengan tingkat keamanan yang tinggi dikarenakan dalam proses enkripsi dan dekripsinya melibatkan perkalian matriks sesuai dengan hasil penelitian yang berbentuk jurnal dengan judul KOMBINASI KRIPTOGRAFI DENGAN HILL CIPHER DAN STEGANOGRAFI DENGAN LSB UNTUK KEAMANAN DATA TEKS. (Suryani,2008).

Dalam skripsi ini akan dilakukan enkripsi dengan 2 metode yaitu, metode *hill cipher* dan Subtitusi dengan judul **ENKRIPSI PADA CITRA DIGITAL MENGGUNAKAN TEKNIK PENGGABUNGAN HILL CIPHER DAN SUBSTITUSI**.

1.2. Rumusan Masalah

Permasalahan yang dijadikan subyek penelitian dari skripsi ini adalah sebagai berikut :

1. Bagaimana mengimplementasikan teknik penggabungan antara *Hill Cipher* dan Subtitusi untuk melakukan enkripsi citra digital?
2. Bagaimana kecepatan dalam melakukan proses enkripsi atau dekripsi, perbedaan ukuran *file input* dan *output*, keamanan data, dan ketahanan hasil enkripsi?

1.3. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. Penelitian dilakukan dengan menggunakan data citra yang memiliki format .jpg, .bmp dan .png.
2. Berjalan pada *Operating System* menggunakan Windows 7.
3. *Input* yang dibutuhkan untuk sistem ini antara lain adalah data citra yang memiliki format .jpg., .bmp dan .png. selain itu kunci yang digunakan juga harus memiliki invers
4. *Output* yang dihasilkan berupa cipher image yang memiliki format .png dan .bmp.
5. Dimensi *file* gambar maksimal berukuran 5760 x 3600

1.4. Tujuan

Tujuan dari penelitian ini adalah :

1. Mengimplementasikan Teknik Penggabungan antara *Hill Cipher* dan *Subtitusi* untuk mengenkripsi citra digital.

2. Mengetahui kecepatan dalam melakukan proses enkripsi atau dekripsi, perbedaan ukuran file input dan output, keamanan data, dan ketahanan hasil enkripsi.

1.5. Manfaat

Manfaat yang diperoleh dari penelitian ini adalah sebagai berikut :

1. Memberikan keamanan data yang berbentuk citra digital, walaupun data tersebut dapat diakses oleh pihak-pihak yang tidak diinginkan.

1.6. Sistematika Penulisan

Pembuatan skripsi ini disusun berdasarkan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN

Berisi latar belakang masalah penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Menguraikan teori – teori yang erat hubungannya dengan kriptografi dengan teknik Hill.

3. BAB III METODOLOGI PERANCANGAN

Pada bab ini akan dijelaskan mengenai metode – metode yang digunakan dalam melakukan enkripsi dan dekripsi dengan menggunakan metode Hill Cipher.

4. BAB IV HASIL DAN PEMBAHASAN

Dalam bab ini akan dijelaskan mengenai implementasi program, pengujian dan analisa hasil penelitian.

5. BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dari seluruh rangkaian penelitian serta saran kemungkinan pengembangannya.

UNIVERSITAS BRAWIJAYA



BAB II

TINJAUAN PUSTAKA

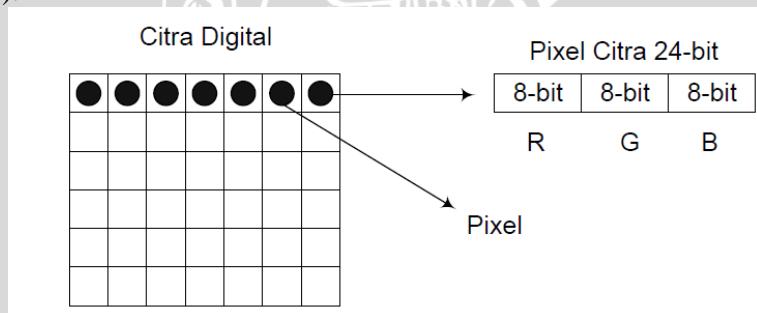
Pada bab II ini menjelaskan teori – teori yang erat hubungannya dengan enkripsi citra digital menggunakan metode hill cipher, metode hill cipher melibatkan perkalian matriks, inverse matriks, koefaktor dan adjoint untuk melakukan proses enkripsi dan dekripsi. Metode penyimpanan citra sendiri ada dua macam yaitu paint dan draw, format file citra digital yang digunakan adalah format .jpg, .png dan .bmp. Semua teori itu akan dijelaskan pada bab II .

2.1. Citra

2.1.1. Pengertian Citra

Image citra terdiri dari semua tipe data kecuali yang berkode ASCII dan tidak mempunyai properti temporal (yaitu berubah sesuai dengan waktu). (Andleigh, 1995)

Citra digital adalah representasi numerik dari objek-objek. Citra digital dibentuk oleh sekumpulan angka dalam *array* dua dimensi. Tiap angka menggambarkan warna dari tiap titik dalam gambar sesuai dengan mode warna yang digunakan. Titik-titik ini disebut *pixel* yang merupakan singkatan dari *picture element* (elemen gambar).



Gambar 2.1 Citra Digital

Gambar diatas menunjukkan citra digital yang menggunakan 24-bit warna tiap *pixel*-nya. 24-bit warna dibagi ke dalam tiga bagian sebesar 8-bit, tiap bagian merupakan representasi intensitas warna dasar yaitu merah, hijau, dan biru (RGB). (Andleigh. 1995)

2.1.2. Warna Citra

Metode untuk menampilkan sebuah citra pada layar monitor diperlukan lebih informasi mengenai letak dari *pixel-pixel* pembentuk citra. Untuk memperoleh gambaran yang tepat dibutuhkan juga informasi tentang warna-warna yang dipakai untuk menggambarkan sebuah citra digital. Beberapa mode warna yang sering digunakan:

1. *Bitmap Mode*, memerlukan 1-bit data untuk menampilkan warna dan warna yang dapat ditampilkan hanya warna hitam dan putih (monokrom)
2. *Indexed Color Mode*, mengurutkan warna dalam jangkauan 0-255 (8-bit)
3. *Grayscale Mode*, menampilkan citra dalam 256 tingkat keabuan
4. *RGB Mode*, menampilkan citra dalam kombinasi tiga warna dasar (*red*, *green*, dan *blue*), tiap warna dasar memiliki intensitas warna 0-255 (8-bit)
5. *CMYK Mode*, menampilkan citra dalam kombinasi empat warna dasar (*cyan*, *magenta*, *yellow* dan *black*), tiap warna memiliki intensitas 0-255 (8-bit). (Andleigh, 1995)

2.1.3. Penyimpanan Citra

Metode penyimpanan citra ke dalam media penyimpanan dalam bentuk digital memiliki bentuk yang beragam. Ada dua cara penyimpanan yang biasa dilakukan oleh perangkat lunak yaitu *bitmap* dan *vector*. Dalam hal ini sering juga digunakan istilah program *paint* dan program *draw*. (Jolly Shah, 2011)

Program *paint* atau program berbasis *bitmap* menyimpan citra sebagaimana ditampilkan di layar yaitu sebagai *array* dari *pixel-pixel*. Perubahan yang dilakukan pada citra dengan menggunakan program ini akan mengubah langsung tiap titik atau *pixel* pada citra. Kelebihan cara ini adalah kemudahannya untuk menampilkan gambar secara rinci dengan pola-pola yang kompleks atau gambar *fotorealistik*, yang tidak dapat dengan mudah direpresentasikan sebagai model matematika. (Jolly Shah, 2011)

Program *draw* atau program berbasis *vector* menyimpan citra sebagai model matematika, dan setiap elemen citra disimpan secara terpisah. Perubahan yang dilakukan pada citra menggunakan program ini akan mengubah deskripsi matematika yang menyusun

gambar dan program menghitung perubahan yang perlu pada warna-warna *pixel* secara tidak langsung. Kelebihan cara ini adalah kemampuannya untuk menciptakan gambar dalam resolusi yang berbeda tanpa kehilangan mutu gambar yang berarti. (Andleigh, 1995)

2.2. Kriptografi

2.2.1. Pengertian Kriptografi

Pengkajian penyandian dan penguraian (decoding) isi sandi pesan disebut kriptografi. Walaupun kode rahasia bermula sejak saat awal komunikasi tertulis, terdapat perhatian yang melonjak akhir-akhir ini terhadap masalah ini karena kebutuhan memelihara keleluasaan pribadi dari informasi yang dipancarkan melalui saluran komunikasi umum.(Lancester, 1985)

Dalam kriptografi, kode-kode disebut sandi (*cipher*), pesan yang tidak dikode disebut teks biasa (*plaintext*), dan pesan-pesan yang dikode disebut teks sandi (*ciphertext*). Proses pengubahan dari *plaintext* ke *ciphertext* disebut penyandian (enkripsi/enciphering) sedangkan kebalikannya pengubahan dari *ciphertext* menjadi *plaintext* disebut penguraian (dekripsi/deciphering). Ada banyak teknik atau metode dalam ilmu kriptografi salah satunya adalah hill cipher. .(Lancester, 1985)

Dalam pengertian luasnya, kriptografi juga mencakup penggunaan untuk menyembunyikan pesan, *cipher*, dan kode-kode. Penyembunyian pesan, seperti ini menyembunyikan teks asli seperti ditulis dengan tinta yang tidak dapat terlihat, tingkat keberhasilan bergantung pada apakah tinta yang tidak terlihat ini dicurigai atau tidak. Sekali ditemukan, maka biasanya sudah sangat mudah untuk dipecahkan. Penggunaan kode, berupa kata-kata, angka-angka, atau simbol yang mewakili huruf atau frase, biasanya tidak mungkin untuk dibaca tanpa adanya kunci dari buku kode. Kriptografi juga mencakup penggunaan komputer sebagai alat untuk memproses enkripsi dan melindungi transmisi data dan pesan. .(Lancester, 1985)

Saat ini kebanyakan sistem komunikasi meninggalkan beberapa data dan informasi yang direkam atau disimpan. Sebagai contoh, komunikasi melalui saluran telepon, termasuk faksimile dan pesan melalui e-mail, menghasilkan suatu rekaman dari nomor telepon yang dipanggil pada saat panggilan dilakukan. Transaksi finansial, data medis, pilihan pada jenis film-film yang disewa, dan

bahkan pilihan atas menu makanan yang dipesan sewaktu melakukan jamuan pada restoran mungkin dapat dilakukan pelacakan pada kartu kredit penerima atau untuk rekaman bagi pihak asuransi. Setiap waktu seorang menggunakan telepon atau kartu kredit, perusahaan telepon atau institusi finansial menyimpan suatu rekord atas nomor yang dipanggil atau jumlah transaksi, lokasi dan tanggal. Pada masa yang akan datang, jaringan telepon menjadi sistem digital, bahkan percakapan biasa dapat direkam dan disimpan. Semua data yang disimpan tersebut mungkin akan dilakukan proses pengiriman dari satu tempat ke tempat lain bagi pihak yang membutuhkan. Untuk menghindari data tersebut jatuh ke tangan yang tidak berkepentingan maka biasanya data tersebut akan dilakukan proses enkripsi untuk menjamin kerahasiaan data itu sendiri. (Lester S, 1929)

Kriptografi sangat penting tidak hanya sebagai privasi saja. Kriptografi melindung sistem perbankan dunia dengan baik. Banyak bank dan institusi finansial melakukan transaksi bisnis pada jaringan yang terbuka, seperti Internet. Tanpa adanya kemampuan untuk memproteksi transaksi bank dan komunikasi, para kriminal dapat mengganggu atau menyadap transaksi tersebut dan mencuri uang tanpa dapat dilacak. (Lester S, 1929)

2.2.2. Ciphers

2.2.2.1. Pengertian Ciphers

Secara umum *Chiper* dibedakan menjadi dua macam, yaitu *cipher substitution* dan *cipher trasposition*. *Cipher* substitusi memerlukan satu *cipher* alphabet untuk menggantikan *plaintext* dengan huruf atau simbol yang lain. *Cipher* transposisi menggunakan pergeseran huruf dalam satu kata untuk membuat kata tersebut tidak mempunyai arti. (Lester S, 1929)

Cipher adalah kode rahasia yang dipakai untuk enkripsi pesan berupa *plaintext*. *Cipher* dengan variasi tipe telah ditemukan, tetapi semuanya kebanyakan berupa *cipher* substitusi atau *cipher* transposisi. *Cipher* komputer adalah *cipher* yang dipakai pada pesan digital. *Cipher* komputer berbeda dari *cipher* substitusi atau transposisi biasa dimana pada *cipher* komputer memakai aplikasi komputer untuk melakukan enkripsi data. Istilah kriptografi kadang-kadang dibatasi pada penggunaan *cipher* atau metode yang dilibatkan untuk menggantikan huruf yang lain atau simbol pada huruf asli dari pesan. (Lester S, 1929)

2.2.3. Hill cipher

2.2.3.1. Pengertian Hill cipher

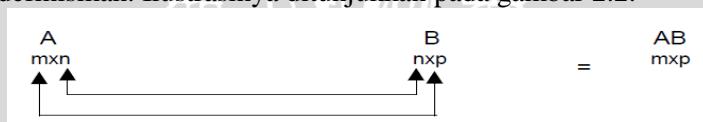
Sejak kekaisaran Romawi, kriptosistem yang lebih rumit dikembangkan oleh orang seperti oleh ahli Matematika Italia Leon Battista Alberti (lahir pada tahun 1404), Matematikawan Jerman Johannes Trithernius (lahir pada tahun 1492), seorang kriptographer dan diplomat Perancis Blaise de Vigenére (1523–1596), Lester S. Hill, yang menemukan Hill Cipher (Hill Cipher) pada tahun 1929. Hill Cipher merupakan jenis lain dari *Polygraphic cipher*. Sandi ini mengenkripsi suatu *string* huruf menjadi bentuk *string* yang lain dengan panjang yang sama. Teknik Hill Cipher dikembangkan oleh Lester S. Hill pada Hunter College dan dipublikasikan pada American Mathematical Monthly, Volume 36, Issue 6 (Juni–Juli, 1929) halaman 306 – 312. Hill Cipher menggunakan matriks untuk mentransformasi *string* berupa blok huruf. (Lester S, 1929)

Metode hill cipher saat melakukan proses enkripsi atau dekripsi selalu menggunakan perkalian matrik berikut penjelasan tentang perkalian matrik.

Perkalian matriks merupakan salah satu bentuk operasi matriks. Jika diberikan matriks $A_{m \times n}$ dan matriks $B_{n \times r}$, maka hasil kali AB didefinisikan sebagai matriks $C_{m \times r}$, yang elemen-elemnya dihitung dari A , B ditunjukkan pada persamaan 2.1.

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj} \quad i = 1, \dots, m \quad j = 1, \dots, r \quad (2.1)$$

Dua buah matriks, matriks A dan Matriks B dapat dikalikan jika dan hanya jika matriks A memiliki jumlah kolom sama dengan matriks B . Jadi $A_{m \times n} B_{n \times p}$ dapat didefinisikan, tetapi $A_{n \times m} B_{n \times p}$ tidak dapat didefinisikan. Ilustrasinya ditunjukkan pada gambar 2.2.



