

**IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI
PADA CITRA DIGITAL MENGGUNAKAN ALGORITMA
RIVEST SHAMIR ADLEMAN (RSA) DAN METODE
DISCRETE COSINE TRANSFORM (DCT)**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

DOVIE YUDHAWIRATAMA NIM. 1341180106



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2017**

**IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI
PADA CITRA DIGITAL MENGGUNAKAN ALGORITMA
RIVEST SHAMIR ADLEMAN (RSA) DAN METODE
DISCRETE COSINE TRANSFORM (DCT)**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

DOVIE YUDHAWIRATAMA NIM. 1341180106



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2017**

HALAMAN PENGESAHAN

IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI PADA CITRA DIGITAL MENGGUNAKAN ALGORITMA RIVEST SHAMIR ADLEMAN (RSA) DAN METODE DISCRETE COSINE TRANSFORM (DCT)

Disusun oleh :

DOVIE YUDHAWIRATAMA NIM. 1341180106

Skripsi ini telah diuji pada tanggal 8 Juni 2017

Disetujui oleh:

- | | | | |
|------------------|---|--|-------|
| 1. Penguji I | : | <u>Dr.Eng. Faisal Rahutomo, ST., M.Kom</u> | |
| | | NIP. 197711162005011008 | |
| 2. Penguji II | : | <u>Ekojono, ST., M.Kom</u> | |
| | | NIP. 195912081985031004 | |
| 3. Pembimbing I | : | <u>Dr. Eng. Rosa Andrie A., S.T., M.T.</u> | |
| | | NIP. 198010102005011001 | |
| 4. Pembimbing II | : | <u>Arida Ferti Syafiandini, S.Kom, M.Kom</u> | |
| | | NIP. | |

Mengetahui,

Ketua Jurusan
Teknologi Informasi

Ketua Program Studi
Teknik Informatika

Rudy Ariyanto, S.T., M.Cs.
NIP. 19711110 199903 1 002

Ir. Deddy Kusbianto P., M.MKom.
NIP. NIP. 19621128 198811 1 001

PERNYATAAN

Dengan ini saya menyatakan bahwa Skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar Kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka

Malang, 8 Juni 2017

Dovie Yudhawiratama

ABSTRAK

Yudhawiratama, Dovie. “Implementasi Kriptografi dan Steganografi Pada Citra Digital Menggunakan Algoritma Rivest Shamir Adleman (RSA) dan Metode Discrete Cosine Transform (DCT)”. **Pembimbing: (1) Dr. Eng. Rosa Andrie A., S.T., M.T., (2) Arida Ferti Syafiandini, S.Kom., M.Kom.**

Skripsi, Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, 2017

Internet menjadi salah satu media untuk mendapatkan sebuah informasi yang memudahkan masyarakat untuk melakukan pertukaran dan pengambilan informasi. Keamanan dan kerahasiaan data dan informasi menjadi salah satu hal penting yang harus diperhatikan, dikarenakan semakin berkembangnya teknologi informasi semakin berkembang pula teknik kejahatan komputer. Teknik kriptografi dan stegano-grafi dapat digunakan untuk memberi perlindungan keamanan pada pesan rahasia.

Penelitian ini bertujuan untuk mengkombinasikan kriptografi RSA yang terintegrasi dengan metode steganografi, untuk memberikan proteksi ganda pada pesan rahasia di dalam sebuah gambar/citra digital. Aplikasi berjalan pada sistem operasi berbasis Windows, dan hanya dapat menyembunyikan pesan teks dalam gambar berformat bitmap atau jpeg. Pengembangan teknik kriptografi dengan algoritma *Rivest Shamir Adleman* (RSA), yang diintegrasikan ke dalam steganografi dengan metode *Discrete Cosine Transform* (DCT) diharapkan dapat melindungi pesan rahasia yang akan disisipkan kedalam citra.

Berdasarkan hasil penelitian, Metode Discrete Cosine Transform dapat menyembunyikan pesan yang dienkripsi. Dalam hasil ujicoba yang dilakukan, untuk citra berukuran 512x512 dapat menampung pesan sepanjang 4096 biner. Sedangkan dengan pengujian MSE dan PSNR, didapatkan hasil yang bagus dengan nilai $MSE = 2.7081$ dan $PSNR = 43.8062$. Penyisipan pesan dilakukan pada frekuensi tinggi dari nilai koefisien DCT agar hasil penyisipan tidak terlihat secara kasat mata. Penyisipan juga bergantung pada nilai yang ditambahkan pada koefisien DCT, semakin kecil nilai koefisien maka nilai yang disisipkan akan hilang.

Kata kunci: *RSA, DCT, kriptografi, steganografi, dan keamanan data.*

ABSTRACT

Yudhawiratama, Dovie. "Implementation of Cryptography and Steganography on Digital Imagery Using Rivest Shamir Adleman (RSA) Algorithm and Discrete Cosine Transform (DCT) Method". Counseling Lecturer: (1) Dr. Eng. Rosa Andrie A., S.T., M.T., (2) Arida Ferti Syafiandini, S.Kom., M.Kom.

Minithesis, Informatics Engineering Study Programme, Department of Information Technology, State Polytechnic of Malang, 2017.

The Internet becomes one of the media to get an information that allows people to exchange information and retrieve information. The security and confidentiality of data and information become one of the important things that must be considered, because not only information technology that is growing but also computer crime techniques as well. Cryptographic techniques and steganography can be used to give a protection on secret messages.

This research combines integrated RSA cryptography with steganography method, to provide double protection for messages in digital images. The application runs on a Windows-based operating system, and can only hide messages in bitmap or jpeg formatted images. Developing cryptographic techniques with the Rivest Shamir Adlemamn (RSA) algorithm, integrated into steganography using the Discrete Cosine Transform (DCT) method, is expected to provide confidential messages.

Based on the result of this research, Discrete Cosine Transform Method can hide message of encrypted messages. In the test results, for a 512x512 image can hide messages with 4096 binary characters long. MSE and PSNR testing shows good results with value of $MSE = 2.7081$ and $PSNR = 43.8062$. Insertion of messages done in high frequencies of DCT process and the results of the insertion convinced that the messages is not visible to the eye. The insertion of the messages also depends on what is added to the DCT Coefficient, the smaller the amount of coefficient, the greater the results of losing some data.

Keywords: *RSA, DCT, cryptography, steganography, and data security.*

KATA PENGANTAR

Puji Syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “Implementasi Kriptografi dan Steganografi Pada Citra Digital Menggunakan Algoritma *Rivest Shamir Adleman* (RSA) dan Metode *Discrete Cosine Transform* (DCT)”. Skripsi ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Penulis menyadari tanpa adanya dukungan dan kerja sama dari berbagai pihak, penulisan skripsi ini tidak akan dapat berjalan baik. Untuk itu, penulis ingin menyampaikan rasa terima kasih kepada:

1. Bapak Rudi Ariyanto, ST., MCs selaku Ketua Jurusan Teknologi Informasi.
2. Bapak Ir. Deddy Kusbianto P., M.MKom selaku Ketua Program Studi Teknik Informatika.
3. Bapak Dr.Eng. Rosa Andrie Asmara, ST, MT selaku Dosen Pembimbing Politeknik Negeri Malang Prodi Teknik Informatika.
4. Orang tua dan keluarga yang telah memberikan dukungan berupa doa dan dorongan semangat dalam proses penyusunan skripsi ini.
5. Teman-teman satu angkatan Teknik Informatika 2013 yang telah banyak memberikan dukungan untuk terselesaikannya skripsi ini.
6. Teman-teman satu kelompok PKL (Praktek Kerja Lapangan) di Dinas Komunikasi dan Informatika Kota Malang yang telah membantu penulis dan dukungan untuk perkembangan ilmu dan perkembangan diri penulis.
7. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Skripsi dari awal hingga akhir yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan skripsi ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan laporan ini. Semoga laporan ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, 8 Juni 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN PENGESAHAN.....	iii
ABSTRAK	v
<i>ABSTRACT</i>	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiii
DAFTAR LAMPIRAN.....	xiv
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 Penelitian.....	3
1.6. Sistematika Penulisan.....	3
BAB II LANDASAN TEORI	5
2.1. Steganografi.....	5
2.2. Kriptografi	6
2.3. Citra	6
2.4. Algoritma Rivest Shamir Adleman (RSA).....	8
2.5. Metode Discrete Cosine Transform.....	11
BAB III METODOLOGI PENELITIAN.....	14
3.1. Studi Literatur.....	14
3.2. Tahapan Penelitian	14
3.2.1. Pembangkitan Kunci RSA	14
3.2.2. Enkripsi Pesan.....	15
3.2.3. Perhitungan DCT	16
3.2.4. Penyisipan Pesan.....	18
3.2.5. Ekstraksi Pesan	19
3.2.6. Dekripsi Pesan.....	19
3.3. Data Set	20
3.3.1. Data Citra	20

3.3.2.	Data Teks	20
3.4.	Evaluasi	20
BAB IV ANALISIS DAN PERANCANGAN		23
4.1.	Analisis	23
4.1.1.	Deskripsi Umum Sistem	23
4.1.2.	Analisis Aktor	24
4.1.3.	Spesifikasi kebutuhan Perangkat Lunak	24
4.1.4.	Analisis Kebutuhan	24
4.1.5.	Use Case Diagram.....	25
4.1.6.	Scenario Diagram.....	27
4.1.7.	Diagram Alir Sistem	33
4.2.	Perhitungan Manual Algoritma RSA	41
4.2.1.	Pembangkitan Kunci RSA	41
4.2.2.	Enkripsi dengan RSA.....	41
4.2.3.	Dekripsi dengan RSA.....	41
4.3.	Perhitungan Manual Metode Discrete Cosine Transform	42
4.4.	Tampilan Antarmuka.....	44
4.4.1.	Menu Encoding	44
4.4.2.	Menu <i>Decoding</i>	46
BAB V IMPLEMENTASI.....		47
5.1.	Implementasi Sistem	47
5.1.1.	Pembangkitan Kunci RSA	47
5.1.2.	Enkripsi RSA	48
5.1.3.	Penyisipan Pesan.....	49
5.1.4.	Ekstraksi Pesan	50
5.1.5.	Dekripsi Pesan.....	52
5.2.	Implementasi Desain	53
5.2.1.	Halaman Encoding	53
5.2.2.	Halaman Decoding.....	55
BAB VI Pengujian dan Pembahasan.....		58
6.1.	Pengujian Sistem	58
6.2.	Pengujian Kesesuaian Data	60
6.3.	Analisis	64
6.3.1.	Analisis Terhadap Pesan	64

6.3.2.	Analisis Perbandingan Waktu Penyisipan dan Ekstraksi.....	65
6.3.3.	Analisis Perbandingan Nilai Piksel.....	68
6.3.4.	Analisis Perbandingan Penyisipan Pesan Antar Frekuensi.....	69
6.3.4.1.	Penyisipan Pesan pada Frekuensi Rendah.....	69
6.3.4.2.	Penyisipan Pesan pada Frekuensi Menengah.....	71
6.3.4.3.	Penyisipan Pesan Pada Frekuensi Tinggi.....	72
6.3.5.	Analisis Perbandingan Penyisipan Pesan pada Sub Blok.....	74
6.3.6.	Analisis Perbandingan Penyisipan Nilai pada Koefisien DCT.....	75
6.3.6.1.	Penyisipan pesan dengan nilai 10.....	75
6.3.6.2.	Penyisipan pesan dengan nilai 30.....	77
6.3.6.3.	Penyisipan pesan dengan nilai 50.....	78
6.3.7.	Analisis Ketahanan Citra Tersisip Pesan.....	79
BAB VII PENUTUP.....		82
7.1.	Kesimpulan.....	82
7.2.	Saran.....	82
DAFTAR PUSTAKA.....		83
DAFTAR LAMPIRAN.....		84

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Frekuensi DCT	11
Gambar 3.1 Diagram Alir Pembangkitan Kunci RSA	14
Gambar 3.2 Diagram Alir Enkripsi Pesan	15
Gambar 3.3 Diagram Alir Perhitungan DCT	16
Gambar 3.4 Diagram Alir Penyisipan Pesan	18
Gambar 3.5 Diagram Alir Ekstraksi Pesan	19
Gambar 3.6 Diagram Alir Dekripsi Pesan	19
Gambar 3.7 Contoh Data Set Citra	20
Gambar 4.1 Use Case Diagram	25
Gambar 4.2 Diagram Alir Keseluruhan	33
Gambar 4.3 Diagram Alir Pembangkitan Kunci RSA	34
Gambar 4.4 Diagram Alir Enkripsi Pesan	35
Gambar 4.5 Diagram Alir Penyisipan (a)	36
Gambar 4.6 Diagram Alir Penyisipan (b)	37
Gambar 4.7 Diagram Alir Ekstraksi (a)	38
Gambar 4.8 Diagram Alir Ekstraksi (b)	39
Gambar 4.9 Diagram Alir Dekripsi	40
Gambar 4.10 Direktur.jpeg	42
Gambar 4.11 Sub Blok pertama	42
Gambar 4.12 Desain Halaman Encoding	45
Gambar 4.13 Desain Halaman Decoding	46
Gambar 5.1 Halaman Encoding	53
Gambar 5.2 Open Image	54
Gambar 5.3 Generate RSA	54
Gambar 5.4 Hasil enkripsi	54
Gambar 5.5 Hasil penyisipan biner	55
Gambar 5.6 Halaman Decoding	56
Gambar 5.7 Pilih Citra	56
Gambar 5.8 Hasil Biner	57
Gambar 5.9 Hasil Ekstraksi	57
Gambar 5.10 Hasil Dekripsi	57
Gambar 6.1 Picture 1 512x512	60
Gambar 6.2 Picture 2 200x200	61
Gambar 6.3 Picture 3 400x400	61
Gambar 6.4 Picture 4 1024x768	62
Gambar 6.5 Koefisien DCT pada citra awal	68
Gambar 6.6 Koefisien DCT pada citra sisip	68
Gambar 6.7 Frekuensi Rendah	69
Gambar 6.8 Frekuensi Menengah	71
Gambar 6.9 Frekuensi Tinggi	72

DAFTAR TABEL

	Halaman
Tabel 3.1 Matriks i,j 2x2	17
Tabel 4.1 Scenario Use Case Encoding	27
Tabel 4.2 Scenario Use Case Input Citra	27
Tabel 4.3 Scenario Use Case Generate Nilai RSA	28
Tabel 4.4 Scenario Use Case Enkripsi Pesan	28
Tabel 4.5 Scenario Use Case Proses Penyisipan	29
Tabel 4.6 Scenario Use Case Menyimpan Nilai Dekripsi	29
Tabel 4.7 Scenario Use Case Simpan Gambar	30
Tabel 4.8 Scenario Use Case Decoding	30
Tabel 4.9 Scenario Use Case Input Citra RGB	31
Tabel 4.10 Scenario Use Case Ekstraksi Pesan	31
Tabel 4.11 Scenario Use Case Dekripsi Pesan	32
Tabel 4.12 Tabel nilai matriks sub blok pertama	43
Tabel 4.13 Tabel nilai koefisien DCT sub blok pertama	43
Tabel 4.14 Koefisien Sub blok ditambahkan nilai biner	43
Tabel 4.15 Invers DCT sub blok pertama	44
Tabel 6.1 Pengujian Blackbox	58
Tabel 6.2 Kesesuaian Data	63
Tabel 6.3 Perbandingan penyisipan pesan	64
Tabel 6.4 Perbandingan Penyisipan Biner	66
Tabel 6.5 Perbandingan Ekstraksi Biner	67
Tabel 6.6 Frekuensi Rendah	70
Tabel 6.7 Frekuensi Menengah	71
Tabel 6.8 Frekuensi Tinggi	73
Tabel 6.9 Perbandingan Frekuensi	74
Tabel 6.10 Perbandingan penyisipan biner	75
Tabel 6.11 Koefisien ditambah 10	76
Tabel 6.12 Koefisien ditambah 30	77
Tabel 6.13 Koefisien ditambah 50	78
Tabel 6.14 Pengujian Resize	80

DAFTAR LAMPIRAN

Lampiran 1 Kode Program Penyisipan	84
Lampiran 2 Kode Program Ekstraksi	86
Lampiran 3 Lembar Bimbingan Pembimbing I	88
Lampiran 4 Lembar Bimbingan Pembimbing II	89
Lampiran 5 Lembar Revisi Penguji I	90
Lampiran 6 Lembar Revisi Penguji II	91
Lampiran 7 Lembar Verifikasi	92
Lampiran 8 Profil Penulis	93

BAB I PENDAHULUAN

1.1. Latar Belakang

Pada era teknologi yang berkembang pesat ini, internet menjadi salah satu media untuk mendapatkan sebuah informasi yang memudahkan masyarakat untuk melakukan pertukaran informasi dan pengambilan informasi. Informasi yang didapatkan bisa berupa gambar, video, lagu ataupun sebuah sistem terpadu berbasis desktop maupun website. Keamanan dari sebuah data dan informasi menjadi salah satu hal penting yang harus diperhatikan, dikarenakan semakin berkembangnya teknologi informasi semakin berkembang pula teknik kejahatan komputer.

Dalam studi kasus Penyisipan Watermark Menggunakan Metode Discrete Cosine Transform Pada Citra Digital yang dilakukan oleh Agustina 2015[3], penelitian tersebut menggunakan metode Discrete Cosine Transform untuk menerapkan teknik *watermarking* pada citra digital. Penerapan metode Discrete Cosine Transform berhasil menyembunyikan citra *watermark* pada sebuah citra dengan mengubah citra dari domain spasial ke domain frekuensi. Berdasarkan evaluasi yang dilakukan pada penelitian tersebut terdapat sebuah kekurangan yaitu hasil ekstraksi dari citra watermark dapat dilihat secara langsung dan tidak bersifat rahasia.

Untuk menjaga kerahasiaan dari pesan yang disisipkan pada citra, pada penelitian Kriptografi Dan Steganografi Menggunakan Algoritma RSA dan Metode LSB yang dilakukan oleh Arifin 2013[4], digunakan metode RSA. Metode tersebut digunakan untuk menyisipkan pesan berupa teks pada citra jpeg. Penerapan metode RSA berhasil melakukan enkripsi pesan yang dimasukkan, sedangkan untuk penyisipan pesannya menggunakan metode LSB. Berdasarkan evaluasi yang dilakukan, terdapat sebuah kekurangan yaitu pesan yang dienkripsi hanya dijadikan sebuah bit dan ditambahkan kedalam citra tanpa mengubah domain dari citra.

Pada penelitian ini digunakan algoritma RSA dan metode Discrete Cosine Transform yang dapat memberikan keamanan yang lebih pada data yang akan disisipkan pada citra. Citra yang digunakan merupakan citra RGB dan untuk data teks yang akan dienkripsi digunakan file txt sebagai inputan. Terdapat dua proses

pada aplikasi yang akan dibuat yaitu proses embedding dan proses *Decoding*. Proses embedding dilakukan untuk mengenkripsi pesan, sedangkan proses *Decoding* dilakukan untuk mengekstrak pesan dari citra kemudian dilakukan proses dekripsi. Output dari aplikasi ini berupa informasi yang sudah disisipkan oleh pengirim pesan ke penerima pesan

Dengan dilakukannya penelitian ini diharapkan sebuah informasi yang bersifat rahasia bisa diberikan sebuah keamanan dimana pesan yang dikirimkan oleh pengirim dapat dienkripsi dengan menggunakan Algoritma RSA untuk kemudian disisipkan kedalam sebuah citra digital dengan menggunakan metode *Discrete Cosine Transform*.