Gambar 2.2 Syarat Perkalian Matrik

Hasil kali dari dua matriks A dan B menghasilkan sebuah matriks B yang berukuran $m \times p$. Elemen-elemen dari matriks AB diperoleh dari hasil kali setiap baris pada matriks A dengan setiap

kolom pada matriks B kemudian dijumlahkan menjadi satu elemen. (Bellman, 1970)

Contoh perhitungan perkalian antara 2 matriks adalah sebagai berikut.

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \text{ dan } B = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, A_{2 \times 2} B_{2 \times 1} = \begin{bmatrix} (1 \times 2) + (2 \times 1) \\ (3 \times 2) + (4 \times 1) \end{bmatrix} = \begin{bmatrix} 4 \\ 10 \end{bmatrix}$$

Selain itu juga menggunakan inverse matrik terhadap keynya. Berikut adalah penjelasan tentang inverse matrik.

Diberikan matriks bujur sangkar A. Jika terdapat matriks bujur sangkar A^{-1} yang memenuhi hubungan

$$A^{-1}A = AA^{-1} = I, \quad (2.2)$$

Maka A^{-1} disebut inverse kebalikan dari A. A disebut bisa dibalik. Suatu matriks yang bisa dibalik memiliki tepat satu inverse.

Suatu matriks memiliki inverse jika determinan matriks tersebut tidak sama dengan 0. (Gantmacher, 1960)

Cara melakukan inverse pada suatu matrik dapat dilakukan dengan beberapa cara salah satunya adalah dengan cara determinan. Berikut cara mencari invers dari matriks ber ordo 3x3.

$$A^{-1} = \frac{1}{\text{Det}(A)} \text{ Adj}(A) \quad (2.3)$$

$$\text{Det } A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

$$= (A_{11} \times A_{22} \times A_{33}) + (A_{12} \times A_{23} \times A_{31}) + (A_{13} \times A_{21} \times A_{32}) - (A_{31} \times A_{22} \times A_{13}) - (A_{32} \times A_{23} \times A_{11}) - (A_{33} \times A_{21} \times A_{12})$$

Hitung kofaktor dari matriks A
Minor dari A_{32}

$$M_{32} = \begin{bmatrix} A_{11} & A_{13} \\ A_{21} & A_{23} \end{bmatrix} = \text{DetM} = (A_{11} \times A_{23}) - (A_{13} \times A_{21})$$

Maka kofaktor A_{32} adalah

$$C_{ij} = (-1)^{i+j} M_{ij}$$

$$C_{32} = (-1)^{3+2} M_{32}$$

Kemudian akan terbentuk matriks kofaktor A sebagai berikut

$$A_c = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

Untuk mencari adjoint dari matrik A dilakukan dengan cara mengganti kolom matrik A_c menjadi baris dan kolom menjadi baris, sehingga akan menjadi sebagai berikut

$$\text{Adj } A = \begin{bmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{bmatrix}$$

(Watkins, 1991)

2.2.3.2. Proses Enkripsi Hill Cipher

Berikut ini adalah proses enkripsi dengan metode hill cipher yang diterapkan pada huruf abjad.

Misalnya pesan yang digunakan adalah TRUELOVENEVERDIE dengan kunci sebagai berikut

$$K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix}$$

proses enkripsinya adalah sebagai

berikut.

a. Pesan atau plaintext yaitu TRUELOVENEVERDIE diubah ke bentuk numerik, di peroleh :

T	R	U	E	L	O	V	E	N	E	V	E	R	D	I	E
19	17	20	4	11	14	21	4	13	4	21	4	17	3	8	4

b. Karena banyaknya abjad dalam plaintext yaitu 16 bukan kelipatan dari ukuran kolom matriks kunci yaitu 3 maka tambahkan

sembarang abjad dalam plaintext sehingga k menjadi kelipatan m. Dalam Implementasi ini ditambahkan abjad Y dan Z.

- c. Buatlah plaintext dalam bentuk blok dengan ukuran blok sama dengan ukuran kolom matriks kunci yaitu 3, sehingga plaintext menjadi:

$$P = \begin{bmatrix} T & R & U \\ E & L & O \\ V & E & N \\ E & V & E \\ R & D & I \\ E & Y & Z \end{bmatrix}$$

- d. Buat P transpose

$$P^T = \begin{bmatrix} T & E & V & E & R & E \\ R & L & E & V & D & Y \\ U & O & N & E & I & Z \end{bmatrix}$$

- e. Korespondensikan hasil d dengan numerik, sehingga diperoleh :

$$P^T = \begin{bmatrix} 19 & 4 & 21 & 4 & 17 & 4 \\ 17 & 11 & 4 & 21 & 3 & 24 \\ 20 & 14 & 13 & 4 & 8 & 25 \end{bmatrix}$$

- f. Kalikan matriks kunci K dengan plaintext transpose dalam modulo 26 berikut:

$$C^T = K \cdot P^T \quad (2.4)$$

$$C^T = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix} \begin{bmatrix} 19 & 4 & 21 & 4 & 17 & 4 \\ 17 & 11 & 4 & 21 & 3 & 24 \\ 20 & 14 & 13 & 4 & 8 & 25 \end{bmatrix} \text{ mod } 26$$

$$= \begin{bmatrix} 88 & 31 & 62 & 71 & 52 & 59 \\ 181 & 112 & 84 & 142 & 59 & 223 \\ -10 & -22 & 39 & -72 & 30 & -63 \end{bmatrix} \text{ mod } 26$$

$$= \begin{bmatrix} 10 & 5 & 10 & 19 & 0 & 7 \\ 25 & 8 & 6 & 12 & 7 & 15 \\ 16 & 4 & 13 & 6 & 4 & 15 \end{bmatrix}$$

- g. Ubah hasil step f ke dalam abjad menggunakan koresponden abjad dengan numerik pada step a sehingga diperoleh ciphertext :

$$C^T = \begin{bmatrix} K & F & K & T & A & H \\ Z & I & G & M & H & P \\ 20 & E & N & G & E & P \end{bmatrix}$$

- h. Diperoleh ciphertext

$$C = (C^T)^T = \begin{bmatrix} K & Z & Q \\ F & I & E \\ K & G & N \\ T & M & G \\ A & H & E \\ H & P & P \end{bmatrix}$$

- i. Ciphertext $C = \text{KZQFIEKGNTMGAHEHPP}$

(Jonathan M. Blackledge, 2010)

2.2.3.3. Proses Dekripsi Hill Cipher

Berikut ini adalah proses enkripsi dengan metode hill cipher yang diterapkan pada huruf abjad.

Dari proses enkripsi sebelumnya diperoleh nilai kunci

$$K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix} \text{ dan ciphertext } C = \text{KZQFIEKGNTMGAHEHPP}$$

Proses dekripsinya adalah sebagai berikut.

- a. Mencari inverse matriks K, dengan cara
- Mencari adjoint matriks K

Koefaktor C_{11} sampai C_{33} adalah

$$C_{11} = (-1)^{1+1} (6 \times 1) - (-4 \times 3) = 18$$

$$C_{12} = (-1)^{1+2} (1 \times 1) - (2 \times 3) = 5$$

$$C_{13} = (-1)^{1+3} (1 \times -4) - (6 \times 2) = -16$$

$$C_{21} = (-1)^{2+1} (3 \times 1) - (-1 \times -4) = 1$$

$$C_{22} = (-1)^{2+2} (3 \times 1) - (-1 \times 2) = 5$$

$$C_{23} = (-1)^{2+3} (3 \times -4) - (3 \times 2) = 18$$

$$C_{31} = (-1)^{3+1} (3x3) - (-1x6) = 15$$

$$C_{32} = (-1)^{3+2} (3x3) - (1x-1) = -10$$

$$C_{33} = (-1)^{3+3} (3x6) - (3x1) = 15$$

Sehingga menjadi matrik koofaktor

$$C = \begin{bmatrix} 18 & 5 & -16 \\ 1 & 5 & 18 \\ 15 & -10 & 15 \end{bmatrix}$$

Adjoint matriks $K = C^T$

$$\text{Adj}(K) = \begin{bmatrix} 18 & 1 & 15 \\ 5 & 5 & -10 \\ -16 & 18 & 15 \end{bmatrix}$$

Mencari determinan matriks K dalam mod 26,

$$\begin{aligned} \text{Det } K &= (3 \times 6 \times 1) + (3 \times 3 \times 2) + (-1 \times 1 \times -4) - (2 \times 6 \times -1) - (-4 \times 3 \times 3) - (1 \times 1 \times 3) = 85 \text{ Mod } 26 = 7 \\ \text{diperoleh } 7 \end{aligned}$$

$$\frac{1}{\text{det } K} \text{ mod } 26 = x$$

$$(\text{det } K^* x) \text{ mod } 26 = 1$$

$$(7 * x) \text{ mod } 26 = 1$$

$$x = 15$$

Dicari invers matrik K :

$$K^{-1} = 15 \begin{bmatrix} 18 & 1 & 15 \\ 5 & 5 & -10 \\ -16 & 18 & 15 \end{bmatrix} \text{ mod } 26$$

$$= \begin{bmatrix} 10 & 15 & 17 \\ 23 & 23 & 6 \\ 20 & 10 & 17 \end{bmatrix}$$

b. Mencari Plaintext transpose

$$P^T = K^{-1} C^T$$

$$P^T = \begin{bmatrix} 10 & 15 & 17 \\ 23 & 23 & 6 \\ 20 & 10 & 17 \end{bmatrix} \begin{bmatrix} 10 & 5 & 10 & 19 & 0 & 7 \\ 25 & 8 & 6 & 12 & 7 & 15 \end{bmatrix} \text{ mod } 26$$

$$\begin{aligned}
 &= \begin{bmatrix} 747 & 238 & 411 & 472 & 173 & 550 \\ 901 & 323 & 446 & 749 & 185 & 596 \\ 722 & 248 & 481 & 602 & 138 & 545 \end{bmatrix} \text{ mod } 26 \\
 &= \begin{bmatrix} 19 & 4 & 21 & 4 & 17 & 4 \\ 17 & 11 & 4 & 21 & 3 & 24 \\ 20 & 14 & 13 & 4 & 8 & 25 \end{bmatrix}
 \end{aligned}$$

c. Dari langkah diatas diperoleh

$$P = (P^T)^T = \begin{bmatrix} 19 & 17 & 20 \\ 4 & 11 & 14 \\ 21 & 4 & 13 \\ 4 & 21 & 4 \\ 17 & 3 & 8 \\ 4 & 24 & 25 \end{bmatrix}$$

d. Ubah hasil step c ke dalam abjad menggunakan koresponden abjad dengan numerik sehingga diperoleh plaintext :

$$P = \begin{bmatrix} T & R & U \\ E & L & O \\ V & E & N \\ E & V & E \\ R & D & I \\ E & Y & Z \end{bmatrix}$$

e. Diperoleh plaintext **TRUELOVENEVERDIEYZ**

2.2.3.4. Algoritma Enkripsi dan Dekripsi Hill Cipher

Algoritma enkripsi dan dekripsi menggunakan teknik Hill cipher yang diterapkan pada data berupa huruf abjad secara umum tahap-tahapnya adalah sebagai berikut.

Algoritma Enkripsi Hill Cipher

1. Korespondenkan abjad dengan numerik
 $A \rightarrow 0, B \rightarrow 1, \dots, Z \rightarrow 25$
2. Buat kunci matriks berukuran $m \times m$, yang ditunjukkan dalam gambar berikut

$$K_{mxm} = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1m} \\ k_{21} & k_{22} & \dots & k_{2m} \\ \dots & \dots & \dots & \dots \\ k_{m1} & k_{m2} & \dots & k_{mm} \end{bmatrix}$$

3. Matriks K harus berupa matriks yang *invertible*, yaitu memiliki *multiplicative inverse* K^{-1} sehingga $K \cdot K^{-1} = I$
4. Matriks K mempunyai inverse jika determinan matriks $K \neq 0$
5. Jika plaintext $P = p_1 \ p_2 \ \dots \ p_n$, plaintext diblok dengan ukuran blok sama dengan ukuran baris atau kolom matriks kunci K, yaitu m seperti gambar berikut :

$$P_{qxm} = \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1m} \\ p_{21} & p_{22} & \dots & p_{2m} \\ \dots & \dots & \dots & \dots \\ p_{q1} & p_{q2} & \dots & p_{qm} \end{bmatrix}$$

Plainttext P setelah di blok menjadi ukuran $q \times m$.

6. Jika pada plaintext, n bukan kelipatan m maka tambahkan sembarang abjad dalam plaintext sehingga menjadi kelipatan m
7. Buat transpose blok matriks P :

$$P^T_{mxq} = \begin{bmatrix} p_{11} & p_{21} & \dots & p_{q1} \\ p_{12} & p_{22} & \dots & p_{q2} \\ \dots & \dots & \dots & \dots \\ p_{1m} & p_{2m} & \dots & p_{qm} \end{bmatrix}$$

8. Kalikan matriks kunci K dengan plaintext transpose dalam modulo 26 berikut:

$$C^T = K_{mxm} P_{mxn}^T$$

$$C^T = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1m} \\ k_{21} & k_{22} & \dots & k_{2m} \\ \dots & \dots & \dots & \dots \\ k_{m1} & k_{m2} & \dots & k_{mm} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1m} \\ p_{21} & p_{22} & \dots & p_{2m} \\ \dots & \dots & \dots & \dots \\ p_{q1} & p_{q2} & \dots & p_{qm} \end{bmatrix}$$

$$C^T = \begin{bmatrix} c_{11} & c_{21} & \dots & p_{m1} \\ c_{12} & c_{22} & \dots & p_{m2} \\ \dots & \dots & \dots & \dots \\ c_{1q} & c_{2q} & \dots & p_{mq} \end{bmatrix}$$

9. Dihasilkan

$$C = (C^T)^T = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1q} \\ c_{21} & c_{22} & \dots & c_{2q} \\ \dots & \dots & \dots & \dots \\ c_{m1} & c_{m2} & \dots & c_{mq} \end{bmatrix}$$

10. Ubah hasil dari langkah ke-9 ke dalam abjad menggunakan koresponden abjad dengan numerik pada step 1 sehingga diperoleh *ciphertext*.

Algoritma Dekripsi Hill Cipher

1. Korespondenkan abjad dengan numerik
 $A \rightarrow 0, B \rightarrow 1, \dots, Z \rightarrow 25$
2. Ubah ciphertext ke dalam numerik
3. Kunci yang digunakan untuk mendekripsi ciphertext ke plaintext adalah invers dari matrik kunci K_{mxm}
4. Menghitung K^{-1} dengan cara:

Mencari adjoint matriks K dan determinan K dalam mod 26

$$K^{-1} = \frac{1}{\det K} \text{adj}(K)$$

$$\frac{1}{\det K} \text{ dalam mod 26}$$

$$\text{Catatan : } \frac{1}{\det K} \text{ mod 26} = x$$

$$(\det K * x) \text{ mod 26} = 1$$

5. Kalikan inverse matriks kunci dengan ciphertext transpose, sehingga diperoleh plaintext transpose
 $P^T = K^{-1} C^T$
6. Dari langkah ke-5 diperoleh plaintext:
 $P = (P^T)^T$
7. Korespondensikan abjad dengan numerik hasil langkah ke-6 sehingga diperoleh plaintext.

(Jolly Shah, 2011)

2.2.4. Subtitusi

2.2.4.1. Pengertian Metode Subtitusi

Subtitusi adalah penggantian setiap karakter *Plaintext* dengan karakter lain. (kurniawan,2004) Dengan kata lain teknik subtitusi adalah suatu teknik enkripsi simetri dimana dilakukan penggantian setiap objek *plaintext* dengan obyek lain, teknik ini menerapkan konsep korespondensi satu-satu untuk tiap-tiap objek *plaintext* yang akan disandikan. Kemudian dalam perkembangannya, dalam metode penyandian subtitusi modern, digunakan sebuah program aplikasi tertentu dimana teks asli yang berbentuk kumpulan karakter dalam sebuah file digital diganti dengan kumpulan karakter lain secara digital sehingga menghasilkan file sandi yang siap dikomunikasikan. Terdapat empat istilah subtitusi kode, (Aryus, 2008.). antara lain :

- a. *Monoalphabetic* : setiap karakter teks-kode mengganti salah satu karakter teks-asli.
- b. *Polyalphabetic* : Setiap karakter teks-kode dapat menggantikan lebih dari satu macam karakter teks-asli.
- c. *Monograf* : satu enkripsi dilakukan terhadap satu karakter teks-asli.
- d. *Polygraph* : satu enkripsi dilakukan terhadap lebih dari satu karakter teks asli.

2.2.4.2. Enkripsi dan Dekripsi Metode Subtitusi

Langkah –langkah untuk melakukan enkripsi pada sebuah image dengan metode subtitusi adalah sebagai berikut

1. Nilai piksel-piksel dari gambar dimasukkan kedalam sebuah matrik dengan ordo yang sama dengan ukuran gambar, dapat dilihat pada gambar 2.2

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48

Gambar 2.3 Nilai piksel dalam matrik

2. Dari matrik tersebut dapat dilakukan pembacaan piksel gambar dengan aturan sepiral dimulai dari pojok kiri atas. Untuk proses pembacaan secara spiral dapat dilihat pada gambar 2.3

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	8	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48

Gambar 2.4 Pembacaan pixel dengan spiral

3. Hasil pembacaan piksel matrik di atas dibuat menjadi deretan nilai piksel dalam bentuk garis lurus, dapat dilihat pada gambar 2.4

1	2	3	4	5	6	7	8
							16
44	45	46	47	48	40	32	24
43							
42	41	33	25	17	9	10	11
							12
37	38	39	31	23	15	14	13
36							
35	34	26	18	19	20	21	22
							30
							27
							28
							29

Gambar 2.5 Deretan pixel dalam garis lurus

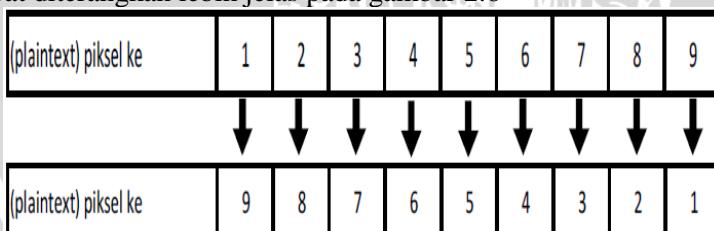
4. Dari hasil pembacaan yang terbentuk, lakukan posisi piksel secara urut. Posisi pixel akhir digantikan oleh pixel pertama sehingga

didapat dalam bentuk urutan nilai pixel baru sebagai ciphertext, lihat gambar 2.5

27	28	29	30	22	21	20	19		18	
31	39	38	37	36	35	34	26			
23										
15	14	13	12	11	10	9	17		25	
47	46	45	44	43	42	41	33			
48										
40	32	24	16	8	7	6	5		4	
								1	2	3

Gambar 2.6 Subtitusi dengan urutan pixel baru

Jadi hubungan antara piksel ciphertext dan piksel penggantinya merupakan hasil substitusi korespondensi satu-satu. Dapat diterangkan lebih jelas pada gambar 2.6



Gambar 2.7 Hasil Subtitusi korespondensi satu-satu

Sedangkan untuk proses dekripsi sama dengan proses enkripsi, tetapi inputan imagennya adalah ciphertext.

2.3. Metode uji coba

2.3.1. Pengukuran parameter waktu proses

Waktu proses yang diperlukan dalam proses enkripsi/deskripsi merupakan faktor yang sangat penting dalam pemilihan algoritma yang tepat untuk mengamankan data khususnya data multimedia. Pengujian waktu proses dilakukan dengan menghitung lamanya waktu proses enkripsi/deskripsi sehingga diketahui lamanya waktu untuk proses tersebut.

Rata-rata waktu proses didapat dari seluruh data waktu proses yang telah diujicobakan. Untuk mendapatkan rata waktu ditunjukkan pada persamaan 2.5.

$$Rata\ waktu = \frac{1}{n} \sum_{i=0}^n \text{waktu} \quad (2.5)$$

dimana n merupakan jumlah percobaan yang dilakukan.

2.3.2. Perbandingan Antara Size File Input dan File Output

Pengukuran besar citra input dilakukan pada saat pertama kali file di-load ke dalam memori. Sedangkan pengukuran besar file output dilakukan setelah penulisan pada citra selesai. Rata-rata besar file input dan output didapat dari seluruh data ukuran file input/output yang telah diujicobakan. Untuk mendapatkan rata-rata waktu proses ditunjukkan pada persamaan 2.6.

$$rata\ size\ file = \frac{1}{n} \sum_{i=0}^n \text{size\ citra} \quad (2.6)$$

dimana n merupakan jumlah percobaan yang dilakukan.

2.3.3. Analisa Keamanan

Pengukuran keamanan dilakukan dengan mengukur seberapa besar akibat apabila key yang dimasukkan memiliki selisih nilai yang kecil dari key yang benar saat melakukan proses dekripsi ditunjukkan pada persamaan 2.7.

$$selisih\ key = \begin{bmatrix} B & B & B \\ B & B & B \\ B & B & B \end{bmatrix} - \begin{bmatrix} S & S & S \\ S & S & S \\ S & S & S \end{bmatrix} \quad (2.7)$$

Total dari pengurangan matrik inilah yang dimaksud selisih key.

2.3.4. Ketahanan Image Cipherteks

Perubahan bentuk terhadap image yang telah dilakukan proses enkripsi sangat mungkin terjadi dikarenakan image hasil enkripsi tersebut dapat diakses oleh orang banyak, sehingga apakah metode Hill cipher dan Subtitusi bisa melakukan proses dekripsi dengan baik apabila image hasil enkripsi atau image ciphertext tersebut dirubah baik secara resize, cropping dan lain-lain.



BAB III

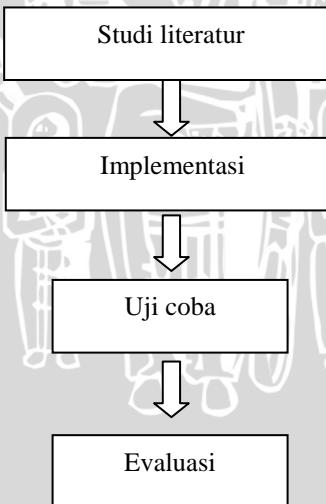
METODOLOGI DAN PERANCANGAN

Pembahasan pada bab ini meliputi metode dan langkah – langkah perancangan yang dilakukan dalam penelitian untuk proses enkripsi dan dekripsi pada citra menggunakan teknik *Hill Cipher* dan *Subtitusi*.

Langkah – langkah yang dilakukan dalam penelitian ini meliputi :

1. Melakukan studi literatur mengenai *Kriptografi* dengan teknik *Hill Cipher* dan *Subtitusi*.
2. Mengimplementasikan rancangan yang dilakukan pada tahap sebelumnya menjadi sebuah perangkat lunak.
3. Melakukan uji coba terhadap perangkat lunak menggunakan data Citra memiliki format .jpg, .png dan .bmp.
4. Mengevaluasi output hasil analisa dari sistem.

Langkah – langkah penelitian ini digambarkan pada Gambar 3.1.



Gambar 3.1 Langkah – Langkah Penelitian

3.1 Analisis dan Perancangan Sistem

3.1.1 Deskripsi Sistem

Sistem yang akan dibuat merupakan sistem yang dikembangkan untuk melakukan proses enkripsi dan dekripsi citra sehingga hanya orang-orang tertentu saja yang dapat mengenalinya. Input data yang dibutuhkan adalah citra yang memiliki format .jpg, .bmp atau .png dan sebuah chiper key atau sandi yang akan digunakan untuk merahasiakan pesan. Ketika melakukan proses enkripsi terhadap citra dibutuhkan chiper key kemudian *output* yang dihasilkan adalah suatu citra digital yang telah terenkripsi atau sudah dirahasiakan sehingga tidak dapat dikenali lagi atau biasa dinamakan Image Chipertext, untuk proses dekripsi diperlukan input berupa Image Chipertext dan chiper key sehingga nantinya akan menghasilkan *Output* berupa data semula (citra awal) atau image plaintext apabila chiper key yang dimasukkan benar dan sebaliknya jika chiper key yang dimasukkan salah maka akan menghasilkan data yang salah.

3.1.2 Rancangan Sistem

Rancangan sistem menjelaskan bagaimana sistem ini akan dibuat. Rancangan sistem terbagi menjadi 2 yaitu ketika melakukan enkripsi dan dekripsi

Proses enkripsi citra adalah dengan tahap sebagai berikut.

1. Masukkan image yang memiliki format .jpg, .bmp atau .png dan key berbentuk matrik ber ordo 3x3.
2. Jika matrik key memiliki invers maka dapat dilakukan proses enkripsi, sebaliknya jika tidak maka masukkan kembali matrik key.
3. Mengambil nilai RGB dan dimasukkan kedalam array 2 dimensi
4. Lakukan subproses enkripsi hill cipher.
5. Lakukan subproses Subtitusi
6. Output berbentuk image yang sudah di enkripsi yang biasa disebut image ciphertext.

Flowchart proses enkripsi citra dapat digambarkan seperti pada Gambar 3.2.



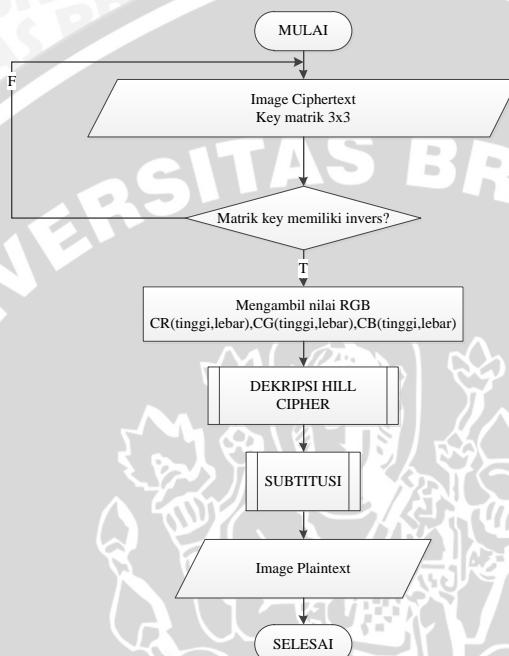
Gambar 3.2 Flowchart Proses Enkripsi Citra

Setelah dilakukan proses enkripsi untuk mengetahui pesan rahasia tersebut maka dilakukan proses dekripsi dengan tahap sebagai berikut :

1. Masukkan image hasil proses enkripsi yang memiliki format .bmp atau .png. dan key yang berbentuk matrik 3x3.
2. Jika matrik key memiliki invers maka dapat dilakukan proses dekripsi, sebaliknya jika tidak maka masukkan kembali matrik key.
3. Mengambil nilai RGB dan dimasukkan kedalam array 2 dimensi
4. Lakukan subproses dekripsi hill cipher
5. Lakukan subproses Subtitusi.
6. Jika key yang dimasukkan benar maka akan menghasilkan image plaintext atau image semula.

7. Output dapat berbentuk image plaintext

Flowchart proses dekripsi citra dapat digambarkan seperti pada Gambar 3.3.



Gambar 3.3 Flowchart Proses Dekripsi Citra

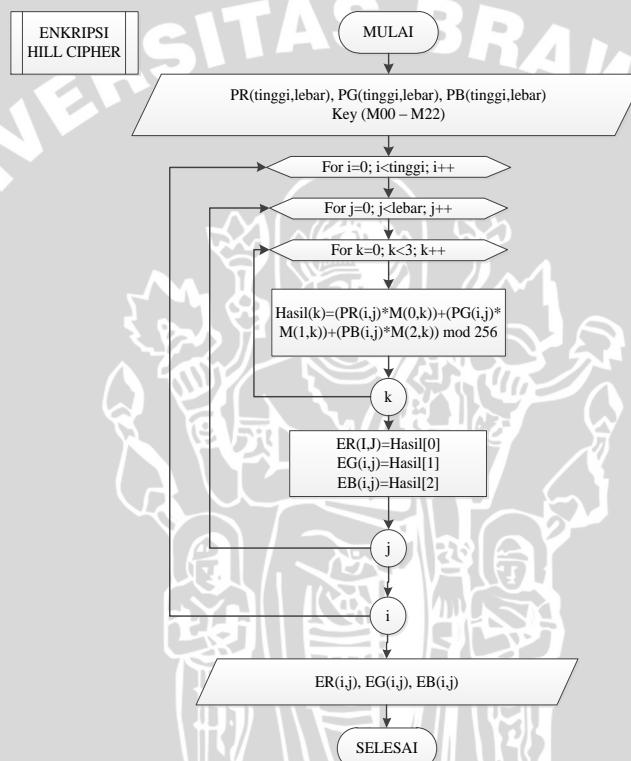
3.1.2.1 Enkripsi Hill cipher

Enkripsi hill cipher merupakan sub proses dari proses Enkripsi citra. *Enkripsi hill cipher* ini digunakan untuk menghasilkan RGB baru yang dirahasiakan. Tahapan untuk *enkripsi hill cipher* adalah sebagai berikut:

1. Inputan yang digunakan adalah nilai RGB dari tiap-tiap pixel suatu citra dan matriks key.
2. Bentuk nilai RGB tersebut menjadi matrik berordo 3x1.
3. Setiap matriks RGB (3x1) kemudian akan dikalikan dengan matriks key (3x3).
4. Setelah itu menghasilkan sebuah matriks (3x1) baru.

5. Modulus matriks baru tersebut dengan 256, sehingga menghasilkan rentang nilai antara 0 sampai 255.
6. Matriks RGB baru inilah yang merupakan cipher dari pixel awal.
7. Hasil akhir dari proses ini adalah didapatkan nilai RGB baru dari tiap-tiap pixel pada sebuah citra.