1.2. Rumusan Masalah

Berdasarkan Latar Belakang diatas, rumusan masalah yang diambil yaitu:

- Bagaimana cara menyisipkan sebuah pesan pada citra digital menggunakan metode *Discrete Cosine Transform* (DCT).
- Bagaimana cara mengimplementasikan Algoritma RSA untuk mengenkripsi pesan yang akan disisipkan pada citra digital.

1.3. Batasan Masalah

Batasan masalah yang diangkat dalam skripsi ini dapat dipaparkan sebagai berikut:

- Aplikasi mendapat masukan berupa citra digital dengan format jpg dan file pesan berupa txt.
- Algoritma yang digunakan untuk enkripsi pesan adalah RSA (*Rivest Shamir Adleman*).
- Metode transformasi yang digunakan adalah Discrete Cosine Transform.
- Kualitas citra hasil watermarking diukur menggunakan PSNR (Peak Signal to Noise Ratio) dengan nilai ratio minimal 30 dB.

1.4. Tujuan

Tujuan dari pengerjaan skripsi ini yaitu:

- Untuk mengetahui cara menyisipkan sebuah pesan pada citra digital menggunakan Metode *Discrete Cosine Transform*.

- Untuk mengetahui cara mengenkripsi sebuah pesan menggunakan Algoritma RSA.

1.5. Manfaat Penelitian

Adapun Manfaat dari penulisan dan pembuatan Skripsi ini adalah:

- Diharapkan bisa menjadi referensi bagi yang ingin mengetahui proses pengolahan informasi untuk disisipkan kedalam citra digital.
- Dapat menjadi acuan untuk lebih lanjut bagi yang tertarik untuk mengembangkannya.

1.6. Sistematika Penulisan

BAB I PENDAHULUAN

Pada bab ini menjelaskan tentang latar belakang diadakannya penelitian ini dan yang menjadi dasar permasalahan, yang meliputi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi, sistematika penulisan, dan penjadwalan kegiatan penelitian.

BAB II LANDASAN TEORI

Bagian ini menjelaskan mengenai sumber dan referensi yang dijadikan acuan dalam pelaksanaan penelitian. Teori-teori tersebut diantaranya mengenai pengenalan dari steganografi, kriptografi, Metode *Discrete Cosine Transform*, Algoritma RSA, serta implementasi program.

BAB III METODOLOGI PENELITIAN

Bagian ini menjelaskan mengenai sumber dan referensi yang dijadikan acuan dalam pelaksanaan penelitian. Teori-teori tersebut diantaranya mengenai pengenalan dari steganografi, kriptografi, Metode *Discrete Cosine Transform*, Algoritma RSA, serta implementasi program.

BAB IV ANALISIS DAN PERANCANGAN

Bab ini menjelaskan tentang perencanaan dan pembuatan sistem secara keseluruhan dan analisa terhadap hasil dari data yang sudah didapat.

BAB V IMPLEMENTASI

Bab ini menjelaskan tentang bagaimana aplikasi dibuat dan berjalan berdasarkan analisa dan perancangan yang dilakukan sebelumnya. Dimana aplikasi diharapkan dapat melakukan penyisipan pesan dengan menggunakan Metode *Discrete Cosine Transform* dan Enkripsi RSA dengan baik.

BAB VI PENGUJIAN DAN PEMBAHASAN

Bab ini berisikan tentang tampilan yang diusulkan seperti form – form penginputan dan output dalam aplikasi yang mengimplementasikan Steganografi Pada Citra Digital Menggunakan Metode *Discrete Cosine Transform* dan Enkripsi RSA. Selain itu dilakukan juga pembahasan tentang analisa hasil yang diperoleh dari aplikasi yang dibuat.

BAB VII PENUTUP

Bab ini dibagi menjadi dua sub bab, kesimpulan yang menjawab permasalahan yang dihadapi dan saran yang berisikan solusi alternatif untuk permasalahan yang terjadi pada laporan akhir ini.

BAB II LANDASAN TEORI

2.1. Steganografi

Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (hiding message) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Kata steganografi berasal dari Bahasa Yunani yang berarti “tulisan tersembunyi” (*covered writing*). Steganografi membutuhkan dua properti: wadah penampung dan data rahasia yang akan disembunyikan.

Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks dan video. Data rahasia yang disembunyikan juga dapat berupa citra, suara, teks atau video. Steganografi dapat dipandang sebagai kelanjutan kriptografi. Jika pada kriptografi, data yang telah disandikan (*ciphertext*) tetap tersedia, maka dengan steganografi *ciphertext* dapat disembunyikan sehingga pihak ketiga tidak mengetahui keberadaannya.

Di negara-negara yang melakukan penyensoran informasi, steganografi sering digunakan untuk menyembunyikan pesan-pesan melalui citra (*images*), video atau suara (*audio*). Penyembunyian data rahasia ke dalam citra digital akan mengubah kualitas citra tersebut.

Kriteria yang harus diperhatikan dalam penyembunyian data adalah:

a) Fidelity

Mutu citra penampung tidak jauh berubah. Setelah penambahan data rahasia, citra hasil steganografi masih terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam citra tersebut terdapat data rahasia.

b) Robustness

Data yang disembunyikan harus tahan terhadap manipulasi yang dilakukan pada citra penampung (seperti perubahan kontras, penajaman, pemampatan, rotasi, perbesaran gambar, pemotongan (*cropping*), enkripsi, dan sebagainya). Bila pada citra dilakukan operasi pengolahan citra, maka data yang disembunyikan tidak rusak.

c) *Recovery*

Data yang disembunyikan harus dapat diungkapkan kembali (*recovery*). Karena tujuan steganografi adalah *data hiding*, maka sewaktu-waktu data rahasia di dalam citra penampung harus dapat diambil kembali untuk digunakan lebih lanjut [6].

2.2. Kriptografi

Kriptografi (*cryptography*) berasal dari Bahasa Yunani: “*cryptós*” artinya “*secret*” (rahasia), sedangkan “*gráphein*” artinya “*writing*” (tulisan). Jadi, kriptografi berarti “*secret writing*” (tulisan rahasia). Ada beberapa definisi kriptografi yang telah dikemukakan di dalam berbagai literatur. Definisi yang dipakai di dalam buku-buku yang lama (sebelum tahun 1980-an) menyatakan bahwa kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Definisi ini mungkin cocok pada masa lalu di mana kriptografi digunakan untuk keamanan komunikasi penting seperti komunikasi di kalangan militer, diplomat, dan mata-mata. Namun saat ini kriptografi lebih dari sekadar privasi, tetapi juga untuk tujuan data *integrity*, *authentication*, dan *non-repudiation*.

Di dalam kriptografi kita akan sering menemukan berbagai istilah atau terminologi. Beberapa istilah yang penting untuk diketahui diberikan di bawah ini.

- Pesan, Plainteks, dan Cipherteks Pesan (*message*)
- Pengirim dan penerima
- Enkripsi dan dekripsi
- Cipher dan kunci [1]

2.3. Citra

Secara fisik atau visual, sebuah citra adalah representasi dari informasi yang berkembang di dalamnya sehingga mata manusia dapat menganalisis dan menginterpretasikan informasi tersebut sesuai dengan tujuan yang diharapkan. Kandungan informasi citra dapat dibagi menjadi dua bagian yaitu informasi dasar dan informasi yang bersifat abstrak.

- Informasi dasar adalah informasi yang dapat diolah secara langsung tanpa membutuhkan bantuan tambahan pengetahuan khusus. Informasi dasar ini adalah berupa warna (*color*), bentuk (*shape*), dan tekstur (*texture*). Analisis terhadap informasi dasar citra dikenal dengan sebutan *low level image analysis*.
- Informasi abstrak adalah informasi yang tidak secara langsung dapat diolah kecuali dengan bantuan tambahan pengetahuan khusus. Contoh informasi yang bersifat abstrak adalah ekspresi wajah di dalam sebuah citra dapat menggambarkan perasaan seseorang.

Kedua informasi ini tidak dapat dianalisis dan dikenali oleh komputer kecuali menggabungkan informasi dasar dengan tambahan pengetahuan khusus.

Secara matematis, sebuah citra dapat didefinisikan sebagai fungsi dua dimensi $f(x,y)$ di mana x dan y adalah koordinat spasial (*plane*) dan f adalah nilai intensitas warna pada koordinat x dan y . Nilai x , y , dan f semuanya adalah nilai berhingga. Bila nilai-nilai ini bersifat kontinu maka citranya disebut **citra analog**, seperti yang ditampilkan pada layer monitor TV, komputer atau foto cetak. Bila nilai-nilai ini bersifat diskret maka citranya disebut **citra digital**, seperti yang tersimpan dalam memori komputer dan CD-ROM.

Citra Digital umumnya dua dimensi yang dinyatakan dalam bentuk matriks dengan jumlah elemen berhingga. Setiap elemen matriks citra memiliki posisi koordinat x dan y tertentu dan juga memiliki nilai. Secara umum, citra digital merupakan representasi piksel-piksel dalam ruang dua dimensi yang dinyatakan dalam matriks berukuran N baris dan M kolom seperti pada gambar di bawah.

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix} \quad (1)$$

Setiap elemen matriks citra disebut **piksel**. Nilai setiap piksel f pada posisi koordinat x dan y merepresentasikan intensitas warna dan dapat dikodekan dalam

24 bit untuk citra berwarna (yang memiliki tiga komponen yaitu **RGB**, R = merah, G = hijau, dan B = biru), 8 bit untuk citra gray level atau 1 bit untuk citra biner [7].

2.4. Algoritma Rivest Shamir Adleman (RSA)

Dari sekian banyak algoritma kriptografi yang pernah dibuat, algoritma yang paling populer adalah algoritma RSA. Algoritma RSA dibuat oleh 3 orang peneliti dari MIT (*Massachusetts Institute of Technology*) pada tahun 1976, yaitu: Ron (R)ivest, Adi (S)hamir, dan Leonard (A)dleman. Algoritma RSA mendasarkan proses enkripsi dan dekripsinya pada konsep bilangan prima dan aritmatika modulo. Kunci enkripsi maupun kunci dekripsi keduanya harus berupa bilangan bulat. Kunci enkripsi tidak dirahasiakan dan diketahui umum (sehingga dinamakan juga kunci publik), namun kunci untuk dekripsi bersifat rahasia. Kunci dekripsi dibangkitkan dari beberapa buah bilangan prima bersama-sama dengan kunci enkripsi. Untuk menemukan kunci dekripsi, suatu bilangan nonprima harus difaktorkan menjadi faktor primanya.

Dalam kenyataannya, memfaktorkan bilangan nonprima menjadi faktor primanya bukanlah pekerjaan yang mudah. Belum ada algoritma yang secara efisien yang dapat melakukan pemfaktoran tersebut. Semakin besar bilangan non primanya maka semakin sulit pula pemfaktorannya. Semakin sulit pemfaktorannya, semakin kuat pula algoritma RSA.

Algoritma RSA merupakan salah satu kriptografi asimetri, yakni jenis kriptografi yang menggunakan dua kunci yang berbeda: kunci publik (*public key*) dan kunci pribadi (*private key*). Dengan demikian, maka terdapat satu kunci, yakni kunci publik, yang dapat dikirimkan melalui saluran yang bebas, tanpa adanya suatu keamanan tertentu. Hal ini bertolak belakang dengan kriptografi simetri yang hanya menggunakan satu jenis kunci dan kunci tersebut harus terus terjaga keamanan serta kerahasiaannya. Dalam kriptografi asimetri, dua kunci tersebut diatur sedemikian sehingga memiliki hubungan dalam suatu persamaan aritmatika modulo.

Berikut ini merupakan langkah kerja dari Algoritma RSA:

1. Pilih dua bilangan prima p dan q secara acak, $p \neq q$. Bilangan ini harus cukup besar.

2. Hitung $n = p.q$.
3. Hitung $\phi(n) = (p-1).(q-1)$.
4. Pilih bilangan bulat (integer) e yang relative prima terhadap $\phi(n)$ dengan menggunakan algoritma *Euclidean* hingga $\gcd(e, \phi) = 1$.
5. Hitung d hingga $d e \equiv 1 \pmod{\phi}$. $d = e^{-1} \pmod{\phi(n)}$ atau digunakan algoritma *Extended Euclidean* [5].

Hasil dari pembangkitan kunci tersebut akan menghasilkan dua kunci yaitu kunci publik (e, n) dan kunci pribadi (d, n) . Untuk perhitungan kunci publik digunakan algoritma *Euclidean*, tujuan dari algoritma ini adalah menghitung gcd (r_0, r_1) untuk bilangan integer r_0 dan r_1 dengan merepresentasikan bilangan integer yang lebih besar yaitu r_0 dalam kelipatan r_1 dan sebuah bilangan sisa r_2 x r_1 yang kemudian direpresentasikan dalam kelipatan r_2 dan sebuah bilangan sisa r_3 . Proses ini dilakukan sampai tidak ada bilangan sisa r_3 yang dapat ditambahkan. Untuk perhitungan algoritma *Euclidean* dijelaskan pada (2):

$$\begin{aligned}
 r_0 &= q_1.r_1 + r_2 \rightarrow \gcd(r_0, r_1) = \gcd(r_1, r_2) \\
 r_1 &= q_2.r_2 + r_3 \rightarrow \gcd(r_1, r_2) = \gcd(r_2, r_3) \\
 r_2 &= q_3.r_3 + r_4 \rightarrow \gcd(r_2, r_3) = \gcd(r_3, r_4) \\
 &\vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad (2) \\
 r_{m-2} &= q_{m-1}.r_{m-1} + r_m \rightarrow \gcd(r_{m-2}, r_{m-1}) = \gcd(r_{m-1}, r_m) \\
 r_{m-1} &= q_m.r_m + 0 \rightarrow \gcd(r_{m-1}, r_m) = r_m = \gcd(r_0, r_1)
 \end{aligned}$$

Di mana pada rumus diatas q merupakan bilangan yang harus dikalikan hingga r_1 mendekati r_0 . Setelah proses algoritma *Euclidean* selesai dilakukan, ditemukan $\gcd(r_0, r_1) = r_m$. Proses perhitungan algoritma *Euclidean* dijelaskan sebagai berikut:

$$39 = 3 \cdot 12 + 3 \rightarrow \gcd(39, 12) = \gcd(12, 3)$$

$$12 = 4 \cdot 3 + 0 \rightarrow \gcd(12, 3) = 3 = \gcd(39, 12)$$

Maka, hasil $\gcd(39, 12) = 3$ [2]

Contoh untuk proses $\gcd(40, 7) = 1$ akan dijelaskan sebagai berikut:

$$40 = 5 \cdot 7 + 5 \rightarrow \gcd(40, 7) = \gcd(7, 5)$$

$$7 = 1 \cdot 5 + 2 \rightarrow \gcd(7,5) = \gcd(5,2)$$

$$5 = 2 \cdot 2 + 1 \rightarrow \gcd(5,2) = \gcd(2,1)$$

$$2 = 2 \cdot 1 + 0 \rightarrow \gcd(2,1) = 1 = \gcd(40,7)$$

Sedangkan untuk menemukan kunci pribadi dari algoritma RSA akan digunakan algoritma *Extended Euclidean*, dimana algoritma ini berbasis pada sebuah teori yang menggunakan dua nilai integer r_0 dan r_1 , dihitung $\gcd(r_0, r_1) = r_m = s \times r_0 + t \times r_1$ dari s dan t . Algoritma ini menjabarkan proses dari algoritma *Euclidean* dengan merepresentasikan setiap bilangan sisa yaitu r_j dalam r_0 dan r_1 . Untuk proses perhitungan algoritma ini dijelaskan pada (3):

$$\begin{aligned}
 r_0 &= q_1 \cdot r_1 + r_2 \rightarrow r_2 = r_0 - q_1 \cdot r_1 = s_2 \cdot r_0 + t_2 \cdot r_1 \\
 r_1 &= q_2 \cdot r_2 + r_3 \rightarrow r_3 = r_1 - q_2 \cdot r_2 = r_1 - q_2(r_0 - q_1 \cdot r_1) \\
 &= -q_2 \cdot r_0 + (1 + q_1 \cdot q_2) \cdot r_1 \\
 &= s_3 \cdot r_0 + t_3 \cdot r_1 \\
 &\vdots \\
 r_{i-2} &= q_i \cdot r_{i-1} + r_i \rightarrow r_i = r_{i-2} - q_i \cdot r_{i-1} = s_i \cdot r_0 + t_i \cdot r_1 \\
 r_{i-1} &= q_{i+1} \cdot r_i + r_{i+1} \rightarrow r_{i+1} = r_{i-1} - q_{i+1} \cdot r_i = s_{i+1} \cdot r_0 + t_{i+1} \cdot r_1 \\
 r_i &= q_{i+2} \cdot r_{i+1} + r_{i+2} \rightarrow r_{i+2} = r_i - q_{i+2} \cdot r_{i+1} = s_{i+2} \cdot r_0 + t_{i+2} \cdot r_1 \\
 &\vdots \\
 r_{m-2} &= q_m \cdot r_{m-1} + r_m \rightarrow r_m = r_{m-2} - q_m \cdot r_{m-1} = s_m \cdot r_0 + t_m \cdot r_1 \\
 r_{m-1} &= q_{m+1} \cdot r_m + 0 \rightarrow s = s_m, t = t_m
 \end{aligned} \tag{3}$$

Setelah proses algoritma *Extended Euclidean* selesai, akan menghasilkan $\gcd(r_0, r_1) = r_m = s_m \times r_0 + t_m \times r_1 = s \times r_0 + t \times r_1$. Untuk melakukan proses algoritma ini kedua bilangan yang dimasukkan harus relatif prima dimana nilai $\gcd(r_0, r_1)$ harus bernilai 1. Jika gcd yang dihasilkan 1 maka rumusnya adalah $s \times m + t \times a = \gcd(m, a) = 1$ yang disederhanakan menjadi (4):

$$\begin{aligned}
 s \cdot m + t \cdot a &= 1 \\
 t \cdot a &= (-s) \cdot m + 1
 \end{aligned} \tag{4}$$

Hasil dari nilai t mungkin negatif, untuk mengubah menjadi nilai positif maka akan ditambahkan nilai modulus dari m ke t sampai $t > 0$. Contoh perhitungan *Extended Euclidean* dari $\gcd(40, 7)$ akan dijelaskan sebagai berikut:

$$40 = 5 \cdot 7 + 5 \rightarrow t_2 = t_0 - q_1 \cdot t_1 = 0 - 5 \cdot 1 = -5$$

$$7 = 1 \cdot 5 + 2 \rightarrow t_3 = t_2 - q_2 \cdot t_2 = 1 - 1 \cdot 5 = 6$$

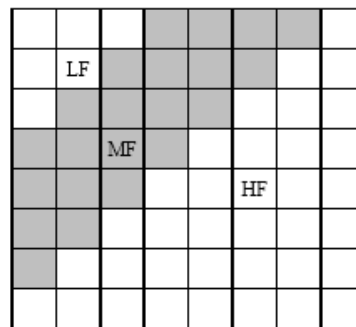
$$5 = 2 \cdot 2 + 1 \rightarrow t_4 = t_3 - q_3 \cdot t_3 = -5 - 2 \cdot 6 = -17$$

$$2 = 2 \cdot 1 + 0$$

Maka $t = -17 \equiv \mathbf{23} \bmod 40 \equiv 7^{-1} \bmod 40$, didapatkan kunci pribadi bernilai 23 [2].

2.5. Metode Discrete Cosine Transform

Discrete Cosine Transform (DCT) pertama kali diperkenalkan oleh Ahmed, Natarajan dan Rao pada tahun 1974 dalam makalahnya yang berjudul “*On image processing and a Discrete Cosine Transform*”. Proses *Discrete Cosine Transform* (DCT) merupakan proses transformasi data dari spasial domain ke domain frekuensi, penggunaan DCT pada pengolahan citra dilakukan dengan membagi citra ke dalam sub blok berukuran standar 8x8 piksel. Dimana sub blok 8x8 piksel tersebut akan menghasilkan 64 koefisien yang terdiri dari 1 koefisien DC (*zero frequencies*) yang terletak pada pojok kiri atas dan 63 koefisien AC yang terdiri dari 3 frekuensi. Dalam sebuah citra, informasi utama yang dikandungnya lebih banyak terdapat pada frekuensi rendah (LF) dan menengah (MF), sedangkan informasi frekuensi tinggi (HF) merupakan informasi detail. Gambar 2.1 merupakan ilustrasi dari frekuensi yang terdapat pada DCT:



Gambar 2.1 Frekuensi DCT

Ini sebagaimana sifat visual mata manusia yang tidak begitu sensitif pada perubahan informasi detail tetapi sangat sensitif pada perubahan informasi global. Hal ini menunjukkan bahwa menghilangkan informasi detail citra tidak akan menghilangkan makna informasi yang terkandung dalam citra itu. Untuk

menghilangkan informasi detail pada frekuensi tinggi ini dapat dilakukan melalui proses DCT.

Proses DCT 2-D ini dilakukan secara terpisah, dimulai dari proses DCT arah-x dan kemudian hasilnya dilakukan proses DCT arah-y. Secara matematis, proses DCT sebuah citra adalah jumlah perkalian antara fungsi cosinus diskrit dan fungsi citra numerik 2-D $f(x,y)$ yang diberikan pada (5) dan (6):

$$f[u, j] = \frac{\sqrt{2}}{n} C_u \sum_{x=0}^{N-1} \cos \frac{(2j+1)i\pi}{2N} \cdot f[x, y] \quad (5)$$

$$f[u, v] = \frac{\sqrt{2}}{n} C_y \sum_{x=0}^{N-1} f[u, j] \cdot \cos \frac{(2i+1)j\pi}{2N} \quad (6)$$

$$\text{di mana } C_u, C_y = \begin{cases} \frac{\sqrt{1}}{2} & \text{Bila } u, v = 0 \\ 1 & \text{Bila } u, v > 0 \end{cases}$$

Pada persamaan 6 merupakan hasil dari proses DCT-2D yang merepresentasikan domain frekuensi dari citra, di mana x,y adalah koordinat diskret dalam domain spasial citra dan u,v adalah koordinat diskret dalam domain frekuensi. N menunjukkan ukuran dari citra, di mana dalam proses ini harus merupakan persegi-empat.

Selanjutnya, persamaan 6 ini dapat direpresentasikan dalam bentuk perkalian matriks seperti yang ditunjukkan pada (7) berikut:

$$f[u, v] = (DC[i, j] \cdot f[x, y] \cdot DC[j, i]) \quad (7)$$

Di mana

$DC[i, j]$ = matriks baru dari satu sub blok

$f[x, y]$ = matriks nilai asli dari satu sub blok

$DC[j, i]$ = matriks transpose dari matriks $DC[i, j]$

Yang berlaku umum untuk ukuran matriks persegi-empat dan merupakan perpangkatan dua. $DC[i, j]$ dan $DC[j, i]$ secara berurutan adalah matriks cosinus dua arah sumbu-x dan arah sumbu-y, yang dapat dihitung dengan menggunakan persamaan 8 dan 9 berikut:

$$DC[i, j] = \sqrt{\frac{2}{N}} \cos \frac{(2j+1)i\pi}{2N} \quad (8)$$

$$DC[j, i] = \sqrt{\frac{2}{N}} \cos \frac{(2i+1)j\pi}{2N} \quad (9)$$

Dengan ketentuan sebagai berikut:

$$[i, j] = \begin{cases} \frac{1}{\sqrt{N}} & \text{jika } i = 0 \\ \sqrt{\frac{2}{N}} \cos \frac{(2j+1)i\pi}{2N} & \text{jika } i \neq 0 \end{cases} \quad (10)$$

di mana

N = Jumlah baris matriks

Dan i, j = indeks matriks

Untuk mendapatkan kembali citra asli dari koefisien DCT berdomain frekuensi dilakukan proses balik dengan DCT invers (iDCT). Secara matematis, proses ini diberikan pada (8) dan (9), serta untuk proses dalam bentuk perkalian matriks pada (11):

$$f[x, y] = DC[i, j] \cdot (f[u, v] \cdot DC[j, i]) \quad (11)$$

Di mana

$DC[i, j]$ = matriks baru dari satu sub blok

$f[u, v]$ = matriks nilai dari koefisien DCT

$DC[j, i]$ = matriks transpose dari matriks $DC[i, j]$

Untuk mendapatkan kembali citra original dari koefisien DCT domain frekuensi, dilakukan proses balik yang disebut dengan DCT invers (iDCT) menggunakan persamaan 11. Proses DCT invers ini diawali oleh perkalian matriks antara koefisien DCT atau $f(u, v)$ dan matriks cosinus invers $DC[i, j]$ kemudian dikalikan dengan matriks cosinus invers $DC[j, i]$ [7].

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan langkah – langkah yang dilakukan untuk membuat Aplikasi yang mengimplementasikan steganografi pada citra digital dengan menggunakan metode *Discrete Cosine Transform* dan Algoritma RSA.

Langkah yang diperlukan antara lain:

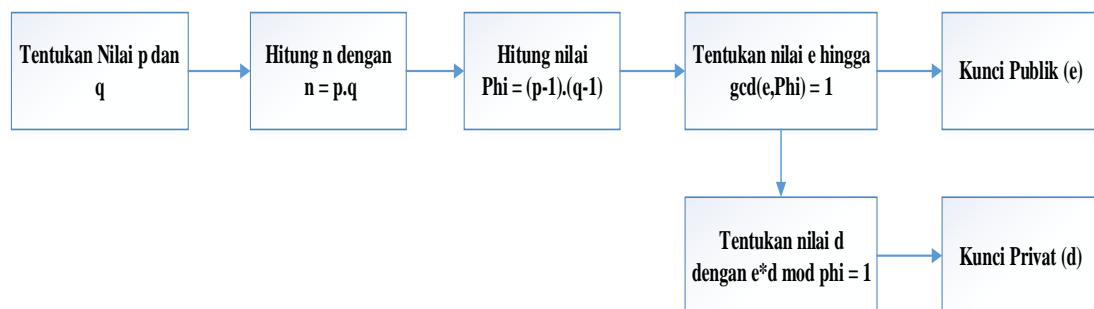
3.1. Studi Literatur

Pada tahap ini penelitian dilakukan dengan mempelajari berbagai literatur melalui pengumpulan dokumen, referensi, buku, dan artikel dari internet yang diperlukan untuk merancang dan mengimplementasikan sistem yang menggunakan algoritma RSA untuk enkripsi data, serta metode *Discrete Cosine Transform* (DCT) yang digunakan untuk proses transformasi citra.

3.2. Tahapan Penelitian

Pada bagian tahapan penelitian, dijelaskan kerangka kerja dari proses penyisipan dan ekstraksi pesan pada citra yang digunakan pada skripsi. Tahapan dalam penelitian ini yaitu:

3.2.1. Pembangkitan Kunci RSA



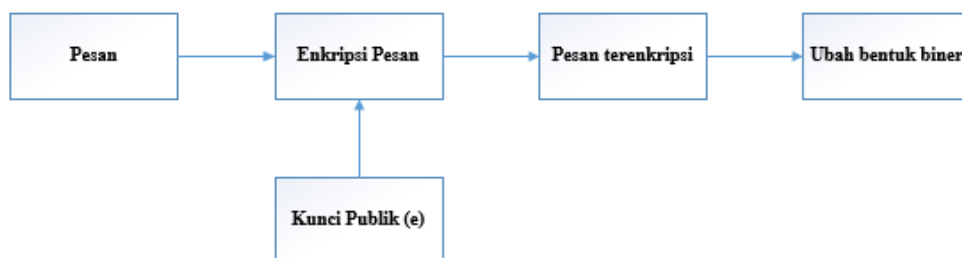
Gambar 3.1 Diagram Alir Pembangkitan Kunci RSA

Mengacu pada diagram alir diatas, langkah - langkah yang dilakukan adalah sebagai berikut:

- Menentukan nilai p dan q dimana kedua bilangan tersebut harus bilangan prima.
- Hitung nilai n dengan mengalikan dua bilangan p dan q.
- Hitung nilai phi dengan rumus $\phi = (p-1).(q-1)$

- d. Menentukan nilai e dimana $1 < e < \phi$, kemudian dilakukan fungsi GCD *Greatest Common Divisor* juga dikenal sebagai FPB (Faktor Persekutuan terBesar), mencari nilai faktor pembagi bersama yang paling besar dari dua nilai masukkan. GCD yang dihasilkan dari kedua bilangan harus bernilai 1.
- e. Menentukan nilai d dengan rumus $d = e^{-1} \bmod \phi$ atau menggunakan rumus $e \cdot d \bmod \phi = 1$.
- f. Proses pembangkitan kunci selesai dan didapatkan dua kunci yaitu kunci public (e) untuk mengenkripsi pesan dan kunci pribadi (d) untuk mendekripsi pesan.

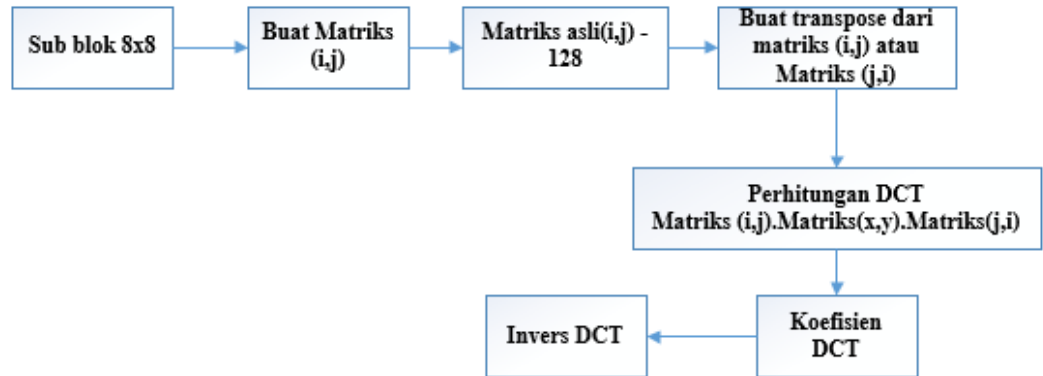
3.2.2. Enkripsi Pesan



Gambar 3.2 Diagram Alir Enkripsi Pesan

Mengacu pada diagram alir diatas, tahapan awal yang harus dilakukan ialah memasukkan sebuah pesan berupa teks dengan format .txt, kemudian dilakukan proses enkripsi pada pesan menggunakan kunci publik (e) yang dihasilkan dari pembangkitan kunci algoritma RSA. Setelah pesan terenkripsi diubah ke bentuk biner untuk selanjutnya akan disisipkan pada citra.

3.2.3. Perhitungan DCT



Gambar 3.3 Diagram Alir Perhitungan DCT

Mengacu pada diagram alir diatas, diberikan contoh dari perhitungan koefisien nilai DCT sebagai berikut:

- Citra masukkan dibagi menjadi sub blok berukuran 8x8 piksel.
- Setelah dibagi menjadi sub blok berukuran 8x8 dilakukan perhitungan untuk matriks [i,j] dengan ketentuan yang diberikan pada (12):

$$(i,j) = \begin{cases} \frac{1}{\sqrt{N}} & \text{jika } i = 0 \\ \sqrt{\frac{2}{N} \cos \frac{(2j+1)i\pi}{2N}} & \text{jika } i \neq 0 \end{cases} \quad (12)$$

di mana

N = Jumlah baris matriks

Dan i,j = indeks matriks

- Nilai matriks asli akan dikurangi dengan 128 karena DCT bekerja pada rentang nilai piksel -128 sampai 127 sesuai dengan ketentuan pengolahan citra digital.
- Setelah didapatkan matriks [i,j] dilakukan transpose nilai dari matriks [i,j] atau matriks [j,i].
- Untuk menghitung koefisien DCT dilakukan perkalian matriks dengan rumus $DCT = matriks[i,j].matriks\ asli[x,y].matriks[j,i]$ (13)
- Setelah dilakukan perkalian maka akan didapatkan koefisien nilai DCT.
- Lakukan Invers DCT dengan menggunakan (14)

$$iDCT = matriks[i,j].nilai\ DCT.matriks[j,i] \quad (14)$$

Berikut merupakan contoh perhitungan DCT menggunakan matriks dengan ordo 2 x 2 yang nilai piksel awal ditunjukkan oleh matriks asli, nilai piksel ini akan ditransformasikan ke domain frekuensi dengan menggunakan DCT.

$$\text{matriks asli } [x, y] = \begin{bmatrix} 217 & 223 \\ 205 & 231 \end{bmatrix}$$

Matriks asli dikurangi dengan 128 karena DCT bekerja pada rentang nilai piksel -128 sampai 127.

$$\text{matriks asli } [x, y] = \begin{bmatrix} 89 & 95 \\ 77 & 103 \end{bmatrix}$$

Buat matriks[i,j] dengan ordo 2x2, dapat dihitung dengan persamaan berikut:

Tabel 3.1 Matriks i,j 2x2

$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
$\sqrt{\frac{2}{2}} \cos \frac{(2.0 + 1)1\pi}{2.2}$	$\sqrt{\frac{2}{2}} \cos \frac{(2.1 + 1)1\pi}{2.2}$

Selanjutnya dilakukan perhitungan koefisien DCT pada matriks ordo 2x2

$$\text{matriks}[i, j] = \begin{bmatrix} 0,7071 & 0,7071 \\ 0,7071 & -0,7071 \end{bmatrix}$$

$$\text{matriks}[j, i] = \begin{bmatrix} 0,7071 & 0,7071 \\ 0,7071 & -0,7071 \end{bmatrix}$$

Untuk mendapatkan koefisien DCT dilakukan perkalian matriks dengan rumus (15):

$$DCT = \text{matriks}[i, j] \cdot \text{matriks asli}[x, y] \cdot \text{matriks}[j, i] \quad (15)$$

maka didapatkan nilai matriks dari DCT:

$$DCT = \begin{bmatrix} 181,996 & -15,99 \\ 1,99 & 9,99 \end{bmatrix}$$

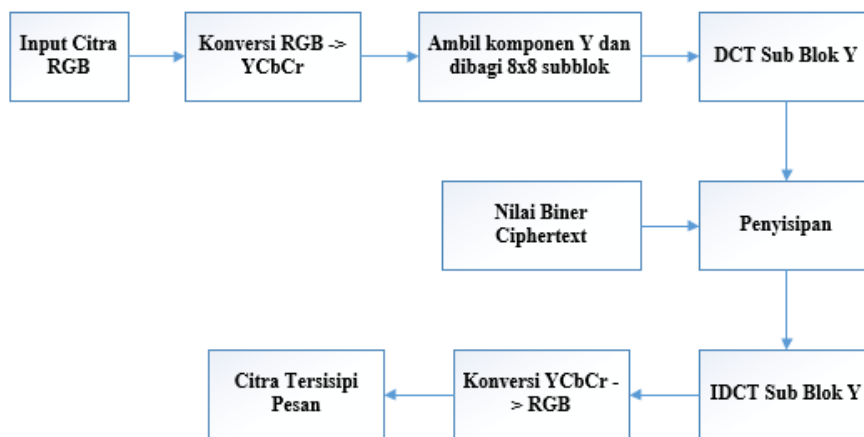
Berikutnya dilakukan proses invers DCT dengan rumus (16)

$$iDCT = matriks[i,j].nilai\ DCT.matriks[j,i] \quad (16)$$

dan didapatkan nilai matriks seperti berikut:

$$iDCT = \begin{bmatrix} 88,99 & 94,99 \\ 76,99 & 102,99 \end{bmatrix}$$

3.2.4. Penyisipan Pesan

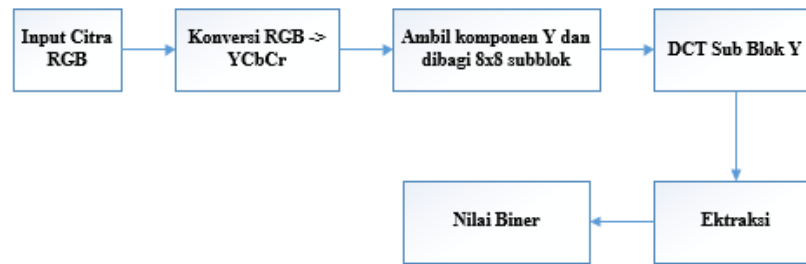


Gambar 3.4 Diagram Alir Penyisipan Pesan

Mengacu pada blok diagram diatas, langkah - langkah yang dilakukan adalah sebagai berikut:

- Memasukkan citra RGB dengan format .JPG kedalam sistem.
- Kemudian dilakukan pengkonversian citra RGB ke citra YCbCr.
- Citra YCbCr diambil komponen Y nya dan dirubah ke subblok berukuran 8x8 piksel.
- Selanjutnya dilakukakn proses DCT pada sub blok Y dan didapatkan koefisien DCT nya.
- Penyisipan dilakukan dengan cara mengambil 1 nilai biner dari *ciphertext* kemudian disisipkan kedalam Koefisien DCT.
- Setelah menyisipkan nilai biner kedalam citra, dilakukan proses invers DCT pada komponen Y kemudian digabungkan dengan Cb dan Cr untuk mendapatkan citra YCbCr yang tersisip oleh pesan.
- Konversikan kembali citra YCbCr ke citra RGB sehingga citra akan kembali kedalam bentuk citra RGB yang sudah tersisip oleh pesan.