Flowchart proses *enkripsi hill cipher* dapat digambarkan seperti pada Gambar 3.4.



Gambar 3.4 *Enkripsi Hill cipher*

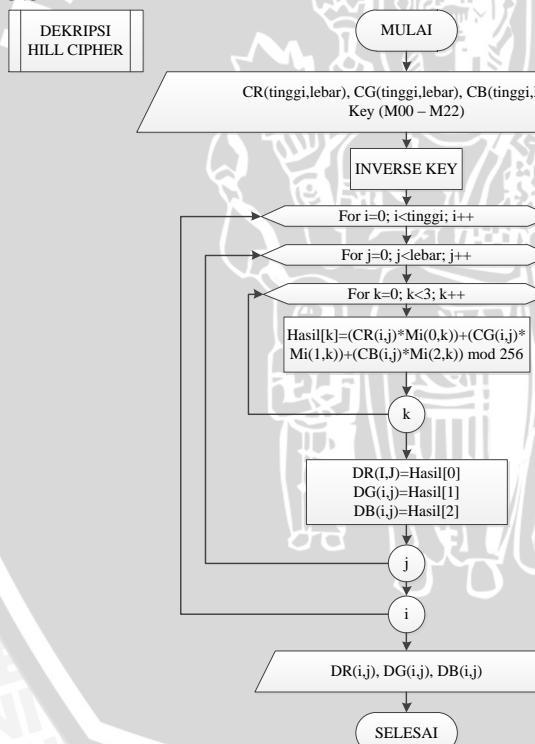
3.1.2.2 Dekripsi hill cipher

Dekripsi hill cipher merupakan sub proses dari proses Dekripsi citra. *Dekripsi hill cipher* ini digunakan untuk mengembalikan warna gambar yang telah dirahasiakan. Tahapan untuk *dekiripsi hill cipher* adalah sebagai berikut:

1. Inputan yang digunakan adalah nilai RGB dari tiap-tiap pixel dari image cipherteks dan matrik key.
2. Bentuk nilai RGB tersebut menjadi matrik berordo 3x1
3. Inverse matrik key.
4. Kalikan inverse matrik key dengan matriks RGB
5. Setelah itu menghasilkan matrik berordo 3x1 baru.
6. Modulus matrik baru tersebut dengan 256, sehingga menghasilkan rentang nilai antara 0 sampai 255.
7. Menghasilkan Matrik RGB yang merupakan Matrik RGB awal.
8. Hasil akhir dari proses ini adalah didapatkan nilai RGB dari tiap-tiap pixel yang merupakan nilai **pixel dari image awal atau image plaintext apabila matrik key yang dimasukkan benar**.

Flowchart proses *Dekripsi hill cipher* dapat ditunjukkan pada gambar

3.5



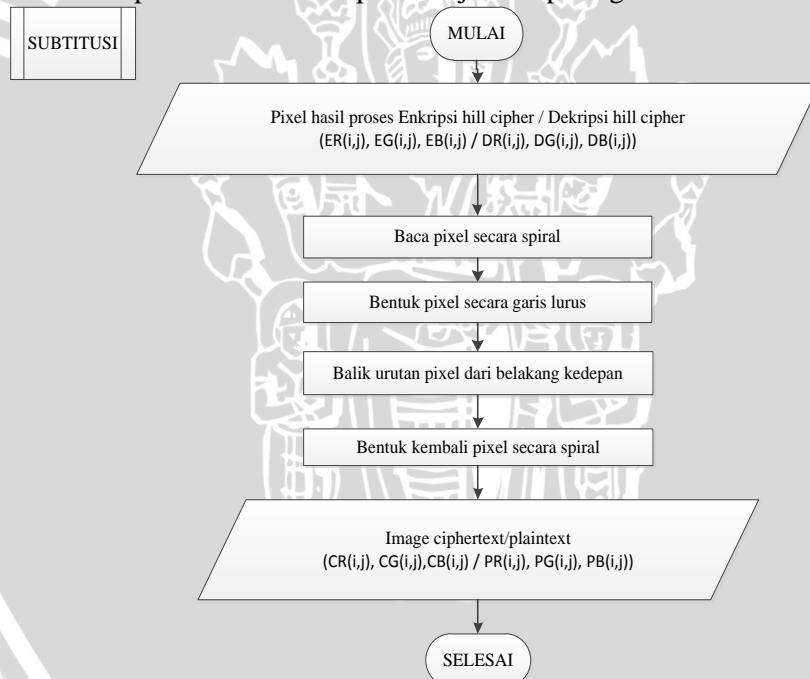
Gambar 3.5 Dekripsi hill cipher

3.1.2.3 Subtitusi

Subtitusi merupakan subproses dari proses dekripsi dan enkripsi, *subtitusi* berfungsi untuk merubah posisi pixel secara spiral sehingga akan mempersulit proses identifikasi image. Tahapan untuk *Subtitusi* adalah sebagai berikut:

1. Nilai tiap-tiap pixel hasil dari proses enkripsi atau dekripsi sebelumnya dibaca secara spiral dimulai dari pojok kiri atas. Kemudian dijadikan deretan nilai piksel dalam bentuk lurus.
2. Balik urutan piksel tadi dari belakang ke depan
3. Bentuk kembali pixel-pixel ciphertext dalam bentuk image, dimulai dari pojok kiri atas
4. Dihasilkan urutan pixel baru sehingga mempersulit proses identifikasi

Flowchart proses Subtitusi dapat ditunjukkan pada gambar 3.6



Gambar 3.6 Subtitusi

3.2. Perhitungan Manual

3.2.1. Perhitungan Parameter

Parameter masukan yang digunakan adalah matrik K berordo 3x3 yang merupakan key, Sebagai berikut.

$$K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix}$$

Setelah key diketahui kemudian akan diambil nilai RGB dari sebuah image yang dimasukkan. Diketahui nilai RGB dari image berukuran 4 x 3. ditunjukkan pada tabel 3.1.

Tabel 3.1 Nilai RGB Image

Pixel ke-	RGB		
	R	G	B
1	100	120	70
2	130	40	90
3	45	180	200
4	95	185	66
5	35	45	100
6	45	190	20
7	100	30	53
8	190	50	53
9	250	69	59
10	230	63	10
11	190	100	49
12	200	48	90

3.2.2. Perhitungan Proses Enkripsi

Dari 12 pixel image yang ada dan 1 buah matrks key berordo 3x3 yang telah diketahui, kemudian mulai melakukan proses enkripsi dengan teknik hill cipher untuk membentuk 12 pixel baru yang dirahasiakan.

Langkah- langkah perhitungan untuk algoritma *Hill Cipher* sesuai dengan flowchart pada gambar 3.2 adalah sebagai berikut :

Langkah 1. Melakukan transpose pada setiap matrik pixel image, ditunjukkan pada tabel 3.2.

Tabel 3.2 Tranpose Matrik Pixel

RGB	Pixel ke-											
	1	2	3	4	5	6	7	8	9	10	11	12
R	100	130	45	95	35	45	100	190	250	230	190	200
G	120	40	180	185	45	190	30	50	69	63	100	48
B	70	90	200	66	100	20	53	53	59	10	49	90

Langkah 2. Melakukan perkalian matrik antara matrik key dengan matrik pixel dalam modulo 256. Perkalian dilakukan mulai dari pixel ke 1 sampai pixel ke 12, sebagai berikut.

$$C^T = K P^T$$

Pixel ke 1

$$C^T = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix} \begin{bmatrix} 100 \\ 120 \\ 70 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 590 \\ 1030 \\ -210 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 78 \\ 6 \\ 46 \end{bmatrix}$$

Pixel ke 2

$$C^T = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix} \begin{bmatrix} 130 \\ 40 \\ 90 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 420 \\ 640 \\ 190 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 164 \\ 128 \\ 190 \end{bmatrix}$$

Pixel ke 3

$$C^T = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix} \begin{bmatrix} 45 \\ 180 \\ 200 \end{bmatrix}$$

$$= \begin{bmatrix} 475 \\ 1725 \\ -430 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 219 \\ 189 \\ 82 \end{bmatrix}$$

Diulangi lagi sampai pixel ke 12 sehingga matrik pixel dengan nilai RGB baru.

RGB	Pixel ke-				
	1	2	3	...	12
R	78	164	219
G	6	128	189
B	46	190	82

Langkah 3. Kemudian lakukan tranpose pada matrik C^T , sehingga akan terbentuk matrik C. Matrik C adalah pixel baru yang memiliki nilai RGB yang berbeda dari RGB pixel sebelumnya atau cipher.

Pixel ke-	RGB		
	R	G	B
1	78	6	46
2	164	128	190
3	219	189	82
...
...
12

Langkah 4. Lakukan metode subtitusi dengan cara membaca pixel image baru tadi dimulai dari pojok kiri atas secara spiral.

1	2	3
4	5	6
7	8	9
10	11	12

Langkah 5. Hasil pembacaan pixel matrik dibuat menjadi deretan nilai piksel dalam bentuk garis lurus

1	2	3	6	9	12	11	10	7	4	5	8
---	---	---	---	---	----	----	----	---	---	---	---

Langkah 6. Dari hasil pembacaan yang terbentuk, lakukan perubahan posisi piksel. Posisi pixel akhir digantikan oleh pixel pertama sehingga didapat dalam bentuk urutan nilai pixel baru sebagai ciphertext.

8	5	4	7	10	11	12	9	6	3	2	1
---	---	---	---	----	----	----	---	---	---	---	---

Langkah 7. Bentuk kembali matrik ciphertext tadi ke dalam bentuk matrik image, dengan cara spiral dimulai dari pojok kiri atas

8	5	4
3	2	7
6	1	10
9	12	11

3.2.3. Perhitungan Proses Dekripsi

Setelah dilakukan proses enkripsi terhadap image, maka akan dihasilkan image ciphertexts. Untuk mengembalikan image ciphertexts menjadi image plainteks atau menjadi semula, maka akan dilakukan proses *dekripsi*. Pada saat proses dekripsi harus memasukkan key yang benar agar menghasilkan image plainteks yang benar.

Tahap - tahap perhitungan manual proses Dekripsi adalah sebagai berikut :

Langkah 1. Melakukan transpose terhadap RGB yang terdapat di setiap pixel cipher image

RGB	Pixel ke-					
	1	2	3	...	12	
R	78	164	219	
G	6	128	189	
B	46	190	82	

Langkah 2. Melakukan invers terhadap matrik key, berikut adalah perhitungan invers matrik key

$$K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix}$$

- Mencari adjoint matriks K

Koofaktor C11 sampai C33 adalah

$$C11 = (-1)^{1+1} (6 \times 1) - (-4 \times 3) = 18$$

$$C12 = (-1)^{1+2} (1 \times 1) - (2 \times 3) = 5$$

$$C13 = (-1)^{1+3} (1 \times -4) - (6 \times 2) = -16$$

$$C21 = (-1)^{2+1} (3 \times 1) - (-1 \times -4) = 1$$

$$C22 = (-1)^{2+2} (3 \times 1) - (-1 \times 2) = 5$$

$$C23 = (-1)^{2+3} (3 \times -4) - (3 \times 2) = 18$$

$$C31 = (-1)^{3+1} (3 \times 3) - (-1 \times 6) = 15$$

$$C32 = (-1)^{3+2} (3 \times 3) - (1 \times -1) = -10$$

$$C33 = (-1)^{3+3} (3 \times 6) - (3 \times 1) = 15$$

Sehingga menjadi matrik koofaktor

$$C = \begin{bmatrix} 18 & 5 & -16 \\ 1 & 5 & 18 \\ 15 & -10 & 15 \end{bmatrix}$$

Adjoint matriks K = C^T

$$\text{Adj (K)} = \begin{bmatrix} 18 & 1 & 15 \\ 5 & 5 & -10 \\ -16 & 18 & 15 \end{bmatrix}$$

Mencari determinan matriks K dalam mod 256,

$$\text{Det K} = (3 \times 6 \times 1) + (3 \times 3 \times 2) + (-1 \times 1 \times -4) - (2 \times 6 \times -1) - (-4 \times 3 \times 3) - (1 \times 1 \times 3) = 85 \text{ Mod } 256 = 85$$

diperoleh 85

$$\frac{1}{\det K} \bmod 256 = x$$

$$(\det K^* x) \bmod 256 = 1$$

$$(85 * x) \bmod 256 = 1$$

$$x = 253$$

Dicari invers matrik K:

$$K^{-1} = 253 \begin{bmatrix} 18 & 1 & 15 \\ 5 & 5 & -10 \\ -16 & 18 & 15 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 4554 & 253 & 3795 \\ 1265 & 1265 & -2530 \\ -4048 & 4554 & 3795 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 202 & 253 & 211 \\ 241 & 241 & 30 \\ 48 & 202 & 211 \end{bmatrix}$$

Langkah 3. Melakukan perkalian invers matrik key dengan transpose matrik cipherteks

$$P^T = K^{-1} C^T$$

Pixel ke 1

$$P^T = \begin{bmatrix} 202 & 253 & 211 \\ 241 & 241 & 30 \\ 48 & 202 & 211 \end{bmatrix} \begin{bmatrix} 78 \\ 6 \\ 46 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 26980 \\ 21624 \\ 14662 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 100 \\ 120 \\ 70 \end{bmatrix}$$

Pixel ke 2

$$P^T = \begin{bmatrix} 202 & 253 & 211 \\ 241 & 241 & 30 \\ 48 & 202 & 211 \end{bmatrix} \begin{bmatrix} 164 \\ 128 \\ 190 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 105602 \\ 76072 \\ 73818 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 130 \\ 40 \\ 90 \end{bmatrix}$$

Pixel ke 3

$$P^T = \begin{bmatrix} 202 & 253 & 211 \\ 241 & 241 & 30 \\ 48 & 202 & 211 \end{bmatrix} \begin{bmatrix} 219 \\ 189 \\ 82 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 109357 \\ 100788 \\ 65992 \end{bmatrix} \bmod 256$$

$$= \begin{bmatrix} 45 \\ 180 \\ 200 \end{bmatrix}$$

Diulangi lagi sampai pixel ke 12 sehingga matrik pixel dengan nilai RGB semula terbentuk.

RGB	Pixel ke-											
	8	5	4	3	2	7	6	1	10	9	12	11
R	190	35	95	45	130	100	45	100	230	250	200	190
G	50	45	185	180	40	30	190	120	63	69	48	100
B	53	100	66	200	90	53	20	70	10	59	90	49

Langkah 4. Kemudian lakukan transpose pada matrik P^T , sehingga akan terbentuk matrik P. Matrik P adalah pixel baru yang memiliki nilai RGB sesuai dengan RGB image semula atau plainteks image.