3.2.5. Ekstraksi Pesan

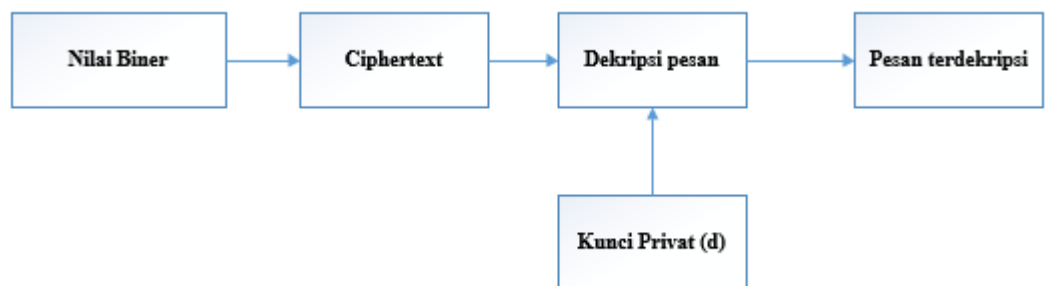


Gambar 3.5 Diagram Alir Ekstraksi Pesan

Mengacu pada diagram alir diatas, langkah - langkah yang dilakukan adalah sebagai berikut:

- Memasukan citra RGB dengan format .JPG yang telah tersisip oleh pesan
- Kemudian dilakukan pengkonversian citra RGB ke citra YCbCr.
- Citra YCbCr diambil komponen Y nya dan dirubah ke subblok berukuran 8x8 piksel.
- Selanjutnya dilakukakn proses DCT pada sub blok Y dan didapatkan koefisien DCT nya.
- Ekstraksi dilakukan dengan cara mengambil nilai biner dari koefisien DCT yang telah mengalami perubahan.
- Didapatkan nilai biner dari *ciphertext*.

3.2.6. Dekripsi Pesan



Gambar 3.6 Diagram Alir Dekripsi Pesan

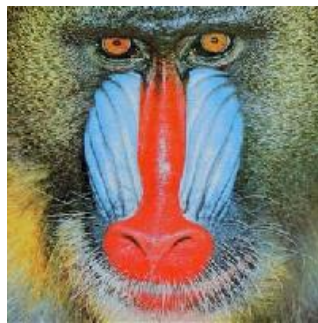
Mengacu pada diagram alir diatas, nilai biner yang dihasilkan dari proses ekstraksi dirubah ke bentuk *ciphertext*. Kemudian dilakukan pendekripsian pesan dengan menggunakan kunci pribadi (d) yang dihasilkan dari perhitungan algoritma RSA. Setelah dilakukan proses dekripsi didapatkan data berupa pesan.

3.3. Data Set

Kebutuhan aplikasi didefinisikan sesuai dengan sasaran yang ingin dicapai. Adapun analisis tersebut menyangkut tentang masukan (input) dan keluaran (ouput) dari aplikasi yang akan dibuat. Adapun data-data yang menjadi masukan bagi aplikasi ini yaitu:

3.3.1. Data Citra

Data citra yang digunakan berupa citra RGB dengan format .JPG berukuran 200x200 piksel, 400x400 piksel, 512x512 piksel dan 1024x768 piksel yang digunakan sebagai citra penampung pesan. Citra yang diambil didapatkan dari mesin pencarian Google dengan menuliskan ukuran citra terlebih dahulu, contoh pengambilan untuk citra 200x200 piksel digunakan kata kunci berikut: “200x200 Image”. Maka dari hasil pencarian tersebut, didapatkan citra RGB dengan format .JPG dengan ukuran 200x200 piksel.



Gambar 3.7 Contoh Data Set Citra

3.3.2. Data Teks

Data yang akan disisipkan kedalam citra berupa teks yang dengan format .txt yang dapat diketik secara langsung oleh pengguna pada saat sistem dijalankan. Teks tersebut kemudian dienkripsi dan dijadikan kedalam bentuk biner untuk kemudian disisipkan kedalam citra.

3.4. Evaluasi

Pengujian dilakukan untuk menjamin dan memastikan bahwa sistem yang dirancang dapat berjalan seperti yang diharapkan. Strategi pengujian perangkat lunak yang digunakan yaitu dengan menggunakan pengujian visual.

Pada pengujian visual ini digunakan perhitungan MSE (*Mean Square Error*) dan perhitungan PSNR (*Peak Signal to Noise Ratio*). MSE adalah nilai error kuadrat

rata-rata antara citra asli dengan citra pembanding. PSNR adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. Hasil yang diinginkan pada pengujian MSE adalah serendah mungkin. Sedangkan nilai PSNR diharapkan memiliki nilai setinggi mungkin. Berikut ini merupakan rumus perhitungan MSE dan PSNR seperti pada (17) dan (18):

$$MSE = \frac{1}{MN} \sum_{X=1}^M \sum_{Y=1}^N (S_{xy} - C_{xy})^2 \quad (17)$$

Di mana:

M = ukuran panjang citra

N = lebar citra

S_{xy} = nilai piksel pada koordinat (x,y) pada citra awal

C_{xy} = nilai piksel pada koordinat (x,y) pada citra sisip

$$PSNR = 10 \log_{10} \left(\frac{C^2_{max}}{MSE} \right) \quad (18)$$

Di mana

C^2_{max} = Nilai maksimum pada koordinat citra

Contoh untuk perhitungan MSE dan PSNR menggunakan matriks berordo 2x2, matriks X merupakan matriks yang mempunyai nilai piksel citra asli, dan matriks Y merupakan matriks yang mempunyai nilai piksel dari citra asli yang sudah dilakukan proses transformasi.

$$X = \begin{bmatrix} 232 & 235 \\ 238 & 242 \end{bmatrix} \quad Y = \begin{bmatrix} 216 & 221 \\ 226 & 233 \end{bmatrix}$$

MSE

$$= \frac{(232 - 216)^2 + (235 - 221)^2 + (238 - 226)^2 + (242 - 233)^2}{4}$$

$$= \frac{677}{4}$$

$$= 169,25$$

$$PSNR = 10 \log_{10} \frac{255^2}{169,25} = 25,8$$

BAB IV ANALISIS DAN PERANCANGAN

4.1. Analisis

Pada tahap ini dilakukan analisa kebutuhan dan keperluan dasar yang akan digunakan dalam pembuatan aplikasi. Disamping itu dilakukan juga analisa terhadap kebutuhan perangkat lunak yang akan dibangun sehingga diperoleh gambaran umumnya.

4.1.1. Deskripsi Umum Sistem

Pada aplikasi ini terdapat dua proses yaitu proses enkripsi dan dekripsi data berupa teks dengan format .txt serta proses penyisipan dan ekstraksi data pada citra RGB dengan format .JPG. Untuk proses enkripsi dan dekripsi, digunakan algoritma RSA dan untuk proses penyisipan dan ekstraksi digunakan metode *Discrete Cosine Transform* (DCT).

Implementasi dari algoritma RSA pada aplikasi yang dibangun menggunakan kunci asimetris, di mana kunci asimetris ini menggunakan kunci yang berbeda pada saat melakukan enkripsi dan dekripsi sebuah pesan. Jika dimasukkan kunci publik untuk mendekripsi sebuah pesan, pesan yang terenkripsi tidak akan bisa kembali ke bentuk awalnya. Untuk itu digunakan kunci pribadi yang berfungsi untuk mendekripsi sebuah pesan yang telah dienkripsi.

Implementasi Metode *Discrete Cosine Transform* akan digunakan untuk merubah citra digital dari domain spasial ke domain frekuensi yang merubah citra digital kedalam tiga frekuensi yaitu rendah, medium, dan tinggi. Pada metode ini sebelum dilakukan penyisipan pesan yang telah terenkripsi, citra RGB dibagi menjadi subblok berukuran 8 x 8. Kemudian dilakukan proses DCT pada setiap subbloknnya, setelah itu pesan yang telah dienkripsi disisipkan pada koefisien DCT. Kemudian dilakukan proses inverse DCT agar citra dapat kembali kedalam bentuk citra RGB.

4.1.2. Analisis Aktor

Pada sistem yang dibuat terdapat dua aktor diantaranya:

- a. Pengirim Pesan, mempunyai tugas yaitu:
 - Memasukkan pesan dan memasukkan citra digital.
 - Enkripsi pesan dan menyisipkan pesan kedalam citra digital.
- b. Penerima Pesan, mempunyai dua tugas yaitu:
 - Memasukkan citra yang telah tersisip oleh pesan.
 - Ekstraksi pesan yang ada dalam citra.
 - Dekripsi pesan yang telah terenkripsi.

4.1.3. Spesifikasi kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan penulis dalam pembuatan aplikasi ini adalah sebagai berikut:

- a. Microsoft Windows 10
- b. Microsoft Visual VB.Net
- c. Microsoft .NET Framework 4.5
- d. Microsoft Visio 2016

Sedangkan untuk spesifikasi dari perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut:

- a. Intel Core i5 4210U
- b. Ram 4 Gb
- c. Video Card Intel HD Graphics 4000

4.1.4. Analisis Kebutuhan

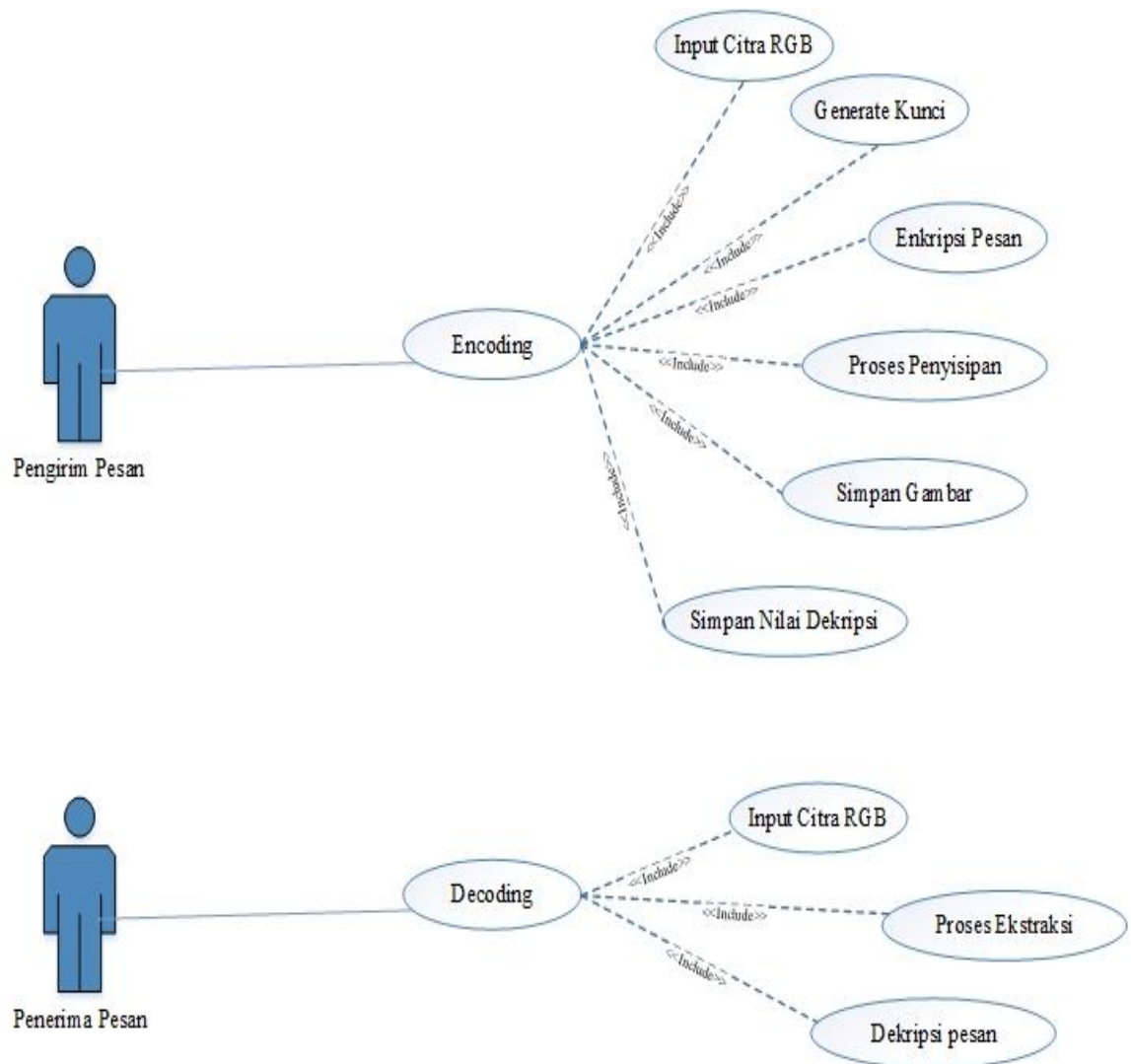
Pada tahap ini semua kebutuhan aplikasi didefinisikan sesuai dengan sasaran yang ingin dicapai. Analisis tersebut menyangkut tentang masukan (input) dan keluaran (ouput) dari aplikasi yang akan dibuat.

Adapun data-data yang menjadi masukan bagi aplikasi ini merupakan citra RGB dengan format .JPG berukuran 1024x768 piksel, 800x800 piksel, 512x512 piksel, dan 200x200 piksel sebagai citra penampung, sedangkan untuk data yang disisipkan berupa teks dengan format .txt. Hasil yang diharapkan sebagai keluaran

dari aplikasi ini merupakan file citra RGB yang berekstensi .JPG, yang telah disisipi watermark dengan menggunakan metode *Discrete Cosine Transform*

4.1.5. Use Case Diagram

Sebelum membangun aplikasi perlu dibuat *Use Case Diagram* untuk mengetahui aksi yang dapat dilakukan oleh pengguna. Berikut *Use Case Diagram* yang menerangkan jalannya program:



Gambar 4.1 Use Case Diagram

Deskripsi Use Case:

1. Pengirim Pesan

Pada aktor pengirim pesan dapat melakukan proses *Encoding* yang terdiri dari beberapa fitur yaitu:

- a. Input Citra RGB, pengirim pesan dapat memasukkan citra RGB dengan format .JPG kedalam sistem.
- b. *Generate Kunci*, pengirim pesan dapat menekan tombol “*Generate Nilai RSA*” untuk mendapatkan kunci RSA yang digunakan untuk mengenkripsi dan mendekripsi pesan.
- c. Masukan Teks, pengirim pesan dapat memasukan teks untuk kemudian dilakukan proses enkripsi.
- d. Enkripsi Pesan, pengirim pesan dapat menekan tombol “Enkripsi Pesan” untuk mendapatkan nilai biner yang akan disisipkan ke citra.
- e. Proses Penyisipan, pengirim pesan dapat menekan tombol “Proses biner” untuk melakukan proses penyisipan kedalam citra.
- f. Simpan Gambar, pengirim pesan dapat menyimpan citra yang sudah tersisip pesan dengan menekan tombol “Simpan Gambar”

2. Penerima Pesan

Pada aktor pengirim pesan dapat melakukan proses *Decoding* yang terdiri dari beberapa fitur yaitu:

- a. Input Citra RGB, pennerima pesan dapat memasukkan citra RGB dengan format .JPG yang telah tersisip pesan kedalam sistem.
- b. Proses Ekstraksi, penerima pesan dapat menekan tombol “Proses” untuk melakukan proses ekstraksi nilai biner sesuai dengan panjang biner yang dimasukkan. Setelah dilakukan proses ekstraksi penerima pesan dapat melihat nilai biner dan nilai *ciphertext*.
- c. Dekripsi Pesan, penerima pesan dapat menekan tombol “Dekripsi” untuk melakukan proses dekripsi pesan.

4.1.6. Scenario Diagram

Berdasarkan *Use Case Diagram* diatas dijabarkan *Scenario Diagram* sebagai berikut:

1. Use Case: *Encoding*

Aktor: Pengirim Pesan

Tujuan: Mengawali proses enkripsi dan penyisipan data

Tabel 4.1 Scenario Use Case Encoding

Aktor	Sistem
1. Pengirim pesan membuka aplikasi dan memilih halaman <i>Encoding</i>	
	2. Sistem Menampilkan Halaman <i>Encoding</i>
3. Pengirim dapat melakukan proses enkripsi dan penyisipan pesan	

2. Use Case: Input Citra RGB

Aktor: Pengirim Pesan

Tujuan: Memasukkan citra RGB berformat .JPG kedalam sistem

Tabel 4.2 Scenario Use Case Input Citra

Aktor	Sistem
1. Pengirim pesan memilih tombol "Open Image"	
	2. Sistem Menampilkan Halaman pilih citra
3. Pengirim pesan memilih citra RGB dengan format .JPG	
	4. Sistem menampilkan citra RGB pada " <i>Picture Box Citra Asli</i> "

3. Use Case: *Generate Kunci RSA*

Aktor: Pengirim Pesan

Tujuan: Mendapatkan nilai RSA untuk proses enkripsi dan dekripsi pesan teks

Tabel 4.3 Scenario Use Case Generate Nilai RSA

Aktor	Sistem
1. Pengirim pesan memilih tombol " <i>Generate Kunci RSA</i> "	
	2. Sistem menampilkan nilai RSA berupa nilai p,q,n,d,e,dan phi secara acak
3. Pengirim pesan dapat mengetahui nilai RSA yang akan digunakan untuk proses enkripsi dan dekripsi	

4. Use Case: Enkripsi Pesan

Aktor: Pengirim Pesan

Tujuan: Memasukkan data berupa teks dan melakukan proses enkripsi

Tabel 4.4 Scenario Use Case Enkripsi Pesan

Aktor	Sistem
1. Pengirim pesan memasukkan data berupa teks pada " <i>TextBox</i> "	
2. Pengirim pesan menekan tombol "Enkripsi"	
	3. Sistem melakukan proses enkripsi pada pesan yang telah dimasukkan pengirim pesan
	4. Sistem menampilkan kotak dialog "Pesan Terenkripsi!"