Pixel ke-	RGB		
	R	G	B
8	190	50	53
5	35	45	100
4	95	185	66
3	45	180	200
2	130	40	90
7	100	30	53
6	45	190	20
1	100	120	70
10	230	63	10
9	250	69	59
12	200	48	90
11	190	100	49

Langkah 5. Lakukan metode subtitusi dengan cara membaca pixel cipher image dimulai dari pojok kiri atas secara spiral

8	5	4
3	2	7
6	1	10
9	12	11

Langkah 6. Hasil pembacaan pixel matrik dibuat menjadi deretan nilai pixel dalam bentuk garis lurus

8	5	4	7	10	11	12	9	6	3	2	1
---	---	---	---	----	----	----	---	---	---	---	---

Langkah 7. Dari hasil pembacaan yang terbentuk, lakukan posisi piksel secara urut. Posisi piksel akhir digantikan oleh pixel pertama sehingga didapat dalam bentuk urutan nilai pixel baru sebagai plaintext

1	2	3	6	9	12	11	10	7	4	5	8
---	---	---	---	---	----	----	----	---	---	---	---

Langkah 8. Bentuk kembali matrik plaintext tadi ke dalam bentuk matrik image, dengan cara spiral dimulai dari pojok kiri atas

1	2	3
4	5	6
7	8	9
10	11	12

3.3. Perancangan Uji Coba dan Analisis

Pada sub bab ini akan dilakukan perancangan pengujian sistem enkripsi citra digital menggunakan teknik hill cipher dan subtitusi, baik pengujian dari sistem maupun *output* dari sistem yang telah dibuat. Pengujian terhadap sistem dibagi menjadi empat, yaitu pengujian waktu proses, perbandingan ukuran file input dengan output (setelah di lakukan enkripsi atau dekripsi), analisa keamanan dan yang terakhir perubahan bentuk pada image cipherteks.

3.3.1. Pengukuran parameter waktu proses

Waktu proses yang diperlukan dalam proses enkripsi/deskripsi merupakan faktor yang sangat penting dalam pemilihan algoritma yang tepat untuk mengamankan data khususnya data multimedia. Pengujian waktu proses dilakukan dengan menghitung lamanya waktu proses enkripsi/deskripsi sehingga diketahui lamanya waktu untuk proses tersebut.

Perkembangan laju pembelajaran pada setiap percobaan dicatat untuk mengetahui rata-rata waktu proses enkripsi dan dekripsi. Tabel untuk mengetahui waktu proses dibuat seperti tabel 3.3

Tabel 3.3 Perancangan Tabel Uji Coba waktu proses

Nama File	Ukuran (KB) / Dimensi	Gambar	Waktu Proses (s)	
			Enkripsi	Dekripsi

Keterangan tabel 3.3

- Kolom nama file menunjukkan nama file yang akan di uji coba.
- Kolom gambar menampilkan gambar yang akan dilakukan proses enkripsi kemudian dekripsi.
- Waktu proses dibedakan menjadi 2 yaitu saat melakukan proses enkripsi dan proses dekripsi

Dengan menggunakan tabel 3.3 akan diketahui waktu proses enkripsi dan dekripsi pada setiap gambar yang diuji.

3.3.2. Perbandingan Antara Ukuran File Input dan File Output

Perbedaan besar file input dan output saat setelah dilakukan proses enkripsi atau dekripsi mempengaruhi pada pemakaian memori. Pada kasus file berukuran besar seperti file multimedia dengan lingkungan hardware yang terbatas, faktor ini juga penting dalam hal pengamanan data.

Tabel untuk mengetahui perbandingan antara size file input dan output dibuat seperti tabel 3.4

Tabel 3.4 Perancangan Tabel Uji coba Perbandingan Ukuran File Input dan output

Nama File		Gambar		Ukuran File (kB)	
Input / Dimensi	Output/ Dimensi	Input	Output	Input	Output

Keterangan tabel 3.4 :

- Kolom nama file menampilkan nama file yang di uji
- Kolom gambar menampilkan gambar input dan output
- Kolom ukuran file menunjukkan ukuran file sebelum diproses/input dan ukuran file setelah diproses/output

Dengan menggunakan tabel 3.4 akan diketahui perbedaan size file, sebelum/input dan setelah/output.

3.3.3. Analisa Keamanan

Pengukuran keamanan dilakukan dengan mengukur seberapa besar kemiripan hasil dekripsi dengan gambar awal, dengan cara key yang dimasukkan memiliki selisih nilai yang kecil dari key yang benar.

Berikut ini adalah perancangan tabel uji coba Analisa keamanan.

Tabel 3.5 Perancangan Tabel Uji Coba Analisa Keamanan

Gambar Awal	Gambar Enkripsi	Selisih key	Hasil	Kesimpulan

Keterangan tabel 3.5 :

- Kolom gambar awal menunjukkan gambar sebelum dilakukan enkripsi
- Kolom gambar enkripsi menunjukkan gambar hasil proses enkripsi
- Kolom selisih key menampilkan total selisih key yaitu matrik key yang benar dikurangi matrik key yang salah kemudian hasil pengurangan tersebut dijumlahkan.
- Kolom Hasil menampilkan gambar yang telah dilakukan dekripsi dengan key yang salah
- Kolom kesimpulan memberikan kesimpulan apakah hasil dekripsi dapat dikenali atau tidak.

Dengan melihat tabel 3.5 Dapat diketahui seberapa kuat keamanan yang dimiliki apabila key yang dimasukkan salah.

3.3.4. Ketahanan Image Cipher teks

Perubahan bentuk terhadap image yang telah dilakukan proses enkripsi sangat mungkin terjadi dikarenakan image hasil enkripsi tersebut dapat diakses oleh orang banyak, sehingga apakah metode Hill cipher dan Subtitusi bisa melakukan proses dekripsi

dengan baik apabila image hasil enkripsi atau image ciphertext tersebut dirubah baik secara scaling, cropping dan lain-lain.

Berikut ini adalah perancangan tabel uji coba Perubahan bentuk Pada Image Cipherteks.

Tabel 3.6 Perancangan Tabel Uji Coba Ketahanan Image Cipherteks

Image Cipherteks	Metode Perubahan	Hasil	Kesimpulan
	Rotasi (90 ⁰)		
	Rotasi (180 ⁰)		
	Rotasi (270 ⁰)		
	Diperbesar(200%)		
	Diperbesar(300%)		
	Diperbesar(150%)		
	Diperkecil(75%)		
	Diperkecil(50%)		
	Diperkecil(25%)		
	Noise Despeckle		
	Gaussian Blur		
	Pixelate Facet		
	Texture Mosaic		
	Crop atas		
	Crop bawah		
	Crop kiri		
	Crop kanan		

Keterangan tabel 3.6

- Nama file menunjukkan nama file yang akan diujicobakan
- Metode perubahan merupakan metode apa yang digunakan untuk merubah image cipherteks, yaitu secara cropping, scaling dan lain-lain.
- Kolom hasil menunjukkan hasil dari proses dekripsi apakah bisa menghasilkan image plainteks dengan baik
- Kolom kesimpulan memberikan kesimpulan apakah hasil dekripsi dapat dikenali atau tidak

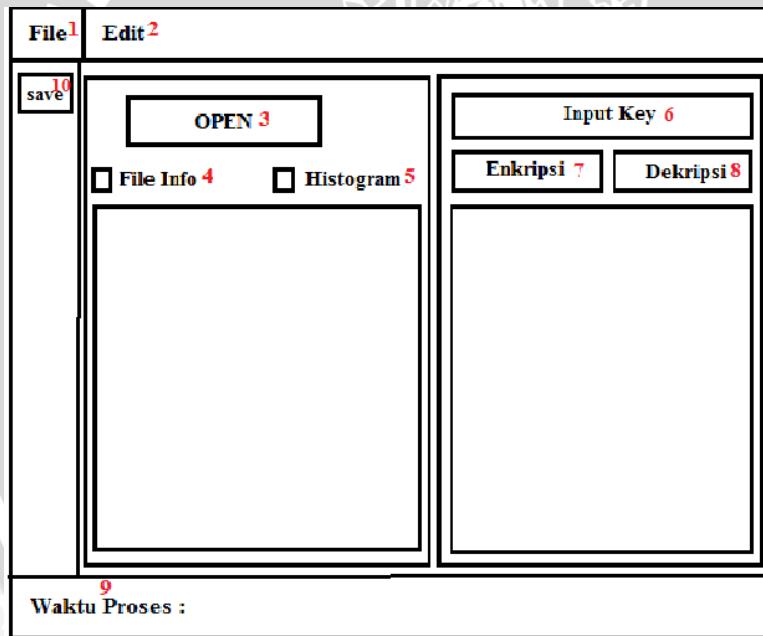
Dengan menggunakan tabel 3.7 akan diketahui apakah metode hill cipher dan subtitusi dapat melakukan dekripsi dengan baik terhadap image yang sebelumnya telah dilakukan perubahan dengan metode - metode tertentu, seperti cropping, resize dan lain-lain..

3.4. Perancangan Antar Muka

Antarmuka (*interface*) untuk sistem ini terdiri dari satu bagian utama, yaitu:

1. Bagian Utama

Bagian ini bertujuan untuk melakukan input parameter, yaitu input image dan input key selain itu juga dapat melakukan proses enkripsi dan dekripsi. Antarmuka dari bagian ini ditunjukkan pada gambar 3.7.



Gambar 3.7 Perancangan Antar Muka Menu Utama

Pada gambar 3.7, antar muka bagian proses pelatihan terdiri dari

:

1. *ToolStripMenuItems* File berisikan dua sub menu yaitu Open untuk memasukkan image dan save untuk menyimpan gambar.
2. *ToolStripMenuItems* Edit berisikan tiga sub menu yaitu Input key untuk memasukkan key yang berbentuk matriks berordo 3x3, Enkripsi untuk melakukan proses enkripsi dan Dekripsi untuk melakukan proses dekripsi.
3. *Button* Open Image untuk mencari file gambar dan membukanya.
4. *ComboBox* file Info untuk menampilkan info file yang sudah dilakukan proses dan sebelum dilakukan proses .
5. *ComboBox* Histogram, untuk menampilkan Histogram gambar.
6. *Button* Input key, untuk memasukkan key yang berbentuk matrik berordo 3x3.
7. *Button* Enkripsi untuk melakukan proses enkripsi.
8. *Button* Dekripsi untuk melakukan proses dekripsi.
9. *Waktu proses* untuk menampilkan waktu yang diperlukan saat melakukan proses enkripsi atau dekripsi
10. *Save* untuk menyimpan gambar hasil proses enkripsi atau dekripsi.

UNIVERSITAS BRAWIJAYA



BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1. Lingkungan Implementasi

Implementasi merupakan representasi rancangan berupa aplikasi Enkripsi citra digital menggunakan teknik *Hill cipher* dan Subtitusi. Adapun yang akan dijelaskan dalam subbab ini meliputi lingkungan implementasi perangkat keras dan perangkat lunak.

4.1.1. Lingkungan Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan dan pengujian aplikasi Enkripsi citra digital menggunakan teknik *Hill cipher* dan Subtitusi ini adalah sebuah PC dengan spesifikasi sebagai berikut :

1. Prosesor Intel(R) Core(TM) i3 CPU 530 @ 2.93 GHz. (4 CPU), ~2,9 GHz
2. Memori 2 Gb
3. Harddisk 500 Gb
4. Monitor 19"
5. Keyboard
6. Mouse

4.1.2. Lingkungan Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan program enkripsi citra digital dan uji coba adalah:

1. Sistem operasi *Windows 7* 64-bit sebagai tempat aplikasi dijalankan.
2. *Microsoft Visual C# 2005 Express Edition* sebagai *programming software development* dalam pembuatan Enkripsi citra digital menggunakan teknik *Hill cipher* dan Subtitusi.

4.2. Implementasi Program

Berdasarkan analisa dan perancangan proses yang telah dipaparkan pada Bab III, maka pada bab ini akan dijelaskan proses-proses implementasinya.

4.2.1. Implementasi Enkripsi

Tahap ini adalah tahap setelah proses memasukkan gambar dan mengambil nilai RGB disetiap pixel dan nilainya disimpan pada Array 2 dimensi. Implementasi Enkripsi ditunjukkan pada sourcecode 4.1

1	<code>public void enkripsi(double[,]myKey)</code>
2	<code>{</code>
3	<code>enkipHill(myKey);</code>
4	<code>subtitusi();</code>
5	<code>ubahCitra();</code>
6	<code>}</code>

Source Code 4.1 Prosedur *Enkripsi*

Pada Source Code 4.1 berisikan method enkripsi dengan parameter array 2 dimensi dengan tipe double. Method ini berfungsi untuk memanggil method EnkripsiHill dengan parameter inputan array 2 dimensi bertipe double ditunjukkan pada baris 3, method subtitusi ditunjukkan baris 4, method ubah citra ditunjukkan pada baris 5

4.2.1.1. Implementasi Enkripsi Hill Cipher

Tahap ini adalah salah satu tahap yang dilakukan dalam proses enkripsi. Implementasi Enkripsi *Hill Cipher* ditunjukkan pada sourcecode 4.2.

1	<code>public void enkipHill(double[,] myKey)</code>
2	<code>{</code>
3	<code>byte Er = 0;</code>
4	<code>byte Eg = 0;</code>
5	<code>byte Eb = 0;</code>
6	<code>for (int i = 0; i < tinggi; i++)</code>
7	<code>{</code>
8	<code>for (int j = 0; j < lebar; j++)</code>
9	<code>{</code>
10	<code>for (int k = 0; k < 3; k++)</code>
11	<code>{</code>
12	<code>if (k == 0)</code>
13	<code>{</code> <code>Er = (byte) (((Pixel[i, j].Merah * myKey[0, k]) + (Pixel[i, j].Hijau * myKey[1, k]) +</code> <code>(Pixel[i, j].Biru * myKey[2, k])) % 256);</code> <code>}</code>
14	<code>else if (k == 1)</code>

15	<pre>{ Eg = (byte) (((Pixel[i, j].Merah * myKey[0, k]) + (Pixel[i, j].Hijau * myKey[1, k]) + (Pixel[i, j].Biru * myKey[2, k])) % 256); }</pre>
16	<pre>else</pre>
17	<pre>{ Eb = (byte) (((Pixel[i, j].Merah * myKey[0, k]) + (Pixel[i, j].Hijau * myKey[1, k]) + (Pixel[i, j].Biru * myKey[2, k])) % 256); }</pre>
18	<pre>} Pixel[i, j].Merah = Er; Pixel[i, j].Hijau = Eg; Pixel[i, j].Biru = Eb; }</pre>
19	<pre>}}}</pre>

Source Code 4.2 Prosedur *Enkripsi Hill Cipher*

Pada Source code 4.2 merupakan method enkripHill berfungsi untuk melakukan proses enkripsi pada setiap pixel yang terbagi menjadi RGB dengan metode *Hill Cipher* dengan parameter input array 2 dimensi dengan tipe double, pada baris 13 untuk melakukan enkripsi warna merah dengan cara melakukan perkalian matrik key dengan pixel (RGB). Pada baris 15 untuk melakukan enkripsi warna hijau dengan cara melakukan perkalian matrik key dengan pixel (RGB). Pada baris 17 untuk melakukan enkripsi warna biru dengan cara melakukan perkalian matrik key dengan pixel (RGB).