	5. Sistem menampilkan hasil dari enkripsi pesan berupa " <i>Ciphertext</i> " dan nilai biner " <i>Ciphertext</i> "
--	--

5. Use Case: Proses Penyisipan

Aktor: Pengirim Pesan

Tujuan: Menyisipkan data biner kedalam citra

Tabel 4.5 Scenario Use Case Proses Penyisipan

Aktor	Sistem
1. Pengirim pesan menekan tombol "Proses Binary"	
	2. Sistem menyisipkan nilai biner dari " <i>Ciphertext</i> " kedalam citra dengan proses DCT
	3. Sistem menampilkan kotak dialog "Pesan Tersisipkan!"
	4. Sistem menampilkan citra RGB yang telah tersisip oleh pesan pada " <i>Picture Box Citra Tersisip Pesan</i> "

6. Use Case: Menyimpan Nilai Dekripsi

Aktor: Pengirim Pesan

Tujuan: Menyimpan nilai dekripsi yang didapatkan dari proses penyisipan pesan

Tabel 4.6 Scenario Use Case Menyimpan Nilai Dekripsi

Aktor	Sistem
1. Pengirim pesan menekan tombol "Simpan Nilai Dekripsi"	
	2. Sistem menyimpan nilai dekripsi berupa nilai n, nilai d dan panjang biner

	3. Sistem menyimpan nilai biner kedalam format .txt dan disimpan dalam komputer
	4.Sistem menampilkan kotak dialog “Nilai Dekripsi Tersimpan!”
5. Pengirim pesan mengirim Nilai Dekripsi ke Penerima pesan	

7. Use Case: Simpan Gambar

Aktor: Pengirim Pesan

Tujuan: Menyimpan citra yang telah tersisip oleh pesan

Tabel 4.7 Scenario Use Case Simpan Gambar

Aktor	Sistem
1. Pengirim pesan menekan tombol "Simpan Image"	
	2. Sistem menampilkan halaman pilih media penyimpanan
3. Pengirim pesan memberikan nama pada citra yang akan disimpan	
	4.Sistem menyimpan citra yang tersisip oleh pesan pada tempat yang dipilih pengirim pesan

8. Use Case: Decoding

Aktor: Penerima Pesan

Tujuan: Mengawali proses dekripsi dan ekstr.aksi data

Tabel 4.8 Scenario Use Case Decoding

Aktor	Sistem
1. Pengirim pesan membuka aplikasi dan memilih halaman <i>Encoding</i>	

	2. Sistem Menampilkan Halaman <i>Encoding</i>
3. Pengirim dapat melakukan proses enkripsi dan penyisipan pesan	

9. Use Case: Input Citra RGB

Aktor: Penerima Pesan

Tujuan: Memasukkan citra RGB dengan format .JPG yang tersisip oleh pesan

Tabel 4.9 Scenario Use Case Input Citra RGB

Aktor	Sistem
1. Penerima pesan memilih tombol "Open Image"	
	2. Sistem Menampilkan Halaman pilih citra
3. Pengirim pesan memilih citra RGB dengan format .JPG	
	4. Sistem menampilkan citra RGB pada " <i>Picture Box Citra</i> "

10. Use Case: Ekstraksi Pesan

Aktor: Penerima Pesan

Tujuan: Mengekstraksi nilai biner dari citra yang tersisip pesan

Tabel 4.10 Scenario Use Case Ekstraksi Pesan

Aktor	Sistem
1. Penerima pesan memasukkan panjang nilai biner pada " <i>TextBox</i> "	

	2. Sistem melakukan proses ekstraksi sesuai dengan panjang nilai biner
	3. Sistem menampilkan kotak dialog "Pesan Terekstraksi"
	4.Sistem menampilkan nilai biner yang telah di ekstraksi dari citra

11. Use Case: Dekripsi Pesan

Aktor: Penerima Pesan

Tujuan: Mendekripsi *Ciphertext* yang telah didapatkan

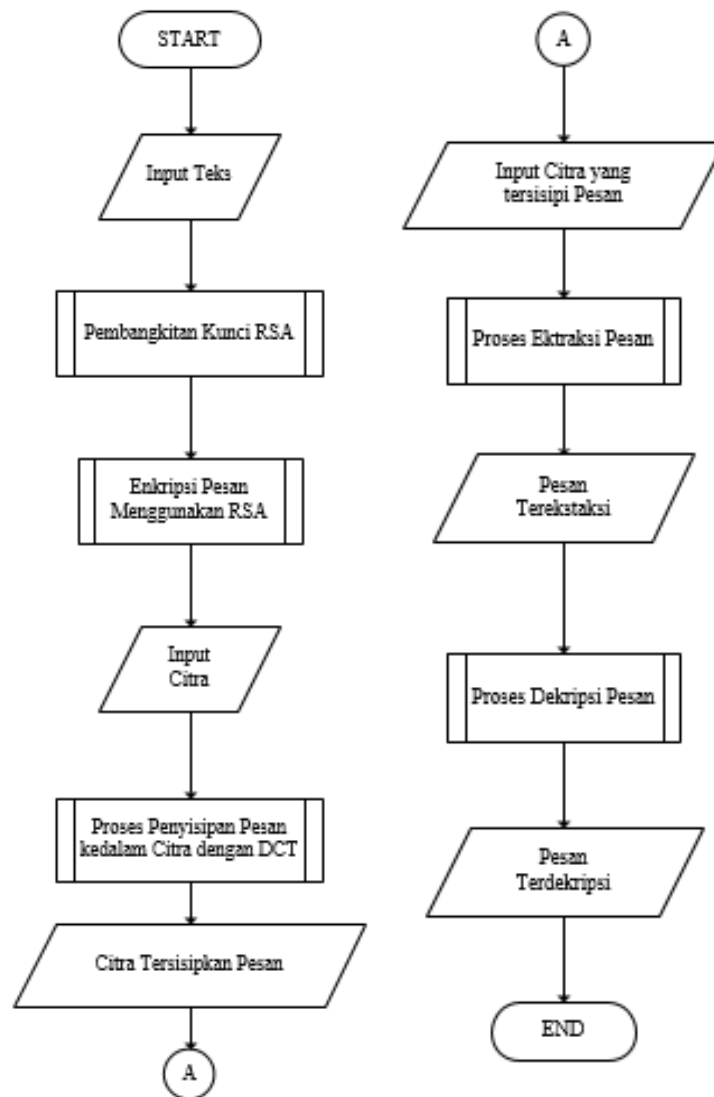
Tabel 4.11 Scenario Use Case Dekripsi Pesan

Aktor	Sistem
1. Penerima pesan memasukkan nilai d dan nilai n	
	2. Sistem mengecek nilai d dan nilai n
	3. Sistem menampilkan kotak dialog "Pesan Terdekripsi"
	4.Sistem menampilkan pesan yang telah di dekripsi

4.1.7. Diagram Alir Sistem

Diagram Alir merupakan serangkaian bagian-bagian yang berfungsi untuk menerangkan alur dari jalannya program. Berikut Diagram Alir yang digunakan untuk pengerjaan aplikasi skripsi ini secara keseluruhan:

1. Alur Keseluruhan Sistem

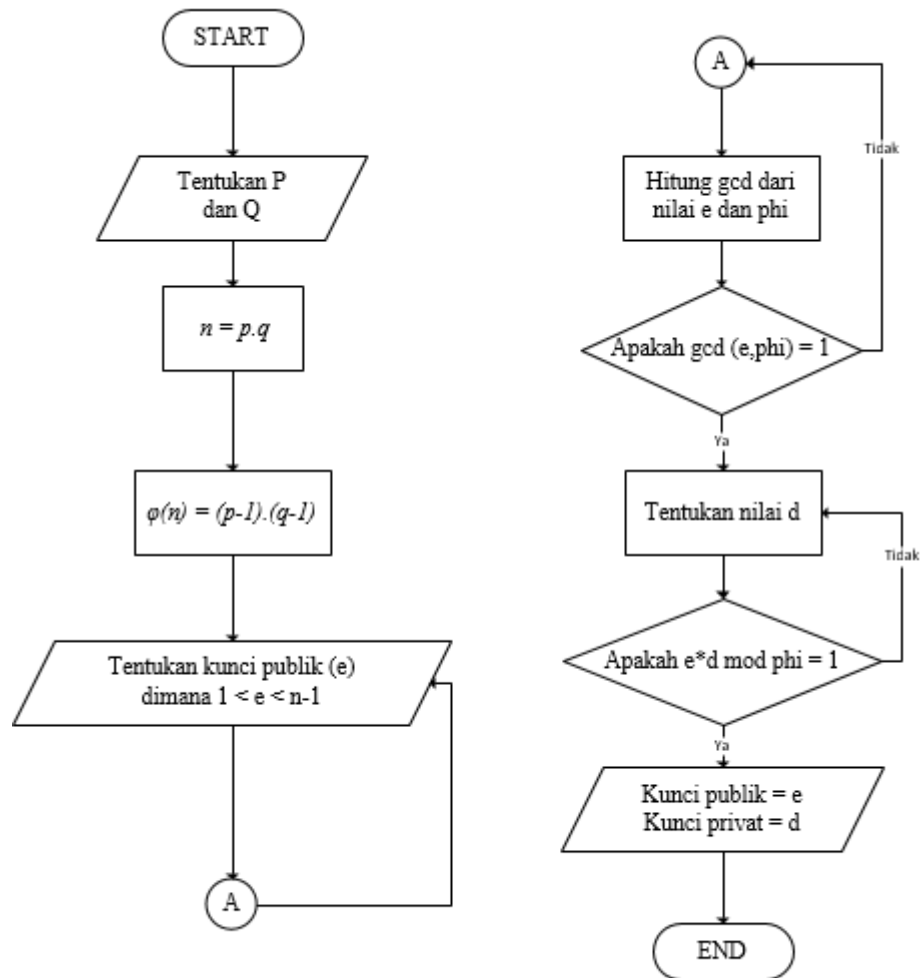


Gambar 4.2 Diagram Alir Keseluruhan

Diagram Alir diatas menggambarkan alur sistem yang akan dibuat mulai dari awal sampai akhir. Terdapat beberapa proses dalam program yang akan dibuat yaitu pembangkitan kunci RSA, Enkripsi Pesan menggunakan RSA,

Penyisipan Pesan menggunakan DCT, Ekstraksi Pesan menggunakan DCT, dan Dekripsi Pesan menggunakan RSA.

2. Pembangkitan Kunci Algoritma RSA



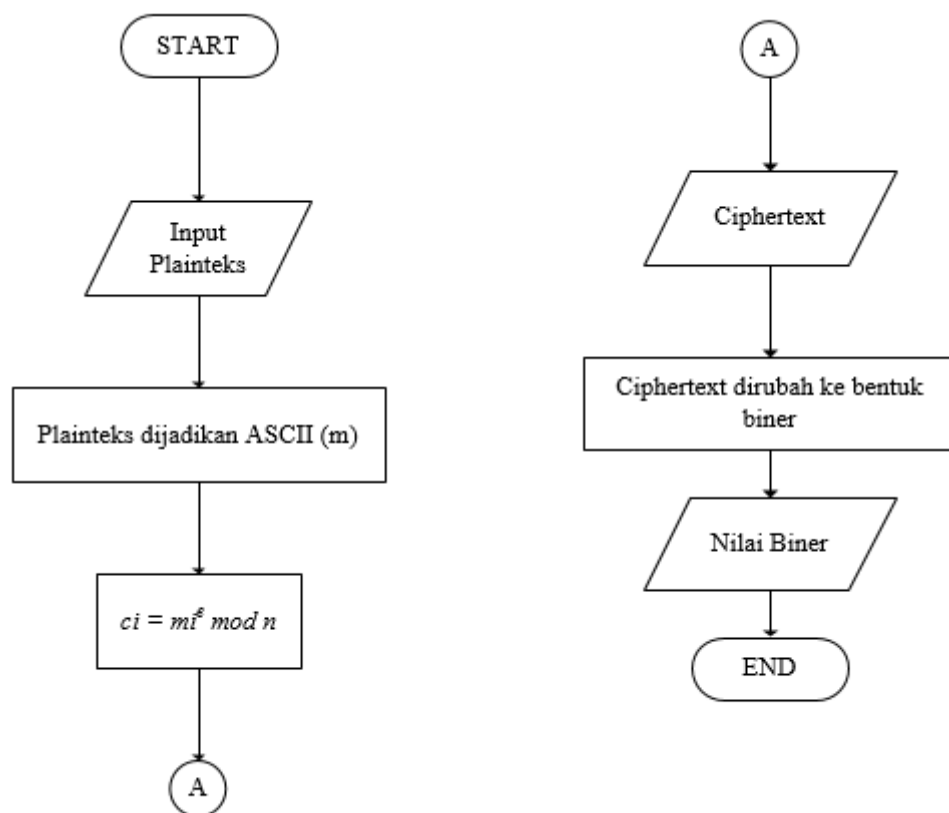
Gambar 4.3 Diagram Alir Pembangkitan Kunci RSA

Pembangkitan Kunci RSA digunakan untuk mendapatkan kunci publik dan kunci pribadi yang masing masing digunakan untuk enkripsi dan dekripsi dari pesan. Langkah yang dilakukan yaitu

- Tentukan nilai p dan q yang merupakan bilangan prima.
- Hitung nilai n dengan mengkalikan p dan q.
- Hitung nilai phi dengan menggunakan rumus $\phi = (p-1).(q-1)$

- Pilih nilai e dimana $1 < e < \phi$ dan jika dilakukan perhitungan GCD hasil dari $GCD(e, \phi)$ harus 1 jika tidak maka akan dilakukan perulangan hingga hasil GCD sama dengan 1.
- Tentukan nilai d dengan syarat jika hasil dari perkalian e dengan d mod ϕ sama dengan 1 maka nilai d akan didapatkan. Untuk mencari nilai d bisa juga dengan menggunakan *Extended Euclidean*.
- Didapatkan kunci publik dan kunci pribadi yang akan digunakan untuk proses enkripsi dan dekripsi.

3. Diagram Alir Enkripsi Algoritma RSA

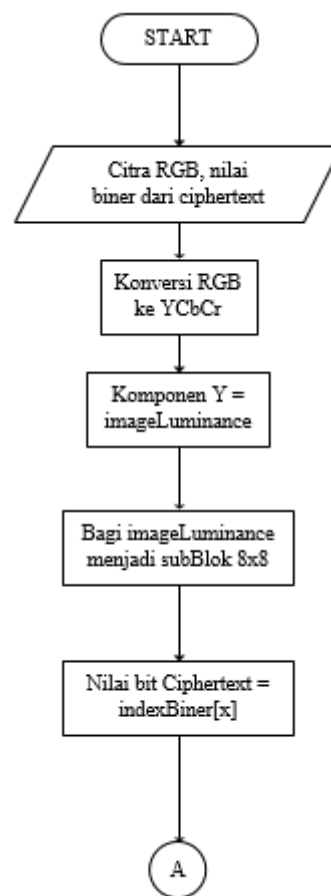


Gambar 4.4 Diagram Alir Enkripsi Pesan

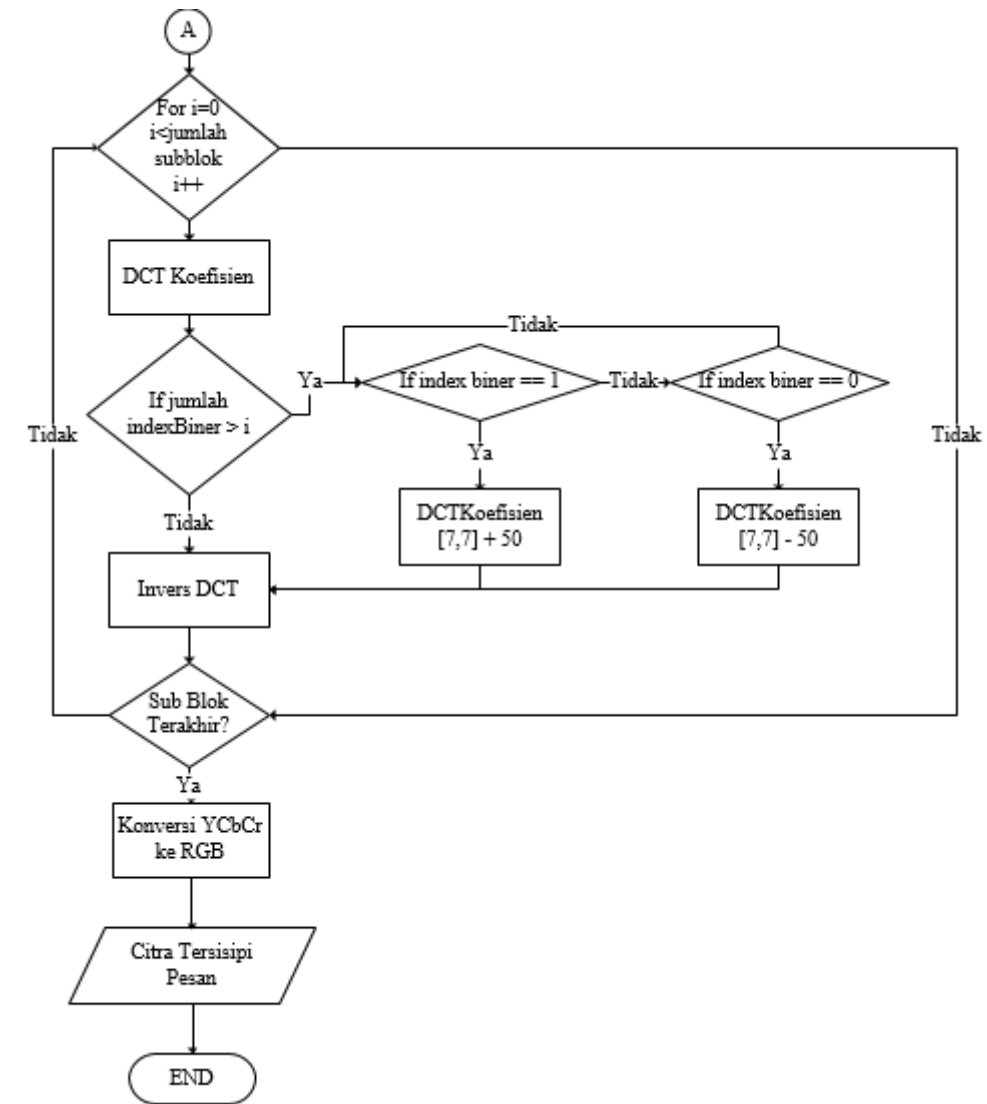
Setelah melakukan pembangkitan kunci RSA, langkah selanjutnya adalah enkripsi pesan. Tahapan dari enkripsi pesan adalah sebagai berikut:

- Masukkan pesan atau *Plaintext* yang akan dienkripsi kedalam sistem.
- Pesan akan dirubah kedalam bentuk ASCII (m).