4.2.2. Implementasi Dekripsi

Pada tahap ini bertujuan untuk mengembalikan gambar yang telah dienkripsi menjadi gambar semula, parameter yang dibutuhkan adalah matrik 2 dimensi. Implementasi *Dekripsi* ditunjukkan pada *Sourcecode 4.3*.

1	<code>public void dekripsi(Double [,]Mykeyl)</code>
2	<code>{</code>
3	<code>dekripHill(Mykeyl);</code>
4	<code>subtitusi();</code>
5	<code>ubahCitra();</code>
6	<code>}</code>

Source Code 4.3 Prosedur *Dekripsi*

Pada source code 4.3 merupakan method dekripsi dengan parameter input array 2 dimensi, method ini berfungsi untuk

memanggil method-method lain untuk melakukan proses dekripsi, method-method yang dipanggil adalah method dekripHill ditunjukkan pada baris 3, method substitusi ditunjukkan baris 4, method ubahcitra ditunjukkan baris 5.

4.2.2.1. Implementasi Dekripsi *Hill Cipher*

Tahap ini adalah salah satu tahap dalam proses enkripsi. Implementasi Dekripsi *Hill Cipher* ditunjukkan pada *Sourcecode 4.4*.

```
1 public void dekripHill(double [,]Mykeyl)
2 {
3     byte Dr = 0;
4     byte Dg = 0;
5     byte Db = 0;
6     for (int i = 0; i < tinggi; i++)
7     {
8         for (int j = 0; j < lebar; j++)
9         {
10            if (k == 0)
11            {
12                Dr = (byte)((Pixel[i, j].Merah * Mykeyl[0, k]) + (Pixel[i, j].Hijau * Mykeyl[1, k]) +
13                (Pixel[i, j].Biru * Mykeyl[2, k])) % 256;
14            else if (k == 1)
15            {
16                Dg = (byte)((Pixel[i, j].Merah * Mykeyl[0, k]) + (Pixel[i, j].Hijau * Mykeyl[1, k]) +
17                (Pixel[i, j].Biru * Mykeyl[2, k])) % 256;
18            }
19            else
20            {
21                Db = (byte)((Pixel[i, j].Merah * Mykeyl[0, k]) + (Pixel[i, j].Hijau * Mykeyl[1, k]) +
22                (Pixel[i, j].Biru * Mykeyl[2, k])) % 256;
23            }
24        }
25        Pixel[i, j].Merah = Dr; Pixel[i, j].Hijau = Dg; Pixel[i, j].Biru = Db;
26    }
27 }
```

Source Code 4.4 Prosedur Dekripsi *Hill Cipher*

Pada Source code 4.4 merupakan method dekripHill berfungsi untuk melakukan dekripsi dengan metode *hill cipher* dengan parameter input array 2 dimensi dengan tipe double. Dekripsi

dilakukan dengan melakukan perkalian matrik antara key yang sudah dilakukan *inverse* dan pixel (RGB). Pada baris 11 untuk melakukan proses dekripsi warna merah, baris 13 untuk melakukan proses dekripsi warna hijau dan baris 17 untuk melakukan proses dekripsi rana biru

4.2.3. Implementasi Subtitusi

Pada tahap ini dilakukan setelah melakukan tahap enkripsi *Hill Cipher* atau dekripsi *Hill Cipher*. Implementasi Subtitusi ditunjukkan pada *Sourcecode 4.5*.

```
1  public void subtitusi(){  
2      String dir = "plus";  
3      int visited = 0;  
4      int iMin = 0, iMax = tinggi - 1;  
5      int jMin = 0, jMax = lebar - 1;  
6      int iCur = 0, jCur = 0;  
7      byte[] indeksR = new byte[(tinggi * lebar)];  
8      byte[] indeksG = new byte[(tinggi * lebar)];  
9      byte[] indeksB = new byte[(tinggi * lebar)];  
10     while (visited < tinggi * lebar)  
11     {  
12         if (dir == "plus")  
13         {  
14             for (jCur = jMin; jCur <= jMax; jCur++)  
15             {  
16                 indeksR[visited] = Pixel[iCur, jCur].Merah;  
17                 indeksG[visited] = Pixel[iCur, jCur].Hijau;  
18                 indeksB[visited] = Pixel[iCur, jCur].Biru;  
19                 visited++;  
20             }  
21             jCur--;  
22             iMin++;  
23             for (iCur = iMin; iCur <= iMax; iCur++)  
24             {  
25                 indeksR[visited] = Pixel[iCur, jCur].Merah;  
26                 indeksG[visited] = Pixel[iCur, jCur].Hijau;  
27                 indeksB[visited] = Pixel[iCur, jCur].Biru;  
28                 visited++;  
29             }  
30             iCur--;  
31             jMax--;  
32             dir = "min";  
33         }  
34         else if (dir == "min")  
35     }
```

```

35  {
36  for (jCur = jMax; jCur >= jMin; jCur--)
37  {
38  indeksR[visited] = Pixel[iCur, jCur].Merah;
39  indeksG[visited] = Pixel[iCur, jCur].Hijau;
40  indeksB[visited] = Pixel[iCur, jCur].Biru;
41  visited++;
42  }
43  jCur++;
44  iMax--;
45  for (iCur = iMax; iCur >= iMin; iCur--)
46  {
47  indeksR[visited] = Pixel[iCur, jCur].Merah;
48  indeksG[visited] = Pixel[iCur, jCur].Hijau;
49  indeksB[visited] = Pixel[iCur, jCur].Biru;
50  visited++;
51  }
52  iCur++;
53  jMin++;
54  dir = "plus";
55  }
56  }
57  Balik(indeksR, indeksG, indeksB);
58  }
59  public void Balik(byte[] indeksR, byte[] indeksG, byte[] indeksB)
60  {
61  byte[] indekBaR = new byte[tinggi * lebar];
62  byte[] indekBaG = new byte[tinggi * lebar];
63  byte[] indekBaB = new byte[tinggi * lebar];
64  int c = 0;
65  for (int i = (tinggi * lebar - 1); i >= 0; i--)
66  {
67  indekBaR[c] = indeksR[i];
68  indekBaG[c] = indeksG[i];
69  indekBaB[c] = indeksB[i];
70  c++;
71  }
72  String dir2 = "plus";
73  int visited2 = 0;
74  int iMin2 = 0, iMax2 = tinggi - 1;
75  int jMin2 = 0, jMax2 = lebar - 1;
76  int iCur2 = 0, jCur2 = 0;
77  while (visited2 < tinggi * lebar)
78  {
79  if (dir2 == "plus")
80  {
81  for (jCur2 = jMin2; jCur2 <= jMax2; jCur2++)
82  {

```

```

83  Pixel[iCur2, jCur2].Merah = indekBaR[visited2];
84  Pixel[iCur2, jCur2].Hijau = indekBaG[visited2];
85  Pixel[iCur2, jCur2].Biru = indekBaB[visited2];
86  visited2++;
87  }
88  jCur2--;
89  iMin2++;
90  for (iCur2 = iMin2; iCur2 <= iMax2; iCur2++)
91  {
92  Pixel[iCur2, jCur2].Merah = indekBaR[visited2];
93  Pixel[iCur2, jCur2].Hijau = indekBaG[visited2];
94  Pixel[iCur2, jCur2].Biru = indekBaB[visited2];
95  visited2++;
96  }
97  iCur2--;
98  jMax2--;
99  dir2 = "min";
100 }
101 else if (dir2 == "min")
102 {
103 for (jCur2 = jMax2; jCur2 >= jMin2; jCur2--)
104 {
105 Pixel[iCur2, jCur2].Merah = indekBaR[visited2];
106 Pixel[iCur2, jCur2].Hijau = indekBaG[visited2];
107 Pixel[iCur2, jCur2].Biru = indekBaB[visited2];
108 visited2++;
109 }
110 jCur2++;
111 iMax2--;
112 for (iCur2 = iMax2; iCur2 >= iMin2; iCur2--)
113 {
114 Pixel[iCur2, jCur2].Merah = indekBaR[visited2];
115 Pixel[iCur2, jCur2].Hijau = indekBaG[visited2];
116 Pixel[iCur2, jCur2].Biru = indekBaB[visited2];
117 visited2++;
118 }
119 iCur2++;
120 jMin2++;
121 dir2 = "plus";
122 }
123 }
124 for (int i = 0; i < tinggi; i++)
125 {
126 for (int j = 0; j < lebar; j++)
127 {
128 Pixel[i, j].Merah = Pixel[i, j].Merah;
129 Pixel[i, j].Hijau = Pixel[i, j].Hijau;
130 Pixel[i, j].Biru = Pixel[i, j].Biru;

```

131	}
132	}
133	ubahCitra();
134	}

Source Code 4.5 Prosedur Subtitusi

Source code 4.5 merupakan method subtitusi berfungsi untuk melakukan proses enkripsi atau dekripsi Subtitusi. Baris 10 – 56 berfungsi untuk membaca pixel secara spiral yaitu dimulai dari pojok kiri atas sampai ketengah, hasil pembacaan pixel tersebut disimpan dalam array 1 dimensi yaitu (indeksR, indeksG, indeksB). Kemudian dimasukkan ke dalam method balik. Method balik berfungsi untuk membalik pixel awal menjadi akhir. Baris 65 -71 berfungsi untuk mebalik pixel sedangkan baris 72 – 134 berfungsi untuk menata kembali pixel yang sudah dibalik menjadi array 2 dimensi dan menampilkannya.

4.2.4. Implementasi Menghitung Determinan

Tahap ini berfungsi untuk menghitung determinan terhadap key, sekaligus untuk mengecek apakah key memiliki determinan atau tidak, apabila memiliki determinan maka key dapat digunakan untuk melakukan proses enkripsi atau dekripsi. Implementasi determinan ditunjukkan oleh *soucecode 4.6*.

1	double determinan(double[,] k)
2	{
3	double d;
4	$d = (((k[0, 0] * k[1, 1] * k[2, 2]) + (k[0, 1] * k[1, 2] * k[2, 0]) + (k[0, 2] * k[1, 0] * k[2, 1]) - (k[2, 0] * k[1, 1] * k[0, 2]) - (k[2, 1] * k[1, 2] * k[0, 0]) - (k[2, 2] * k[1, 0] * k[0, 1])) \% 256 + 256) \% 256;$
5	for(int i=0;i<257;i++){
6	if ((d * i) \% 256 == 1)
7	{
8	x = i;
9	break;
10	}
11	return (x);
12	}

Source Code 4.6 Prosedur Determinan

Source code 4.6 merupakan method determinan dengan parameter input array 1 dimensi dengan tipe double dan menghasilkan nilai yang bertipe double. Method ini berfungsi untuk mencari determinan key. Baris 4 merupakan rumus determinan

matrik 3 dimensi, baris 6 – 10 berfungsi untuk mencari nilai x yang nantinya akan digunakan dalam proses inverse

4.2.5. Implementasi Menghitung Koofaktor

Tahap ini berfungsi untuk mencari koofaktor dari yang nantinya digunakan untuk proses dekripsi *Hill Cipher*. Implementasi Koofaktor ditunjukkan *Sourcecode 4.7*.

1	<code>public void kofaktor() {</code>
2	<code>for (int i = 0; i < 3; i++)</code>
3	<code>{</code>
4	<code>for (int j = 0; j < 3; j++)</code>
5	<code>{</code>
6	<code>koofaktor[i, j] = Math.Pow(-1, i + j) * (Minnor(i, j));</code>
7	<code>}</code>
8	<code>}</code>
9	<code>}</code>

Source Code 4.7 Prosedur Koofaktor

Source code 4.7 merupakan method kofaktor berfungsi untuk menghitung kooaktor. Baris 6 merupakan rumus untuk mencari koofaktor dari matrik key.

4.2.6. Implementasi Inverse

Tahap ini berfungsi untuk menghitung key, matrik *inverse* key inilah yang digunakan untuk mengembalikan gambar menjadi semula. Implementasi *Inverse* key ditunjukkan oleh *sourcecode 4.8*

1	<code>public void inverskey() {</code>
2	<code>kofaktor();</code>
3	<code>tranpose();</code>
4	<code>for (int i = 0; i < 3; i++)</code>
5	<code>{</code>
6	<code>for (int j = 0; j < 3; j++)</code>
7	<code>{</code>
8	<code>IKey[i, j] = ((x * Tkofaktor[i, j])%256+256)%256;</code>
9	<code>}</code>
10	<code>}</code>
11	<code>}</code>

Source Code 4.8 Prosedur Inverse

Source code 4.8 merupakan method inversekey berfungsi untuk mencari inverse dari matrik key. Baris 8 merupakan rumus untuk menghitung nilai inverse key

4.2.7. Implementasi Minnor

Tahap ini berfungsi untuk melakukan proses *minnor* terhadap matrik key dan digunakan saat mencari koofaktor matrik key. Implementasi Minnor ditunjukkan pada *sourcecode* 4.9

1	<code>double Minnor(int a, int b){</code>
2	<code>double [,] s=new double [2,2];</code>
3	<code>int kolom = 0;</code>
4	<code>int baris = 0;</code>
5	<code>for (int i = 0; i < 3; i++) {</code>
6	<code>for (int j = 0; j < 3; j++) {</code>
7	<code>if (i != a && j != b) {</code>
8	<code>s[kolom, baris] = Key[i, j];</code>
9	<code>kolom++;</code>
10	<code>if (kolom == 2) {</code>
11	<code>kolom = 0;</code>
12	<code>baris++;</code>
13	<code>}</code>
14	<code>}</code>
15	<code>}</code>
16	<code>}</code>
17	<code>m = (s[0, 0] * s[1, 1]) - (s[0, 1] * s[1, 0]);</code>
18	<code>return (m); }</code>

Source Code 4.9 Prosedur Minnor

Source code 4.9 merupakan method *minnor* dengan 2 parameter input yang bertipe double dan menghasilkan nilai bertipe double. Method ini berfungsi untuk menghitung minnor dari matrik key yang nantinya digunakan untuk mencari koofaktor. Baris 17 merupakan rumus untuk mencari minnor.

4.2.8. Implementasi Tranpose

Pada tahap ini dilakukan tranpose terhadap matrik key yang telah dibentuk menjadi matrik koofaktor. Implementasi Tranpose ditunjukkan pada *sourcecode* 4.10

1	<code>public void tranpose() {</code>
2	<code>for (int i=0;i<3;i++){</code>
3	<code>for (int j = 0; j < 3; j++) { Tkoofaktor[j, i]=koofaktor [i,j];</code>

4	}
5	}
6	}

Source Code 4.10 Prosedur Tranpose

Source code 4.10 merupakan method tranpose berfungsi untuk melakukan tranpose terhadap koefaktor. Baris 3 merupakan rumus untuk melakukan tranpose.

4.3. Implementasi Antarmuka

Berdasarkan rancangan antarmuka yang telah dikemukakan pada Bab 3 maka dihasilkan antaramuka Enkripsi Citra Digital pada gambar 4.1.



Gambar 4.1 Antarmuka Enkripsi Citra Digital

Keterangan gambar 4.1 :

1. *Button Open* berfungsi untuk membuka file gambar yang memiliki format .jpg, .bmp atau .png.
2. *Button Input Key* berfungsi untuk memasukkan matrik key yang berukuran 3x3.
3. *Button Enkripsi* berguna untuk melakukan Proses Enkripsi

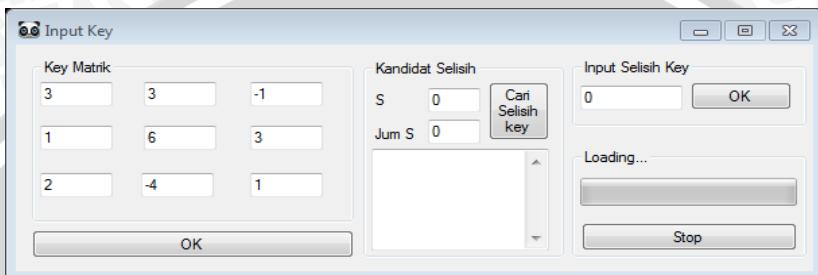
4. *Button Dekripsi* berguna untuk melakukan proses Dekripsi.
5. *Tool Strip save* berguna untuk menyimpan hasil enkripsi atau dekripsi, format penyimpanan file yaitu .png atau .bmp.
6. *Combo box file info* berguna untuk menampilkan info file yang ditampilkan.
7. *Combo box Histogram* berguna untuk menampilkan Histogram dari masing-masing gambar.
8. *Text field waktu proses* berguna untuk menampilkan waktu saat melakukan proses enkripsi atau dekripsi.

Pada antarmuka tersebut terdapat beberapa tombol, tombol yang aktif ketika dijalankan aplikasi ini adalah tombol open, dan tombol yang lain belum diaktifkan hal ini dilakukan agar user tidak mengalami kesalahan ketika menjalankan aplikasi. Ketika user menekan tombol open maka akan keluar halaman yang menampilkan gambar yang ingin digunakan setelah selesai membuka gambar maka gambar akan ditampilkan seperti gambar 4.4



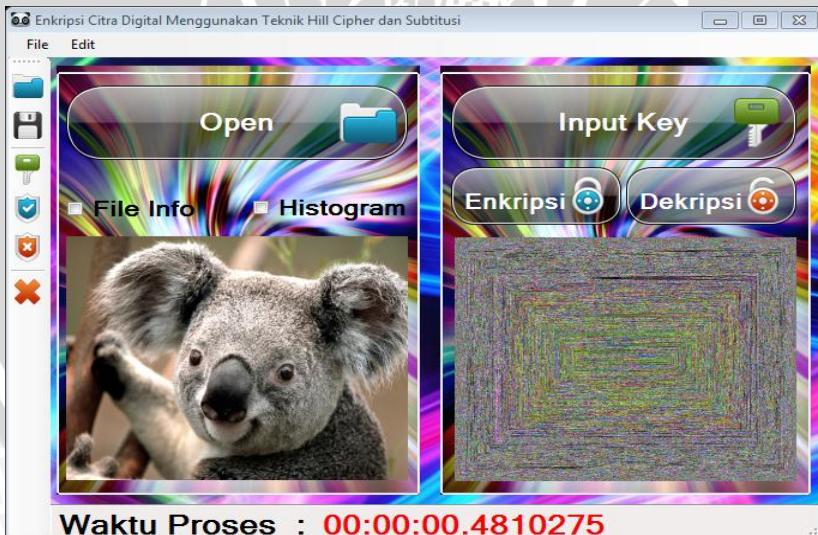
Gambar 4.2 Antarmuka Open File

Setelah gambar berhasil dibuka maka tombol yang diaktifkan adalah input key. Key yang dimasukkan harus memiliki invers. berikut adalah tampilan form input key ditunjukkan pada gambar 4.3



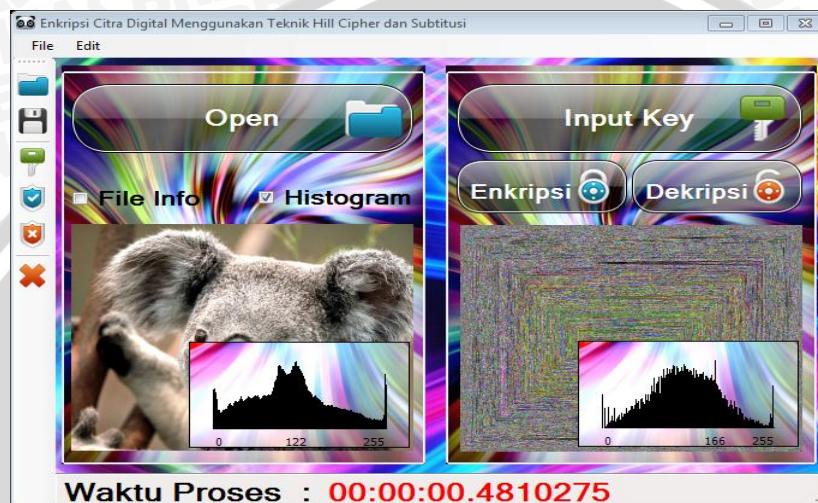
Gambar 4.3 Antarmuka Input Key

Setelah key berhasil dimasukkan maka tombol yang diaktifkan adalah tombol enkripsi dan dekripsi. Apabila user memasukkan gambar yang sudah jelas, maka dilakukan proses enkripsi dan apabila memasukkan gambar yang tidak jelas maka dilakukan dekripsi. berikut adalah tampilan apabila user menekan tombol enkripsi. Gambar 4.4 Antarmuka Enkripsi dan Dekripsi

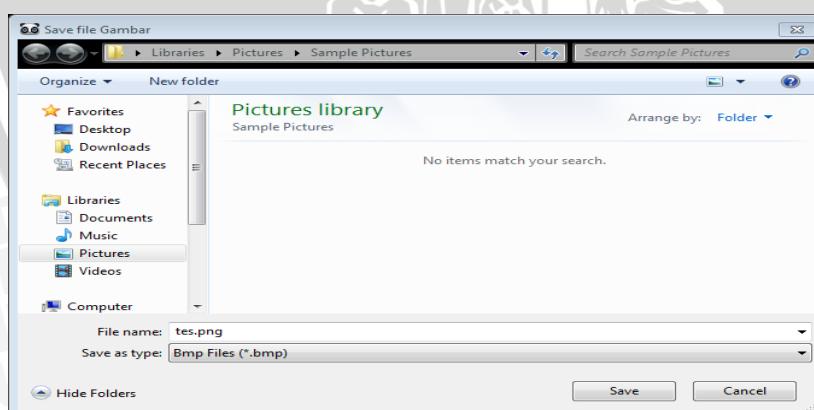


Gambar 4.4 Antarmuka Enkripsi dan Dekripsi

Apabila ingin melihat komposisi warna pada masing-masing gambar tersebut maka centang *checkbox Histogram*. Antarmuka histogram ditunjukkan pada Gambar 4.5

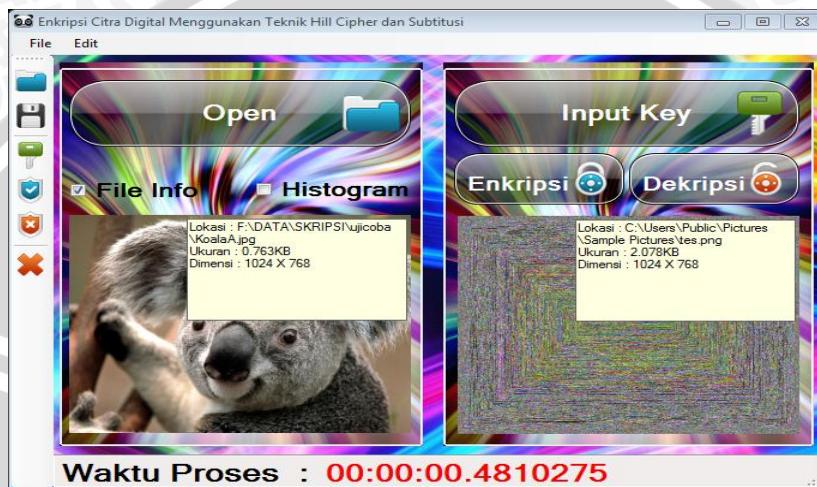


Apabila ingin melakukan penyimpanan terhadap hasil proses enkripsi atau dekripsi, maka tekan *tool strip save*. Penyimpanan file dapat dilakukan dengan format .png atau .bmp. Antarmuka proses penyimpanan ditunjukkan pada Gambar 4.6



Gambar 4.6 Antarmuka Save File

Apabila ingin melihat info file dari hasil penyimpanan tadi maka centang pada *checkbox File info*. File info meliputi Lokasi, ukuran dan dimensi dari file tersebut. Berikut adalah antarmuka file info ditunjukkan pada Gambar 4.7



Gambar 4.7 Antarmuka *File Info*

4.4. Uji Coba

Uji coba dilakukan dengan melakukan 4 macam percobaan yaitu uji coba pengukuran parameter waktu proses, uji coba perbandingan ukuran *file input* dan *file output*, uji coba analisis keamanan dan uji coba ketahanan *cipher image*. Dalam melakukan uji coba digunakan matrik key, yaitu sebagai berikut.

$$K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix}, \text{ Matrikkey ini digunakan ketika melakukan enkripsi.}$$

4.4.1. Uji Coba Pengukuran Parameter Waktu Proses

Uji coba Pengukuran Parameter waktu dilakukan dengan mencatat setiap proses enkripsi atau dekripsi dengan ukuran file yang beragam, tujuan dari pengujian ini adalah untuk mengetahui kecepatan proses enkripsi atau dekripsi dengan metode *hill cipher* dan *subtitusi*. Percobaan dilakukan sebanyak 7 kali dengan gambar ukuran dan dimensi yang beragam. Hasilnya seperti table 4.1.

Tabel 4.1 Tabel Uji Coba Parameter Waktu Proses

Nama File	Ukuran (KB) / Dimensi	Gambar	Waktu Proses (s)	
			Enkripsi	Dekripsi
Hiu.jpg	1.192 / 2560 x 1600		00:00:02.434 1392	00:00:02. 4161382
koalaA.jpg	763 / 1024 x 768		00:00:00.507 0290	00:00:00. 4960284
Orang A.jpg	438 / 1600 x 1200		00:00:01.149 0657	00:00:01. 2110693
orang B.jpg	551 / 1600 x 1200		00:00:01.145 0655	00:00:01. 1530659
Pemandangan A.jpg	1.196 / 1920 x 1080		00:00:01.219 0698	00:00:01. 2190697
Pemandangan B.jpg	1.072 / 2560 x 1600		00:00:02.412 1380	00:00:02. 4341393
Singa.jpg	664 / 1920 x 1200		00:00:01.351 0772	00:00:01. 3550775

Dari tabel hasil penelitian diatas dapat diketahui rata-rata waktu prosesnya adalah:

Rata – rata waktu proses enkripsi sebesar 1,459655 detik

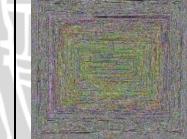
Rata – rata waktu proses dekripsi sebesar 1,885608 detik

Dari hasil tersebut rata-rata waktu yang dibutuhkan untuk melakukan enkripsi atau dekripsi tidak sampai 2 detik dan memiliki selisih sebesar 0,42595.

4.4.2. Uji Coba Perbandingan Ukuran *File Input* dan *File Output*

Uji coba perbandingan antara ukuran file setelah dilakukan proses enkripsi atau dekripsi dan sebelum dilakukan proses, tujuan dari uji coba ini adalah untuk mengetahui perbedaan ukuran file sebelum dan setelah dilakukan proses enkripsi atau dekripsi. Percobaan dilakukan sebanyak 7 kali dengan gambar, ukuran dan dimensi yang beragam. Hasilnya ditunjukkan pada Tabel 4.2 Tabel Uji Coba Perbandingan Ukuran *File Input* dan *File Output*.

Tabel 4.2 Tabel Uji Coba Perbandingan Ukuran *File Input* dan *File Output*

Nama File		Gambar		Ukuran File (kb)	
Input	Output	Input	Output	Input	Output
Hiubm p.bmp	Hiupn gE.png			11.701	6.817
Koala Apng. png	Koala Abmp E.Bmp			1.911	2.305
Orang A.jpg	Orang ApngE .png			0.438	5.016

Orang Bpng. png	Orang Bbmp E.Bmp			4.381	5.626
Pemandangan A.jpg	PemandanganApng E.png			1.196	4.576
PemandanganBbmp. bmp	PemandanganBpng E.png			12.001	10.067
Singa.jpg	Singa. bmp			0.664	6.751

Dari tabel hasil penelitian diatas dapat diketahui rata-rata ukuran file setelah dan sebelum dilakukan proses enkripsi atau dekripsi.

Rata-rata ukuran *file input* : 6.238 kb

Rata-rata ukuran *file output* : 6.784 kb

Dari hasil tersebut rata-rata ukuran *file input* dan *output* hampir sama dan memiliki selisih sebesar 546 kb

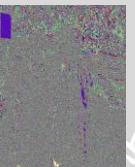
4.4.3. Uji Coba Analisa Keamanan

Uji coba analisa keamanan dilakukan dengan cara memberikan key yang hampir mirip dengan key yang asli, dalam kasus ini key berbentuk matrik angka berordo 3x3, sehingga uji coba key yang diberikan mempunyai selisih key yang mendekati key asli.Uji coba dilakukan sebanyak 3 kali dengan gambar **orangBpngE.bmp** namun selisih key yang berbeda-beda dan

menggunakan key $K = \begin{bmatrix} 3 & 3 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 1 \end{bmatrix}$.