- Nilai ASCII akan dipangkatkan dengan kunci publik (e) dan dimodkan dengan nilai n untuk mengenkripsi nilai ASCII kedalam bentuk pesan terenkripsi atau *Ciphertext*.
 - Nilai *ciphertext* akan dirubah kedalam bentuk bilangan biner untuk dapat disisipkan kedalam citra digital.
4. Diagram Alir Penyisipan Pesan menggunakan DCT
- Untuk Diagram Alir penyisipan pesan akan dijelaskan pada gambar 4.5 dan 4.6



Gambar 4.5 Diagram Alir Penyisipan (a)



Gambar 4.6 Diagram Alir Penyisipan (b)

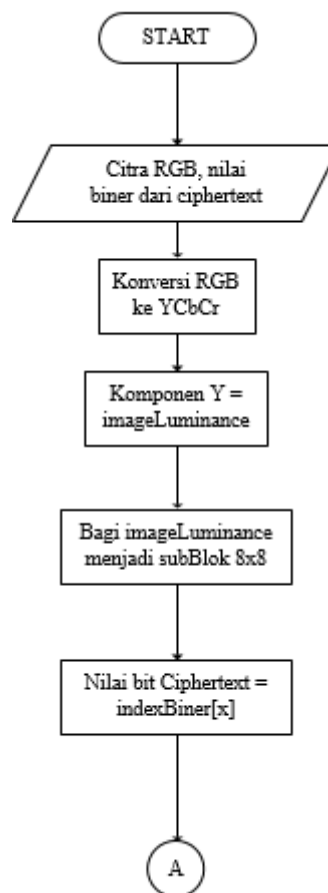
Penjelasan dari Diagram Alir penyisipan pesan adalah sebagai berikut:

- Untuk melakukan penyisipan pesan pengirim memasukkan citra RGB dengan format .JPEG kedalam sistem.
- Selanjutnya citra RGB akan dikonversikan menjadi citra YCbCr, kemudian diambil komponen *luminance* (Y) untuk proses DCT.
- Bagi komponen Y menjadi sub blok berukuran 8 x 8 piksel. Selanjutnya dilakukan transformasi DCT pada setiap sub blok.
- Selanjutnya untuk proses penyisipan, digunakan *ciphertext* yang telah diubah menjadi biner. Jika nilai biner *ciphertext* bernilai 1 maka koefisien dari DCT akan ditambahkan 50, dan jika nilai biner *ciphertext*

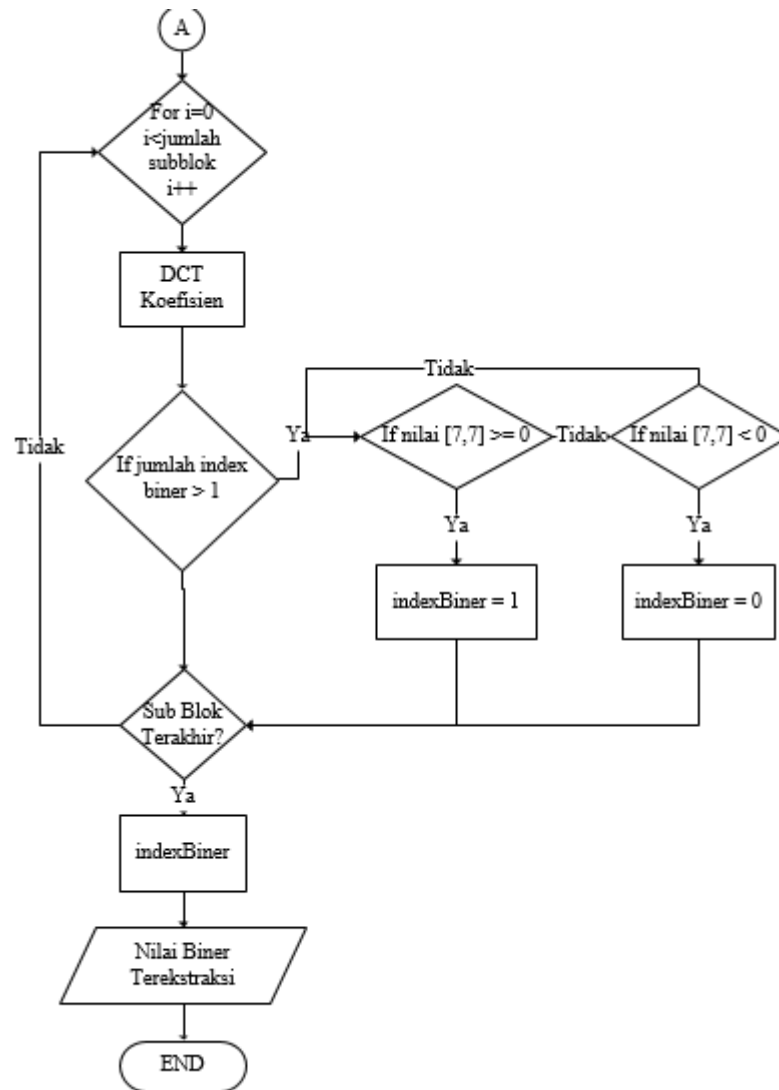
bernilai 0 maka koefisien dari DCT akan dikurangi dengan 50. Proses ini diulang sampai nilai biner *ciphertext* yang terakhir.

- Setelah mencapai nilai biner yang terakhir kemudian dilakukan proses invers DCT untuk mendapatkan komponen Y yang tersisip oleh pesan.
- Kemudian komponen Y digabungkan kembali dengan komponen Cb dan Cr sehingga membentuk citra YCbCr. Proses terakhir konversikan kembali citra YCbCr menjadi citra RGB sehingga menjadi citra RGB yang tersisip oleh pesan.

5. Diagram Alir Ekstraksi Pesan Menggunakan DCT



Gambar 4.7 Diagram Alir Esktraksi (a)

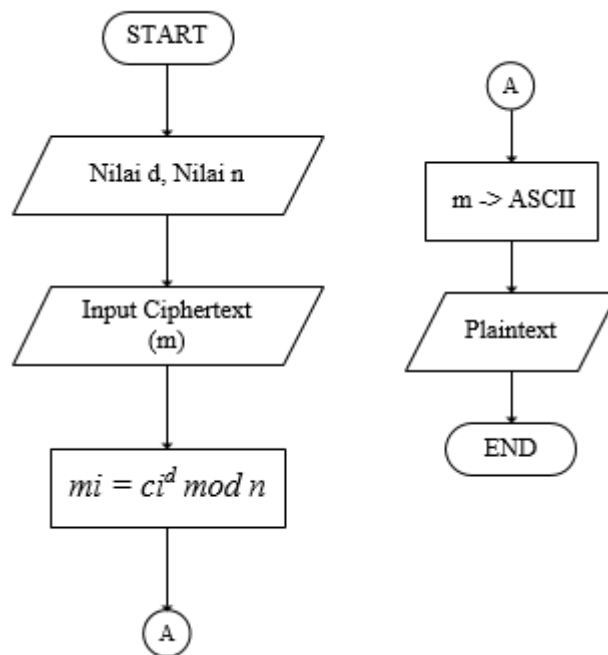


Gambar 4.8 Diagram Alir Ekstraksi (b)

Penjelasan dari Diagram Alir ekstraksi watermark adalah sebagai berikut:

- Citra tersisip oleh pesan merupakan citra RGB. Selanjutnya citra RGB akan dikonversikan menjadi citra YCbCr
- kemudian diambil komponen *luminance* (Y) untuk proses transformasi DCT. Kemudian bagi komponen Y menjadi sub blok berukuran 8 x 8 piksel.
- Dilakukan transformasi DCT pada setiap sub blok. Untuk proses ekstraksi, jika indeks [7, 7] dari koefisien DCT lebih dari 0, maka nilai indeks biner sama dengan 1. Sedangkan jika indeks [7, 7] dari koefisien DCT bernilai lebih kecil dari 0 maka indeks biner sama dengan 0.

- Proses ini diulang sampai jumlah indeks biner yang terakhir. Setelah mencapai indeks biner yang terakhir maka dilakukan proses pengembalian *ciphertext* untuk mengembalikan *ciphertext* yang tersisipkan kedalam citra.
6. Diagram Alir Dekripsi Algoritma RSA



Gambar 4.9 Diagram Alir Dekripsi

- Proses yang terakhir yaitu dekripsi pesan *ciphertext* setelah ekstraksi nilai biner dari citra, nilai biner akan dikembalikan kedalam bentuk *ciphertext*.
- Setelah itu dilakukan dekripsi pada *ciphertext* dengan memasukkan nilai *d* dan nilai *n*.
- Untuk mendekripsi, *ciphertext* akan dipangkatkan dengan nilai *d* dan dimod kan dengan nilai *n*.
- Maka akan kembali kedalam bentuk ASCII dan dikonversi menjadi sebuah pesan atau *Plaintext*.

4.2. Perhitungan Manual Algoritma RSA

4.2.1. Pembangkitan Kunci RSA

- Tentukan p dan q, untuk p dan q harus bilangan prima misalkan p = 61 dan q = 59.
- Hitung nilai n dengan mengkalikan p dan q, $61 * 59 = 3599$.
- Setelah menghitung nilai n maka cari phi dengan rumus $\phi = (p-1).(q-1)$, $\phi = (61-1).(59-1) = 3480$.
- Setelah itu tentukan nilai e dengan memilih bilangan prima antara 1 sampai bilangan phi 3840 yang jika dilakukan algoritma *Euclidean gcd*(3480,e) harus bernilai 1, misalkan kita pilih e dengan nilai 89 karena hasil $\gcd(3480,89) = 1$.
- Cari nilai d dengan persamaan $e*d \bmod \phi = 1$, atau digunakan algoritma *Extended Euclidean gcd*(3480,89) dan ditemukan hasilnya 3089.
- Didapatkan nilai e = 89, d = 3089 dan n = 3599 nilai dari e akan digunakan untuk enkripsi, sedangkan nilai d digunakan untuk dekripsi.

4.2.2. Enkripsi dengan RSA

- Pilih huruf untuk dienkripsi misalkan ABC, huruf ABC akan diubah kedalam bentuk ASCII menjadi 656667 kemudian dibagi perbagian ASCII untuk dilakukan enkripsi dengan kunci e

$$C1 = 65^{89} \bmod 3599 = 1144$$

$$C2 = 66^{89} \bmod 3599 = 49$$

$$C3 = 67^{89} \bmod 3599 = 1352$$

- Maka didapati nilai *ciphertext* 1144, 49, 1352 akan ditambahkan sebuah pemisah dari *ciphertext* dengan menambahkan “,”. Nilai *ciphertext* akan dirubah menjadi bentuk biner dan akan disisipkan ke dalam citra

4.2.3. Dekripsi dengan RSA

- Untuk proses dekripsi kita bagi blok dari *ciphertext* terlebih dahulu kemudian kita gunakan kunci d untuk mendekripsi

$$C1 = 1144^{3089} \bmod 3599 = 65$$

$$C2 = 49^{3089} \bmod 3599 = 66$$

$$C3 = 1352^{3089} \bmod 3599 = 67$$

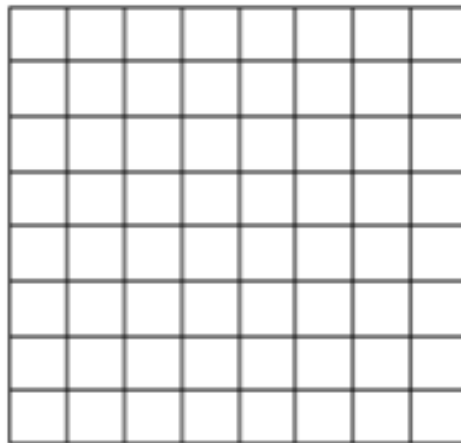
- Didapati nilai dekripsi nya 65, 66, 67 nilai ASCII dari A, B, C

4.3. Perhitungan Manual Metode Discrete Cosine Transform

Proses perhitungan manual akan dilakukan dengan menguji sebuah citra berekstensi jpg dengan ukuran 512 x 512 dengan menggunakan 1 sub blok berukuran 8x8 piksel. Sebelum melakukan proses DCT citra akan dibagi menjadi sub blok yang setiap sub bloknya berukuran 8x8 piksel.



Gambar 4.10 Direktur.jpeg



Gambar 4.11 Sub Blok pertama

Pada gambar direktur.jpeg merupakan gambar berukuran 512 x 512 yang merupakan citra RGB, dan gambar grayscale merupakan sub blok pertama dari gambar direktur.

Tabel 4.12 Tabel nilai matriks sub blok pertama

196	196	196	195	196	196	196	195
197	196	196	197	197	196	196	196
197	197	196	196	197	196	196	198
196	196	196	196	196	196	196	197
196	196	196	195	196	196	196	197
196	197	197	196	196	197	197	197
196	196	197	197	196	197	197	196
196	195	196	197	196	197	197	196

Kemudian dilakukan proses DCT pada subblok pertama dari gambar direktur dan dihasilkan koefisien DCT pada tabel dibawah

Tabel 4.13 Tabel nilai koefisien DCT sub blok pertama

546,35	-0,78	0,419	0,04	-0,0099	-0,0399	0,38	0,1499
-0,81	1,11	0,5299	0,4	10,09	-1,18	-0,42	0,59
-0,3699	0,37	-1,45	0,93	-0,41	1,49	-0,13	-0,049
-0,65	0,32	0,06	0,06	-1,61	-0,2	0,12	0,39
-2,1	-0,43	0,109	0,309	-0,229	0,16	0,229	0,22
-0,08	0,79	-0,05	-0,031	-0,72	-0,02	0,04	-0,08
0,43	-0,3	0,85	-0,1	-0,37	-0,82	-0,05	0,049
-0,2	-0,019	0,51	0,07	0,15	-0,53	0,08	0,34

Proses selanjutnya adalah penyisipan dari nilai *ciphertext* yang sudah dirubah menjadi angka biner 0 dan 1. Penyisipan dilakukan pada indeks [7, 6] dan indeks [7, 7] dengan ketentuan jika nilai biner ciphertext bernilai 1 maka koefisien DCT ditambah 50 dan jika 0 koefisien DCT dikurangi 50. Berikut merupakan hasil dari penyisipan angka biner bernilai 1.

Tabel 4.14 Koefisien Sub blok ditambahkan nilai biner

546,35	-0,78	0,419	0,04	-0,0099	-0,0399	0,38	0,1499
-0,81	1,11	0,5299	0,4	10,09	-1,18	-0,42	0,59
-0,3699	0,37	-1,45	0,93	-0,41	1,49	-0,13	-0,049
-0,65	0,32	0,06	0,06	-1,61	-0,2	0,12	0,39
-2,1	-0,43	0,109	0,309	-0,229	0,16	0,229	0,22
-0,08	0,79	-0,05	-0,031	-0,72	-0,02	0,04	-0,08
0,43	-0,3	0,85	-0,1	-0,37	-0,82	-0,05	0,049
-0,2	-0,019	0,51	0,07	0,15	-0,53	50,08	50,34

Setelah menyisipkan teks kedalam citra, langkah selanjutnya yaitu invers DCT dimana pada langkah ini akan dilakukan proses pengembalian dari citra DCT menjadi citra Grayscale.

Tabel 4.15 Invers DCT sub blok pertama

197	195	198	194	197	196	196	195
195	200	191	201	195	196	197	195
199	191	203	190	200	196	195	199
193	203	187	203	193	196	198	196
199	189	205	188	199	196	194	198
194	203	190	202	194	197	199	196
198	192	202	193	198	197	196	197
195	196	194	198	195	197	197	196

4.4. Tampilan Antarmuka

Pada Aplikasi Kriptografi dan Steganografi Pada Citra Digital Menggunakan Algoritma *Rivest Shamir Adleman* (RSA) dan Metode *Discrete Cosine Transform* (DCT) ini terdapat dua menu utama yaitu Menu *Encoding* yang digunakan untuk proses penyisipan dan menu *Decoding* yang digunakan untuk melakukan proses ekstraksi. Dua menu tersebut yaitu:

4.4.1. Menu Encoding

Pada menu *Encoding* merupakan tampilan utama pada aplikasi ini seperti ditunjukkan pada gambar. Terdapat lima *Button*, dua *Picture Box*, dan sepuluh *TextBox* sebagai berikut:

- Button Generate* Nilai RSA: Berfungsi untuk mendapatkan nilai dari Algoritma RSA ketika diklik maka akan didapatkan kunci untuk mengenkripsi dan dekripsi pesan.
- Button Enkripsi*: Berfungsi untuk mengenkripsi pesan yang sudah dimasukan oleh user.
- Button Proses*: Berfungsi untuk memproses biner kedalam citra.
- Button Open Image*: Berfungsi untuk mengambil citra yang akan disisipkan pesan.
- Button Simpan*: Berfungsi untuk menyimpan citra yang sudah tersisip oleh pesan.

- f. *Picture Box* Citra Asli: Berfungsi untuk menampilkan citra yang akan disisipi pesan.
- g. *Picture Box* Citra Tersisip: Berfungsi untuk menampilkan citra yang sudah tersisip oleh pesan.
- h. *TextBox* nilai p, q, phi, N, d, e: berfungsi untuk menampilkan nilai RSA yang sudah diGenerate.
- i. *TextBox* Masukan Teks: Berfungsi untuk memasukan teks yang akan dienkripsi.
- j. *TextBox Ciphertext*: Berfungsi untuk melihat nilai *ciphertext* yang dihasilkan dari enkripsi pesan.
- k. *TextBox* Nilai biner: Berfungsi untuk melihat nilai biner yang dihasilkan dari konversi *ciphertext* ke biner.
- l. *TextBox* Panjang biner: Berfungsi untuk melihat jumlah dari nilai biner yang akan disisipkan kedalam citra.

The screenshot shows a software interface for encoding and decoding. At the top, there are tabs for 'Encoding' and 'Decoding'. Below the tabs are two buttons: 'Open Image' and 'Save Image'. The main area is divided into several sections:

- Left Section:** Contains input fields for 'Nilai p', 'Nilai q', 'Nilai N', 'Nilai Phi', 'Nilai e', and 'Nilai d'. Below these is a 'Generate Nilai RSA' button.
- Middle Section:** Contains a 'Masukan Teks' (Text Input) field, an 'Enkripsi' (Encrypt) button, and a 'Ciphertexts' output field.
- Right Section:** Contains a 'Panjang Binery' (Binary Length) input field, a 'Proses' (Process) button, and two image display areas labeled 'Gambar Awal' (Initial Image) and 'Gambar Tersisipi' (Image with Message). The 'Gambar Tersisipi' area contains the text 'Picture Box Citra Tersisipi Pesan'.

Gambar 4.12 Desain Halaman Encoding

4.4.2. Menu *Decoding*

Pada menu *Decoding* merupakan tampilan utama pada aplikasi ini. Terdapat empat *Button*, satu *Picture Box*, dan enam *TextBox* sebagai berikut:

- Button* Proses: Berfungsi untuk ekstraksi nilai biner yang sudah tersisipkan dalam citra.
- Button* Dekripsi: Berfungsi untuk mengembalikan *ciphertext* kedalam bentuk informasi asli.
- Button* Open Image: Berfungsi untuk mengambil citra yang akan diekstaksi.
- Picture Box* Citra: Berfungsi untuk menampilkan citra yang sudah tersisip oleh pesan.
- TextBox* Input Panjang biner: Berfungsi untuk menginputkan panjang biner pada citra yang sudah tersisip oleh pesan.
- TextBox* Nilai biner: Berfungsi untuk menampilkan nilai biner yang sudah terekstraksi dari citra.
- TextBox* Nilai *Ciphertext*: Berfungsi untuk menampilkan nilai *ciphertext* yang dihasilkan dari pengembalian nilai biner kedalam bentuk *ciphertext*.
- TextBox* Input Nilai *d, n* : Berfungsi untuk menginputkan nilai *d* dan *N* yang digunakan untuk mendekripsi *ciphertext*.
- TextBox* Plainteks: Berfungsi untuk menampilkan informasi yang sudah terdekripsi.

Gambar 4.13 Desain Halaman Decoding

BAB V IMPLEMENTASI

Pada bab ini akan membahas implementasi aplikasi yang telah dibuat dengan menggunakan bahasa pemrograman Visual Vb.Net. Di bawah ini merupakan langkah-langkah penggunaan aplikasi kriptografi dan steganografi yang telah dibuat.