Tujuan dari uji coba ini adalah untuk mengetahui apakah dengan key yang hampir mirip dapat melakukan dekripsi dengan baik. Hasil dari uji coba ini ditunjukkan pada tabel 4.3 uji coba Analisa Keamanan

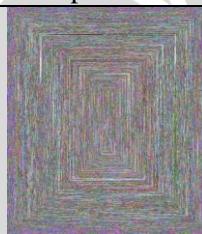
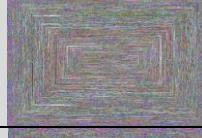
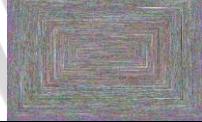
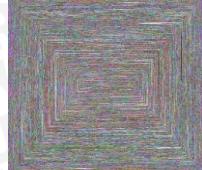
Tabel 4.3 Tabel uji coba Analisa Keamanan

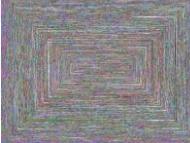
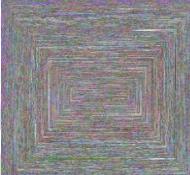
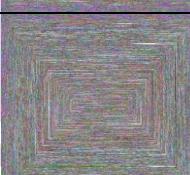
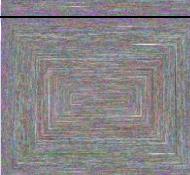
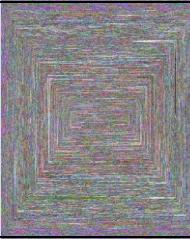
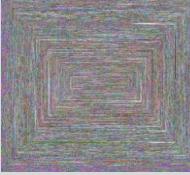
Gambar Awal	Gambar Enkripsi	Selisih key	Hasil	Kesimpulan
 orangB.jpg	 orangBp ngE.bmp	0,5	 orangBD0, 5.png	Tidak begitu jelas gambar orangnya. Tetapi bentuk kotak tembok disebelah pojok kiri atas masih terlihat namun dengan warna yang berbeda.
		0,01	 orangBD0, 01.png	Tidak begitu jelas gambar orangnya. Tetapi bentuk kotak tembok disebelah pojok kiri atas masih terlihat namun dengan warna yang berbeda
		0,06	 orangBD0, 06.png	Tidak begitu jelas gambar orangnya. Tetapi bentuk kotak tembok disebelah pojok kiri atas masih terlihat namun dengan warna yang berbeda

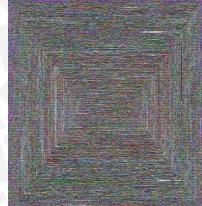
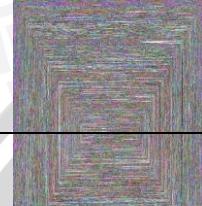
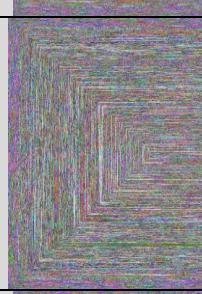
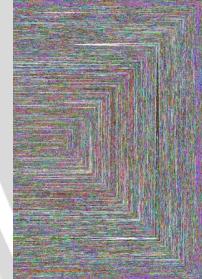
4.4.4. Uji Coba Ketahanan *Image Cipher*

Uji coba ketahanan *image cipher* dilakukan dengan cara melakukan perubahan pada gambar yang sudah dilakukan proses enkripsi. Uji coba ini bertujuan untuk menguji ketahanan *image cipher*. Uji coba dilakukan dengan 17 metode perubah gambar dan menggunakan 1 gambar yang telah dienkripsi yaitu **orangApngE.png**. hasil dari uji coba tersebut ditunjukkan pada tabel 4.4 Tabel uji coba Ketahanan *Image Cipher*

Tabel 4.4 Tabel uji coba Ketahanan *Image Cipher*

Image Cipherteks	Metode Perubahan	Hasil	Kesimpulan
	Rotasi (90 ⁰)		Tidak dapat dikenali
	Rotasi (180 ⁰)		Tidak dapat dikenali
	Rotasi (270 ⁰)		Tidak dapat dikenali
	Diperbesar(200%)		Tidak dapat dikenali
	Diperbesar(300%)		Tidak dapat dikenali

	Diperbesar(150%)		Tidak dapat dikenali
	Diperkecil(75%)		Tidak dapat dikenali
	Diperkecil(50%)		Tidak dapat dikenali
	Diperkecil(25%)		Tidak dapat dikenali
	Noise Despeckle		Masih samar-samar terlihat
	Gaussian Blur		Tidak dapat dikenali
	Pixelate Facet		Tidak dapat dikenali

	Texture Mosaic		Masih samar-samar terlihat
	Crop atas		Tidak dapat dikenali
	Crop bawah		Tidak dapat dikenali
	Crop kiri		Tidak dapat dikenali
	Crop kanan		Tidak dapat dikenali

4.5. Analisa Hasil

Pada sub bab ini menjelaskan atau menganalisa hasil dari uji coba yang sudah dilakukan sebelumnya. Analisa hasil dilakukan sebanyak 4 kali yaitu sebagai berikut.

4.5.1. Analisa Hasil Pengukuran Parameter Waktu Proses

Dari hasil uji coba pengukuran parameter waktu proses yang sudah dilakukan, dapat diketahui waktu proses saat melakukan enkripsi maupun dekripsi tidak sampai 1 menit dengan dimensi dan ukuran file yang cukup besar.

Gambar yang memiliki waktu proses yang paling cepat adalah koalaA.jpg dengan ukuran file 763 kb, dimensi 1024 x 768, waktu enkripsi 00:00:00.5070290 dan waktu dekripsi 00:00:00.4960284. sedangkan gambar yang memiliki waktu proses yang paling lama adalah pemandanganB.jpg dengan ukuran file 1.072 kb, dimensi 2560 x 1600, waktu enkripsi 00:00:02.4121380 dan waktu dekripsi 00:00:02.4341393 dari hasil tersebut dapat diketahui apabila gambar memiliki dimensi yang besar maka memiliki waktu proses yang besar pula, hal ini disebabkan metode *Hill cipher* dan subtitusi melakukan enkripsi atau dekripsi pada setiap pixel, jadi semakin besar dimensi gambar maka semakin banyak juga jumlah pixel. Hal ini lah yang menyebabkan waktu enkripsi atau dekripsi lama. Ukuran file tidak terlalu berpengaruh karena kecepatan proses metode *Hill cipher* dan subtitusi tergantung dari jumlah pixel atau dimensi dari gambar.

4.5.2. Analisa Hasil Perbandingan Ukuran *File Input* dan *File Output*

Dari hasil percobaan perbandingan ukuran *File Input* dan *File Output*, dapat diketahui bahwa hasil dari proses enkripsi maupun dekripsi bervariasi, apabila disimpan dengan format .bmp maka file akan menjadi lebih besar dengan catatan *File Input* tersebut harus .jpg atau .png. apabila disimpan dengan format .png maka ukuran file tersebut akan menjadi lebih kecil dengan catatan *File Input* tersebut harus memiliki format .bmp.

Hasil enkripsi atau dekripsi dengan metode *Hill Cipher* dan Subtitusi tidak akan merubah ukuran file jika *File Output* disimpan dengan format file yang sama dengan format *File Input*. Misal *File Input* berformat .bmp maka *File Output* juga harus disimpan dengan format .bmp.

4.5.3. Analisa Hasil Kekuatan Keamanan

Dari hasil percobaan kekuatan keamanan dengan memberikan key yang hampir benar ternyata menghasilkan gambar yang masih sulit untuk dikenali.

Percobaan dengan memasukkan key yang memiliki selisih key sebesar 0,5 menghasilkan gambar yang masih sulit dikenali, dengan selisih key sebesar 0,01 menghasilkan gambar yang relatif masih sulit dikenali tetapi warna tembok yang berwarna putih memiliki RGB 255,255,255 dirubah menjadi warna merah muda yang memiliki RGB tinggi mendekati warna putih. Sedangkan selisih key sebesar 0,06 menghasilkan gambar yang masih sulit dimengerti.

Kelemahan metode *Hill Cipher* dan Subtitusi adalah apabila berwarna hitam (RGB 0,0,0) maka tidak dapat dilakukan enkripsi karena metode ini menggunakan perkalian matrik jadi bilangan berapapun apabila dikalikan dengan $RGB = 0$ maka tetap akan menghasilkan $RGB = 0$ atau hitam.

4.5.4. Analisa Hasil Ketahanan *Image Cipher*

Dari percobaan yang sudah dilakukan yaitu uji coba ketahanan *Image Cipher* menghasilkan bahwa dengan menggunakan metode *Hill Cipher* dan Subtitusi *Image Cipher* sama sekali tidak kuat.

Ketika dilakukan perubahan *Image Cipher* secara rotasi, cropping, dan pembesaran menghasilkan gambar tidak jelas ketika dilakukan dekripsi. Hal ini disebabkan metode subtitusi yang mensubtitusi setiap pixel secara spiral.

Ada satu metode perubahan *Image Cipher* yang menghasilkan gambar yang terlihat agak jelas ketika dilakukan proses dekripsi yaitu ketika dilakukan metode noise dispeckle dengan menggunakan aplikasi photosop, Sehingga dapat diketahui *Image Cipher* dengan metode enkripsi *Hill Cipher* dan Subtitusi ternyata cukup kuat dengan metode perubahan *Image Cipher* noise dispeckle.

4.5.5. Analisa Hasil Keseluruhan

Dari semua uji coba yang sudah dilakukan dapat diketahui kekurangan dan kelebihan metode enkripsi citra digital menggunakan teknik *Hill Cipher* dan Subtitusi, yaitu sebagai berikut.

Kelemahan metode ini adalah sebagai berikut.

1. Tidak dapat melakukan enkripsi pixel yang nilai RGB sama dengan 0 atau warna hitam
2. Ketahanan *Image Cipher* sangat lemah, sehingga apabila *Image Cipher* sedikit dirubah maka gambar tidak dapat dikembalikan seperti semula.
3. Tidak semua matrik key dapat digunakan untuk melakukan proses enkripsi atau dekripsi, tetapi harus berbentuk matrik yang memiliki invers.
4. Tidak dapat melakukan penyimpanan *file* gambar hasil proses enkripsi atau dekripsi dengan format .jpg yang ukurannya lebih kecil dibandingkan format .bmp dan .png. Hal ini dikarenakan format .jpg melakukan kompresi terlebih dahulu sehingga *file* gambar dapat berukuran kecil. Apabila gambar hasil enkripsi disimpan dengan format .jpg maka tidak dapat didekripsi kembali seperti gambar semula hal ini disebabkan karena sifat format *file* .jpg yang melakukan kompresi terhadap *file* gambar.

Kelebihan dari metode enkripsi menggunakan teknik *Hill Cipher* dan Subtitusi, adalah sebagai berikut.

1. Kekuatan gambar hasil proses enkripsi cukup kuat, menurut hasil percobaan yang telah dilakukan dengan memasukkan key yang memiliki selisih sebesar 0,01 dari key yang benar, masih menghasilkan gambar yang kurang jelas dan warna yang tidak berhasil kembali seperti semula
2. Kecepatan proses enkripsi atau dekripsi tergantung dari ukuran dimensi bukan dari ukuran *File* gambar. Hal ini disebabkan metode *Hill Cipher* dan Subtitusi melakukan enkripsi disetiap pixel.
3. Ukuran *file* setelah dilakukan proses enkripsi atau dekripsi tetap. Kecuali *file input* berformat .jpg karena format file .jpg adalah format yang telah dilakukan kompresi terlebih dahulu sehingga ukurannya lebih kecil dibandingkan *file* yang berformat .bmp dan .png.
4. Komposisi warna gambar hasil proses enkripsi sangat berbeda dari komposisi warna gambar semula, sehingga gambar sulit untuk dimengerti. Hal ini dapat dilihat dari histogram yang terbentuk sebelum gambar dilakukan enkripsi dan sesudah dilakukan enkripsi.

UNIVERSITAS BRAWIJAYA



BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah melakukan penelitian maka dapat disimpulkan bahwa

1. Metode enkripsi penggabungan *Hill Cipher* dan Subtitusi dapat diimplementasikan pada data yang berbentuk citra digital dengan melakukan perhitungan perkalian matrik antara nilai RGB citra digital dengan key ketika melakukan proses enkripsi dan ketika melakukan proses dekripsi dilakukan dengan mengalikan matrik RGB citra digital dengan matrik key yang sudah diinverse kemudian dilakukan proses subtitusi.
2. Dari hasil percobaan yang sudah dilakukan dapat diketahui rata-rata kecepatan proses enkripsi sebesar 1,459655 detik dan rata-rata kecepatan proses dekripsi sebesar 1,885608 detik, selisih diantaranya sebesar 0,42595 detik. Rata rata Perbedaan Ukuran *file input* dan *output* sebesar 546 kb. Keamanan data, hasil enkripsi tidak dapat dikembalikan seperti semula apabila key yang dimasukkan salah walaupun dengan key yang hampir mirip dengan key yang benar, belum dapat mengembalikan gambar seperti semula. Ketahanan hasil enkripsi sangat lemah tidak ada satu metodepun yang berhasil dikembalikan seperti semula ketika dilakukan dekripsi.

5.2. Saran

Untuk pengembangan lanjut perangkat lunak maka ada beberapa saran yang dapat diberikan :

1. Dapat melakukan penelitian lebih lanjut terhadap faktor-faktor lain yang mempengaruhi hasil enkripsi citra digital. Faktor-faktor tersebut dapat digunakan sebagai parameter dari metode enkripsi Penggabungan *Hill Cipher* dan subtitusi.

UNIVERSITAS BRAWIJAYA



DAFTAR PUSTAKA

- Andleigh, Prabhat K; Thakhar, Kiran. 1995. *Multimedia System Design*. Prentice Hall, Inc, New Jersey.
- Anton, Howard and Chris R. 2005. *Aljabar Linier Elementer Versi Aplikasi, edisi 8 jilid 2*. Erlangga, Jakarta
- Aryus, Dony, 2008. *Pengantar Ilmu kriptografi keamanan Teori analisis dan Implementasi*. Andi. Yogyakarta.
- Bellman, Richard. 1970. *Introduction to Matrix Analysis, 2nd ed.* McGraw-Hill Book Company, New York
- Bruaaldi, Richard A., and Herbert J. Ryser. 1991. *Combinatorial Matrix Theory*. Cambridge University Press, New York
- Bruce, Schneier. 1996. *Applied Cryptography*, John Wiley & Sons, New York
- Datta, Biswa Nath. 1995. *Numerical Linear Algebra and Applications*. Brooks/Cole Publishing Company, Pacific Grove
- Gantmacher, F R. 1960. *The Thery of Matrices, 2 vols.* Chelsea Publishing Company, Inc, New York
- Golub, Gene H., and Charles F Van Loan. 1996. *Matrix Computations, 3rd ed.* Johns Hopkins University Press, Baltimore
- Horn, Roger A., and Charles R. Johnson. 1985. *Matrix Analysis*. Cambridge University Press, New York
- Jepsen, Charles H., and Eugene A. Herman. 1988. *Matrix Algebra Calculator: Linear Algebra Problems for Computer Solution*. Brooks/Cole Publishing Company, Pacific Grove

Jolly Shah, Vikas Saxena, 2011. *Performance Study on Image Encryption Schemes*, IJSI International Journal of Computer Science Issues, New York

Jonathan M.Blackledge, Musheer Ahmad, Omar Farooq , 2010 . *Chaotic Image Encryption Algorithm Based on Frequency Domain Scrambling*. School of Electrical Engineering Systems, Dublin Institute of Technology

Kurniawan, Yusuf. 2004, *Kriptografi Keamanan Internet dan Jaringan Komunikasi*. Informatika. Bandung.

Lancester, Peter, and M. Tismenetsky. 1985. *The Theory of Matrices with Applications*, 2nd ed. Academic Press, New York

Lester S. Hill. 1929. *Cryptography in an algebraic Alphabet*. Mathematical Association of America, New York

Lester S. Hill. 1931. *Concerning Certain Linier Transformation Apparatus of Cryptography*. Mathematical Association of America, New York

Ortega, James M. 1987. *Matrix Theory: A Second Course*. Plenum Press, New York

Suryani, Esti., dan Titin Sri Martini.2008.*Kombinasi Kriptografi Dengan Hill Cipher Dan Steganografi Dengan LSB Untuk Keamanan Data*. Universitas Muhammadiyah Magelang.

Watkins, David S. 1991. *Fundamentals of Matrix Computation*. Willey, New York