5.1. Implementasi Sistem

Bab implementasi adalah melakukan penulisan kode sesuai dengan apa yang direncanakan. Aplikasi memiliki 5 tahapan yaitu pembangkitan kunci RSA, enkripsi pesan, penyisipan pesan, ekstraksi pesan, dan dekripsi pesan. Untuk tahapan dari implementasi penulisan kode adalah sebagai berikut:

5.1.1. Pembangkitan Kunci RSA

Untuk melakukan enkripsi dan dekripsi pesan, maka dilakukan proses pembangkitan kunci dimana proses ini digunakan menemukan kunci publik dan kunci pribadi. Berikut merupakan potongan kode dari proses pembangkitan kunci:

```
Dim D, E1, N As Double
Const pqAtas As Short = 90
Const pqBawah As Short = 10
Const limitPq As Integer = 10
p = 0 : q = 0
Randomize()

Do Until D > limitPq
    Do Until cekPrima(p) And cekPrima(q)
        p = Int((pqAtas - pqBawah) * Rnd()
+ pqBawah)
        q = Int((pqAtas - pqBawah) * Rnd()
+ pqBawah)
    Loop

    N = p * q
    PHI = (p - 1) * (q - 1)
    E1 = GCD(PHI)
    D = ExtendedEuclid(E1, PHI)
Loop

Key(1) = E1
Key(2) = D
Key(3) = N
txtP.Text = CStr(p) 'P
txtQ.Text = CStr(q) 'Q
```

```

txtPhi.Text = CStr(PHI) 'PHI
txtE.Text = CStr(Key(1)) 'E
txtD.Text = CStr(Key(2)) 'D
txtN.Text = CStr(Key(3)) 'N

```

Kode Sumber 5.1 Pembangkitan Kunci RSA

5.1.2. Enkripsi RSA

Setelah didapatkan nilai RSA dari proses pembangkitan kunci, selanjutnya dilakukan proses enkripsi. Sebelum melakukan enkripsi, pengguna sistem harus memasukkan pesan pada *TextBox*. Setelah dilakukan proses enkripsi maka bentuk nilai *ciphertext* akan diubah menjadi bentuk biner. Berikut merupakan potongan kode dari proses enkripsi:

```

Dim i As Double
    Dim hasilEnkripsi As String

    For i = 1 To
Len(txtPlaintext.Text)
        hasilEnkripsi = hasilEnkripsi
& Mult(CInt(Asc(Mid(txtPlaintext.Text, i, 1))) _
        , Key(1), Key(3)) &
        ", "

    Next i
    txtEncrypt.Text = hasilEnkripsi
    txtCipher.Text = hasilEnkripsi

    Dim Temp As String
    Dim Builder As New
System.Text.StringBuilder
    For Each Character As Byte In
System.Text.ASCIIEncoding.ASCII.GetBytes(txtEncrypt
t.Text)

Builder.Append(Convert.ToString(Character,
2).PadLeft(8, "0"))
        Builder.Append("")
    Next
    Temp = Builder.ToString
    txtEncrypt.Text = Temp

    Dim text As String =
txtEncrypt.Text

```

```

Dim arrayBinari() As Char
arrayBinari = text.ToCharArray
Dim jmlArrayBinary As Integer =
arrayBinari.Length() - 1

```

Kode Sumber 5.2 Enkripsi RSA

5.1.3. Penyisipan Pesan

Setelah mendapatkan nilai biner dari proses enkripsi pesan, selanjutnya dilakukan proses penyisipan pesan kedalam citra. Proses penyisipan memiliki ketentuan jika nilai biner 1 maka koefisien dari matriks nilai DCT ditambah 50 dan jika nilai biner 0 maka koefisien dari matriks nilai DCT dikurangi 50. Berikut merupakan potongan kode dari proses penyisipan:

```

Dim x As Integer = 0
For i As Integer = 0 To
subBlokCitra.Count - 1
    blockImage = subBlokCitra(i)
    Dim citraDCT As Double(,)
    Dim citraIDCT As Double(,)

    DCT = New RumusDCT(blockImage,
block)
    citraDCT = DCT.DCT(blockImage)

    Dim arrayBinari() As Char
    Dim text As String =
txtEncrypt.Text
    arrayBinari = text.ToCharArray

    Dim jumlahArrayBinary As
Integer = arrayBinari.Count()

    Dim v As Integer = 6
    Dim h As Integer = 7
    If (jumlahArrayBinary > i)
Also (x < jumlahArrayBinary) Then

        For z As Integer = 0 To 1
            If arrayBinari(x) =
"1" Then
                citraDCT(h, v) =
citraDCT(h, v) + 50
            ElseIf arrayBinari(x) = "0" Then

```

```

citraDCT(h, v) - 50
citraDCT(h, v) =
End If
x += 1
v += 1
Next
End If

```

Kode Sumber 5.3 Penyisipan Pesan

5.1.4. Ekstraksi Pesan

Setelah dilakukan proses penyisipan selanjutnya dilakukan proses ekstraksi pesan. Untuk melakukan proses ekstraksi nilai biner, pengguna harus memasukan panjang nilai biner yang terdapat pada citra. Panjang nilai biner dikirimkan oleh pengirim kepada penerima sehingga apabila panjang biner yang dimasukkan salah maka hasilnya akan error dan tidak bisa dilakukan proses pengembalian ke bentuk *ciphertext* nya. Proses ekstraksi memiliki ketentuan jika nilai koefisien dari matriks $DCT < 0$ maka nilai biner sama dengan 0 dan jika nilai koefisien dari matriks $DCT \geq 0$ maka nilai biner sama dengan 1. Setelah dilakukan proses ekstraksi angka biner yang sudah terekstraksi akan diubah menjadi *Ciphertext* untuk kemudian dilakukan proses dekripsi. Berikut merupakan potongan kode dari proses ekstraksi:

```

Dim x As Integer = 0
    Dim arrayBinari() As Char
    Dim text As String = txtRahasia.Text
'siasati panjang binary
    arrayBinari = text.ToCharArray
    Dim jmlArrayBinary As Integer =
Integer.Parse(txtRahasia.Text)
    Dim listIndexBinary As New List(Of
String)

    For i As Integer = 0 To
subBlokWatermark.Count - 1
        blokWatermark =
subBlokWatermark(i)
        Dim matrixEnkripsi As Double(,)

        DCT = New RumusDCT(blokWatermark,
block)

```



```

matrixEnkripsi =
DCT.DCT(blokWatermark)

Dim v As Integer = 6
Dim h As Integer = 7
If (jmlArrayBinary > i) AndAlso (x
< jmlArrayBinary) Then

    For z As Integer = 0 To 1
        If matrixEnkripsi(h, v) >=
0 Then

            'indexBinary(x) = "1"

listIndexBinary.Add("1")
        ElseIf matrixEnkripsi(h,
v) < 0 Then

            'indexBinary(x) = "0"

listIndexBinary.Add("0")
        End If
        x += 1
        v += 1

    Next
End If

```

```

Next
x = 0
Dim indexBinary() As String =
listIndexBinary.ToArray()
For i = 0 To indexBinary.Length - 1
    txtBinary.Text += indexBinary(i)

Next

Dim Val As String = Nothing
Dim Characters As String =
System.Text.RegularExpressions.Regex.Replace(txtBi
nary.Text, "[^01]", "")
Dim ByteArray((Characters.Length / 8)
- 1) As Byte

For Index As Integer = 0 To
ByteArray.Length - 1

    ByteArray(Index) =
Convert.ToByte(Characters.Substring(Index * 8, 8),
2)

```

```

Next

Val =
System.Text.ASCIIEncoding.ASCII.GetString(ByteArra
y)

encrypt.Text = Val

```

Kode Sumber 5.4 Ekstraksi Pesan

5.1.5. Dekripsi Pesan

Selanjutnya dilakukan proses dekripsi pesan pada ciphertext yang telah didapatkan dari proses ekstraksi. Untuk melakukan proses dekripsi, pengguna harus memasukkan nilai d dan nilai n. Berikut merupakan potongan kode dari proses dekripsi pesan:

```

Dim nilai_ciphertext As Double
    Dim pisah_ciphertext As Double
    Dim z As Int16
    Dim hasilDekripsi As String
    For z = 1 To Len(encrypt.Text) - 1

        pisah_ciphertext = InStr(z,
encrypt.Text, ",")

        nilai_ciphertext =
Val(Mid(encrypt.Text, z, pisah_ciphertext))
        hasilDekripsi = hasilDekripsi
& Chr(Mult(nilai_ciphertext, nilaiD.Text,
nilaiN.Text))
        z = pisah_ciphertext

    Next z
decrypt.Text = hasilDekripsi

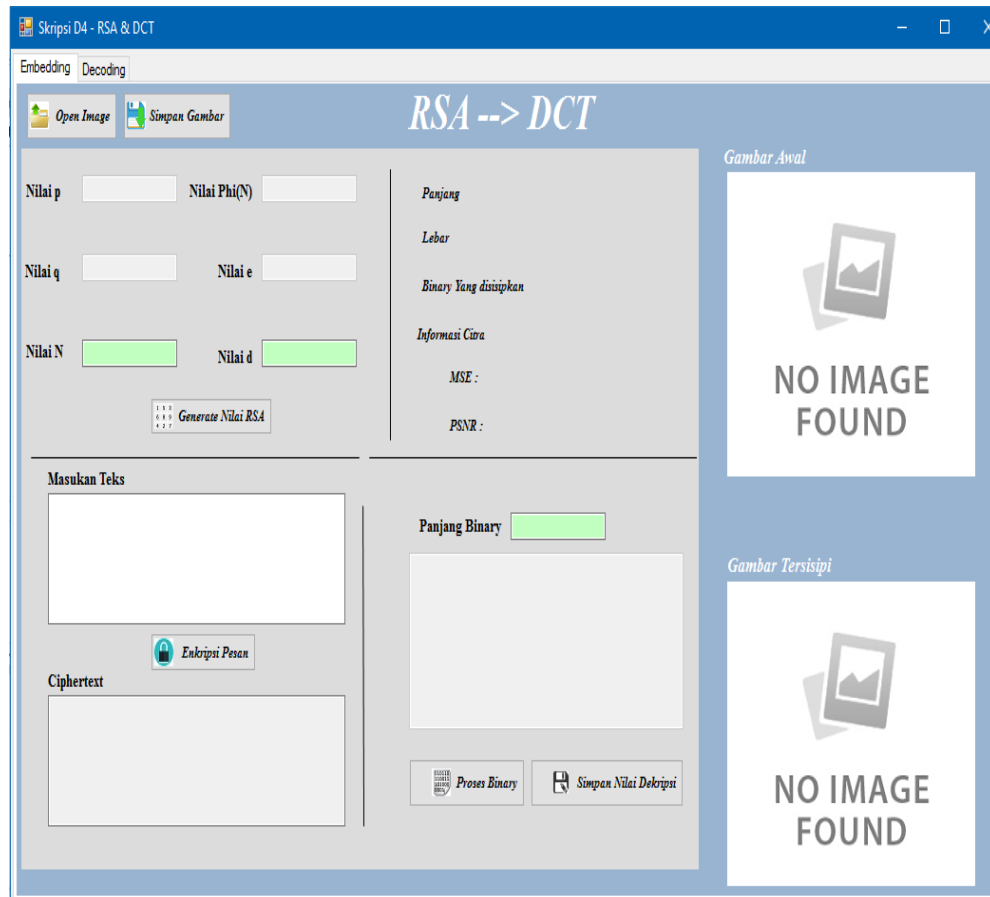
```

Kode Sumber 5.5 Dekripsi Pesan

5.2. Implementasi Desain

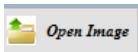
5.2.1. Halaman Encoding

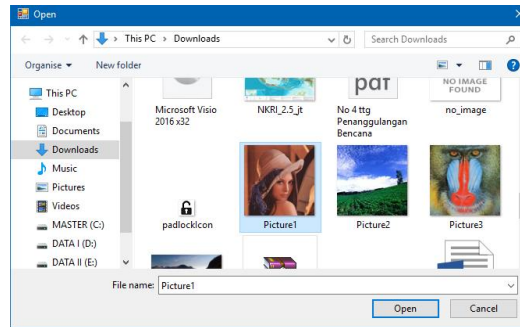
Halaman *Encoding* merupakan halaman yang digunakan sebagai penyisipan pesan ke dalam citra. Pada halaman ini ditampilkan fitur untuk mendapatkan nilai RSA, citra awal, citra tersisip oleh pesan, serta hasil enkripsi dari data teks yang dimasukan oleh pengguna.



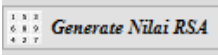
Gambar 5.1 Halaman Encoding

Tahapan untuk melakukan proses Encodding adalah sebagai berikut:

1. Pilih tombol  untuk membuka dan memilih citra yang akan tersisip oleh pesan.



Gambar 5.2 Open Image

2. Pilih tombol  untuk mendapatkan nilai RSA yang digunakan pada saat enkripsi pesan teks yang akan dimasukkan pengguna.

Gambar 5.3 Generate RSA

3. Pilih tombol  untuk mengenkripsi pesan yang telah dimasukkan.

Gambar 5.4 Hasil enkripsi

4. Setelah dilakukan proses enkripsi pada teks yang dimasukan, pilih tombol

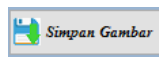


untuk menyisipkan angka biner kedalam citra.

5. Penyisipan pesan kedalam citra berhasil dilakukan, setelah penyisipan selesai akan ditampilkan citra awal dan citra tersisipi pesan. Selain itu juga ditampilkan nilai MSE dan PSNR dari perbandingan dua citra.

Gambar 5.5 Hasil penyisipan biner

6. Untuk menyimpan citra yang sudah tersisipi pesan, pilih tombol



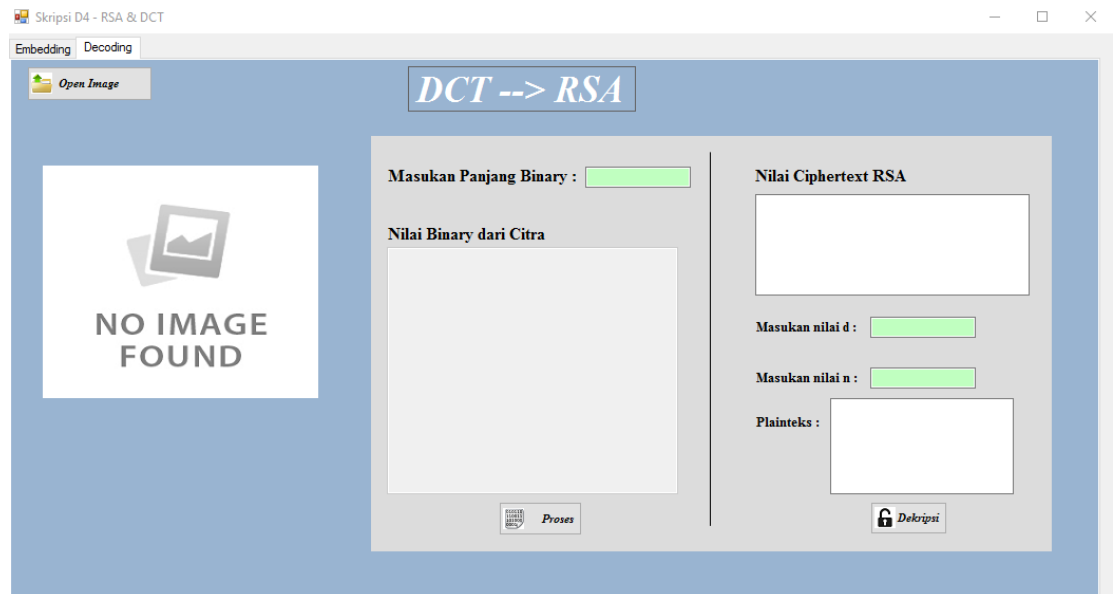
dan untuk menyimpan nilai d, n, dan panjang biner pilih



tombol maka citra dan nilai dekripsi akan tersimpan pada komputer.

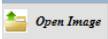
5.2.2. Halaman Decoding

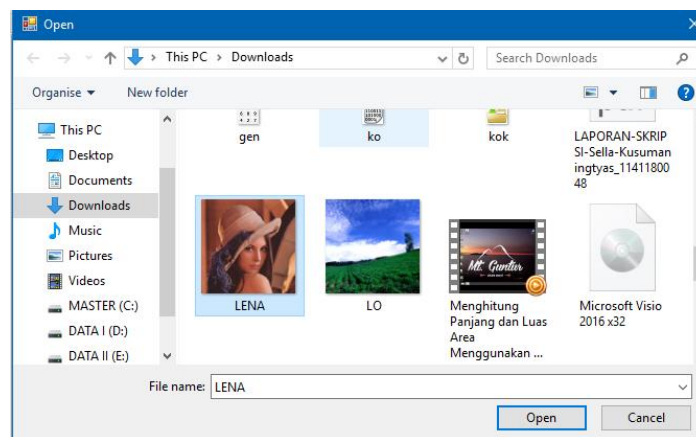
Halaman *Decoding* digunakan sebagai pengambilan pesan yang telah disisipkan pada citra. Pada halaman ini terdapat fitur untuk memasukan dan mendapatkan nilai biner, merubah biner ke *ciphertext*, serta melakukan dekripsi *ciphertext*.




Gambar 5.6 Halaman Decoding

Tahapan untuk melakukan proses Decodding adalah sebagai berikut:

1. Pilih tombol  untuk membuka dan memilih citra yang sudah tersisipi oleh pesan.



Gambar 5.7 Pilih Citra

2. Masukkan panjang biner dan tekan tombol  untuk mendapatkan nilai biner dari citra yang sudah tersisipi pesan.

Masukan Panjang Binary : 287

Nilai Binary dari Citra

```
001101110011000100110110001011000011010000110
100001110010011011100101100001100010011010100
111000001100100010110000110100001100110011100
000110111001011000011001100110100001110000010
110000110011001100000011100000111000001011000
011100100110101001101110010110000110101001100
110011010000101100
```

Gambar 5.8 Hasil Biner

3. Nilai biner terekstraks

Masukan Panjang Binary : 287


Nilai Binary dari Citra

```
001101110011000100110110001011000011010000110
100001110010011011100101100001100010011010100
111000001100100010110000110100001100110011100
000110111001011000011001100110100001110000010
110000110011001100000011100000111000001011000
```

Nilai Ciphertext RSA

716,4497,1582,4387,348,3088,957,534,

Gambar 5.9 Hasil Ekstraksi

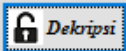
4. Masukkan nilai d dan nilai N , kemudian tekan tombol  untuk melakukan proses dekripsi *ciphertext* ke dalam bentuk pesan.

Masukan nilai d : 509

Masukan nilai n : 5063

Plainteks :

POLINEMA



Gambar 5.10 Hasil Dekripsi

BAB VI Pengujian dan Pembahasan

Setelah melakukan proses penyisipan pesan dan ekstraksi pesan, maka untuk melihat apakah hasil dari penyisipan pesan dan ekstraksi pesan telah berhasil akan dilakukan suatu pengujian.

Pengujian dilakukan berdasarkan spesifikasi sistem dan pengujian ketahanan data. Pengujian spesifikasi sistem yang dilakukan meliputi pengujian kesesuaian proses, pengujian kesesuaian data, dan pengujian kualitas citra. Pengujian berdasarkan spesifikasi sistem dan ketahanan data diuraikan menjadi 2 faktor pengujian sebagai berikut:

- a. Kesesuaian proses, yaitu aplikasi dapat melakukan proses penyisipan pesan dan ekstraksi pesan.
- b. Kesesuaian data, yaitu pengujian kesesuaian antara data yang berhasil diekstraksi dengan data yang disisipkan.

6.1. Pengujian Sistem

Untuk tahap pengujian sistem menggunakan metode blackbox. Metode ini memungkinkan adanya pengembangan untuk melatih seluruh fungsi pada sistem. Metode ini digunakan untuk mendemonstrasikan jalannya aplikasi dan menemukan kesalahan saat aplikasi dijalankan. Dengan menggunakan metode ini dapat dinilai apakah input yang diterima dan output yang dihasilkan sudah tepat atau belum. Berikut blackbox dari pengujian sistem:

Tabel 6.1 Pengujian Blackbox

No	Skenario Pengujian	Hasil Yang diharapkan	Hasil Pengujian	Kesimpulan
1	Pada Menu <i>Encoding</i> Citra Awal kosong, klik "Proses Binary"	Muncul Message Box "error"	Sesuai Harapan	Berhasil
2	Pada menu <i>Encoding</i> klik "Open Image" untuk disisipkan pesan	<i>Picture Box</i> Citra Asli menampilkan citra	Sesuai Harapan	Berhasil

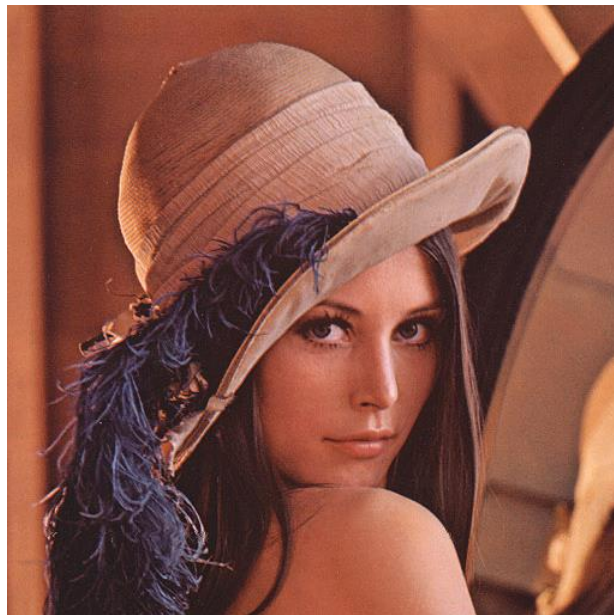
3	Pada menu <i>Encoding</i> klik " <i>Generate Nilai RSA</i> " untuk mendapat nilai RSA	Sistem menampilkan Nilai RSA secara acak	Sesuai Harapan	Berhasil
4	Pada menu <i>Encoding</i> pesan kosong, klik "Enkripsi"	Muncul Message Box "error"	Sesuai Harapan	Berhasil
5	Pada menu <i>Encoding</i> nilai d, n dan panjang biner kosong, klik "Simpan Nilai Dekripsi"	Muncul Message Box "error"	Sesuai Harapan	Berhasil
6	Pada menu <i>Encoding</i> Citra Awal dan pesan sudah di enkripsi, klik "Proses Binary"	Sistem menyisipkan pesan pada Citra Awal, jika berhasil akan muncul Message Box "Success"	Sesuai Harapan	Berhasil
7	Pada menu <i>Encoding</i> nilai n, d, dan panjang biner terisi, klik "Simpan Nilai Dekripsi"	Sistem menyimpan nilai d, n, dan panjang biner dalam bentuk .txt	Sesuai Harapan	Berhasil
8	Pada menu <i>Encoding</i> klik "Simpan Image"	Tampil kotak dialog "Save As" untuk menyimpan Citra yang sudah tersisipi pesan	Sesuai Harapan	Berhasil
9	Pada menu <i>Decoding</i> panjang biner belum terisi, klik "Proses"	Muncul Message Box "error"	Sesuai Harapan	Berhasil
10	Pada menu <i>Decoding</i> Citra Sisip kosong, klik "Proses"	Muncul Message Box "error"	Sesuai Harapan	Berhasil
11	Pada menu <i>Decoding</i> panjang biner dan citra terisi klik "Proses"	Sistem menampilkan hasil ekstraksi pesan	Sesuai Harapan	Berhasil
12	Pada menu <i>Decoding</i> nilai n dan nilai d kosong, klik "Dekripsi"	Muncul Message Box "error"	Sesuai Harapan	Berhasil

13	Pada menu <i>Decoding</i> nilai n dan nilai d terisi, klik "Dekripsi"	Sistem mendekripsi pesan	Sesuai Harapan	Berhasil
----	---	--------------------------	----------------	----------

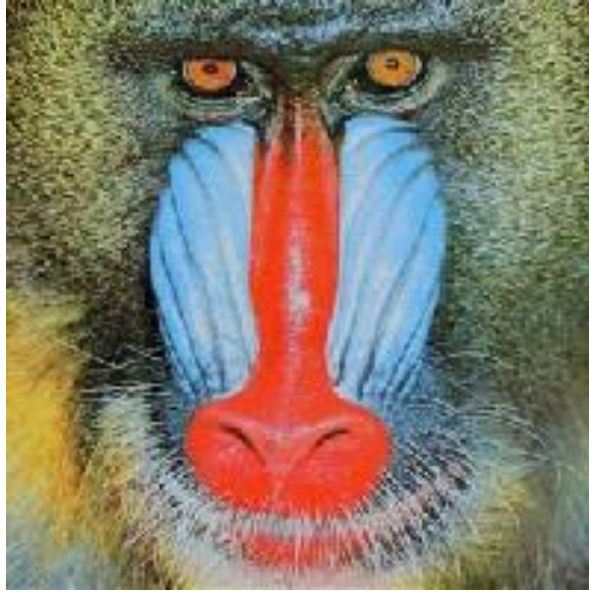
Pada Tabel dapat dilihat hasil pengujian sistem menggunakan blackbox. Berdasarkan dari hasil pengujian sistem menggunakan blackbox, maka dapat ditarik kesimpulan bahwa aplikasi penyisipan pesan menggunakan algoritma RSA dan metode DCT sudah berjalan sesuai dengan harapan.

6.2. Pengujian Kesesuaian Data

Pengujian terhadap kesesuaian data dilakukan untuk mengetahui apakah pesan yang berhasil diekstrak dari citra tersisip pesan sesuai dengan pesan yang disisipkan. Kriteria pengujian adalah pesan yang berhasil diekstrak dari citra tersisip pesan sesuai dengan pesan yang disisipkan. Gambar 6.1, Gambar 6.2, Gambar 6.3 dan Gambar 6.4 merupakan yang digunakan untuk pengujian:



Gambar 6.1 Picture 1 512x512



Gambar 6.2 Picture 2 200x200



Gambar 6.3 Picture 3 400x400



Gambar 6.4 Picture 4 1024x768

Sedangkan pesan yang digunakan dilakukan proses enkripsi dengan menggunakan nilai RSA sebagai berikut:




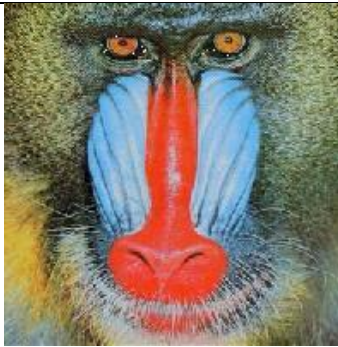




- Nilai $p = 59$
- Nilai $q = 37$
- Nilai $n = 2183$
- Nilai $\phi = 2088$
- Nilai $e = 79$
- Nilai $d = 1903$

Kemudian pesan yang digunakan untuk proses penyisipan adalah sebagai berikut:

- Pesan = "POLINEMA"
- *Ciphertext* = "993,425,1386,2071,1806,1943,633,280,"
- Panjang nilai biner *ciphertext* = "287"

Dari data citra penampung dan citra watermark diatas. Selanjutnya lakukan pengujian kesesuaian data dengan menyisipkan watermark kemudian lakukan ekstraksi. Hasil pengujian kesesuaian data ditunjukkan pada Tabel 6.2:

Tabel 6.2 Kesesuaian Data

	Citra Awal	Citra Tersisip Pesan	<i>Ciphertext</i>
1			993,425,1386,20 71,1806,1943,63 3,280,
2			99s(27,1786,207 1,18<V,q943,63 3,280,
3			993,425,1386,20 71,1806,1943,63 3,280,
4			993,425,1386,20 71,1806,1943,63 3,280,

Pada hasil ekstraksi diatas masing masing citra menghasilkan hasil *ciphertext* yang sesuai, namun untuk “Picture2.jpg” menghasilkan nilai *ciphertext* yang berbeda sehingga tidak dapat dilakukan proses dekripsi pesan.

6.3. Analisis

6.3.1. Analisis Terhadap Pesan

Pada sub bab ini dilakukan analisis pemilihan pesan yang baik untuk proses penyisipan pada citra awal dengan cara membandingkan panjang ukuran biner dari pesan yang telah dienkripsi menggunakan nilai RSA.

- Pesan 1 = “POLINEMA”, biner *ciphertext* 287
- Pesan 2 = “POLINEMA JURUSAN TEKNOLOGI INFORMASI”, biner *ciphertext* 1255
- Pesan 3 = “POLINEMA JURUSAN TEKNOLOGI INFORMASI PROGRAM STUDI TEKNIK INFORMATIKA”, biner *ciphertext* 2303

Masing masing pesan akan disisipkan kedalam citra pada frekuensi tinggi dan diambil 2 bilangan biner untuk disisipkan dalam 1 subblok, pada Tabel ditunjukan hasil pengujian yang diukur dengan PSNR dari Citra Awal dan Citra Tersisip Pesan.

Tabel 6.3 Perbandingan penyisipan pesan

No	Pesan	Citra	MSE	PSNR
1	Pesan 1	Picture1.jpg	2,70681	43,8062
		Picture2.jpg	30,5758	33,2770
		Picture3.jpg	4,4353	44,6615
		Picture4.jpg	0,9019	48,5788
2	Pesan 2	Picture1.jpg	11,8048	37,4102
		Picture2.jpg	98,3487	28,2031
		Picture3.jpg	19,3397	35,2662
		Picture4.jpg	5,5479	40,6894
3	Pesan 3	Picture1.jpg	21,6471	34,7767
		Picture2.jpg	105,7660	27,8874
		Picture3.jpg	77,5739	29,2336
		Picture4.jpg	26,5283	33,8937

Dari hasil pengujian pemyisipan pesan, dapat disimpulkan bahwa penyisipan pesan dengan panjang biner berbeda memiliki kualitas Citra Tersisip Pesan yang

bagus. Citra Tersisip Pesan dengan panjang biner 287 memberikan hasil ekstraksi nilai biner yang baik pada masing masing citra, namun pada citra “Picture2.jpg” berukuran 200 x 200 piksel menghasilkan proses esktraksi yang berbeda dari citra lainnya dimana pada saat dikembalikan ke *ciphertext* ada bagian yang berubah. Pada Citra Tersisip Pesan dengan panjang biner 1255 menghasilkan proses ekstraksi yang baik pada citra “Picture1.jpg” berukuran 512 x 512 dan “Picture4.jpg” berukuran 1024 x 768 piksel, namun pada citra berukuran 400 x 400 piksel dan 200 x 200 piksel nilai biner pada saat dikembalikan ke *ciphertext* terdapat bagian yang berubah. Pada Citra Tersisip Pesan dengan panjang nilai biner 2303 menghasilkan proses ekstraksi yang baik pada citra “Picture1.jpg” berukuran 512 x 512 piksel , namun pada citra “Picture2.jpg”, “Picture3.jpg”, dan “Picture4.jpg” terdapat perubahan nilai biner dan jika dikembalikan ke *ciphertext* terdapat bagian yang berubah.

Dari Tabel diatas dapat disimpulkan bahwa semakin panjang nilai biner yang disisipkan kedalam citra semakin rendah nilai PSNR yang didapat. Nilai PSNR terendah ditunjukkan pada citra “Picture2.jpg” dengan nilai PSNR 27,8874. Semakin besar dimensi citra yang digunakan maka semakin lama proses penyisipan dan proses ekstraksi nilai biner, maka dari itu dipilih Pesan 1 dengan panjang nilai biner 287 untuk percobaan selanjutnya. Sehingga untuk mendapatkan hasil ekstraksi yang baik, pada saat melakukan enkripsi sebaiknya nilai biner tidak terlalu panjang agar dapat dilakukan proses dekripsi pesan.




6.3.2. Analisis Perbandingan Waktu Penyisipan dan Ekstraksi

Pada sub bab ini dilakukan analisis waktu penyisipan dan ekstraksi pada pesan yang akan disisipkan pada citra. Terdapat tiga pesan masukan dengan panjang biner yang berbeda yaitu:

- Pesan 1 = “POLINEMA”, biner *ciphertext* 287
- Pesan 2 = “POLINEMA JURUSAN TEKNOLOGI INFORMASI”, biner *ciphertext* 1255
- Pesan 3 = “POLINEMA JURUSAN TEKNOLOGI INFORMASI PROGRAM STUDI TEKNIK INFORMATIKA”, biner *ciphertext* 2303




Untuk melakukan pengujian, digunakan citra “Picture 1.jpg” dengan ukuran 512x512 piksel. Berikut hasil perbandingan pada tabel 6.4:

Tabel 6.4 Perbandingan Penyisipan Biner

Pesan	Citra Sisip	Waktu Penyisipan
Pesan 1 = 287 biner		8,88 detik
Pesan 2 = 1255 biner		9,03 detik
Pesan 3 = 2303 biner		10,29 detik

Sedangkan untuk hasil perbandingan waktu ekstraksi dijelaskan pada tabel berikut:

Tabel 6.5 Perbandingan Ekstraksi Biner

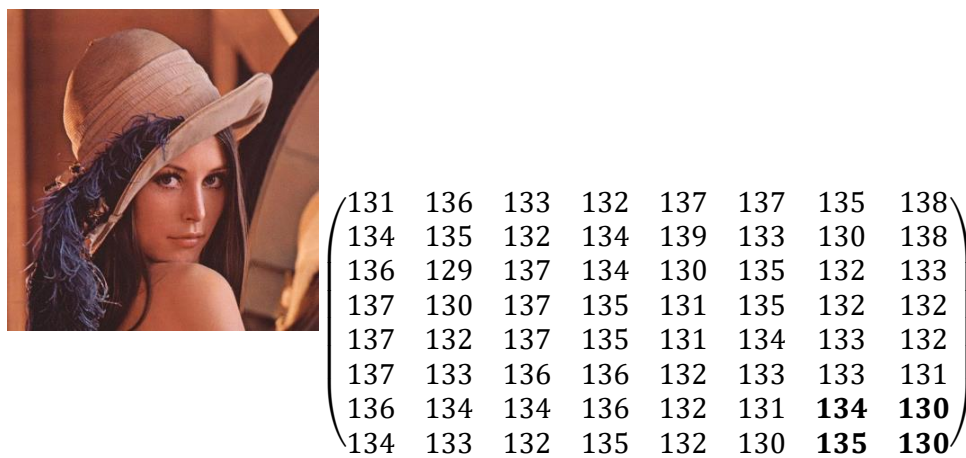
Pesan	Citra Sisip	Waktu Ekstraksi
Pesan 1 = 287 biner		5,54 detik
Pesan 2 = 1255 biner		12,03 detik
Pesan 3 = 2303 biner		30,65 Detik

Dari tabel penyisipan dan ekstraksi diatas dapat disimpulkan, pada saat melakukan penyisipan pesan pada citra waktu yang dibutuhkan bergantung pada nilai biner yang disisipkan, semakin besar nilai biner, semakin lama proses penyisipan. Ketika dilakukan proses ekstraksi pada pesan dengan nilai biner,

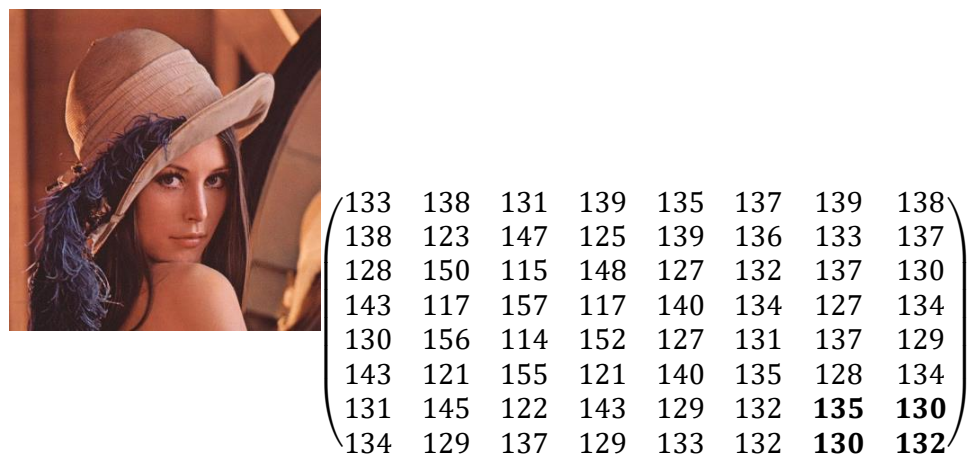
semakin besar nilai biner yang dimasukkan akan semakin lama proses ekstraksi pesan.

6.3.3. Analisis Perbandingan Nilai Piksel

Pada sub bab ini akan dijelaskan contoh perhitungan nilai MSE dan PSNR dari dua citra yaitu citra awal dan citra tersisip pesan. Perhitungan MSE dan PSNR dilakukan pada sub blok pertama citra, untuk data citra yang digunakan yaitu citra “Picture 1.jpg” yang mempunyai ukuran 512x512 piksel. Hasil dari perhitungan MSE dan PSNR sebagai berikut:



Gambar 6.5 Koefisien DCT pada citra awal



Gambar 6.6 Koefisien DCT pada citra sisip

Setelah didapati nilai sub blok pertama akan dilakukan perhitungan MSE dan PSNR dari indeks yang telah disisipkan pesan yaitu indeks [7,6] dan [7,7]. Untuk

perhitungannya akan digunakan indeks matriks yaitu [6,6], [6,7], [7,6], dan [7,7] berikut contoh perhitungan MSE dan PSNR dari sub blok pertama:

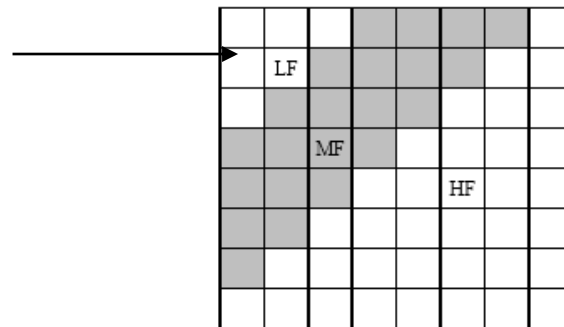
$$MSE = \frac{(134 - 135)^2 + (130 - 130)^2 + (135 - 130)^2 + (130 - 132)^2}{4} = 16,5$$

$$PSNR = 10 \log_{10} \frac{255}{16,5} = 154.54$$

6.3.4. Analisis Perbandingan Penyisipan Pesan Antar Frekuensi

Pada metode DCT, host image yang ditransformasikan ke dalam koefisien DCT berukuran 8 x 8 piksel terbagi menjadi tiga, yaitu: frekuensi rendah, frekuensi menengah, dan frekuensi tinggi. Pada percobaan ini akan dilakukan analisis perbandingan penyisipan citra watermark pada masing-masing frekuensi. Kemudian kualitas citra tersisip pesan setelah disisipkan akan diukur menggunakan PSNR. Pesan yang akan disisipkan yaitu “POLINEMA” dengan panjang nilai biner *ciphertext* 287.









6.3.4.1. Penyisipan Pesan pada Frekuensi Rendah



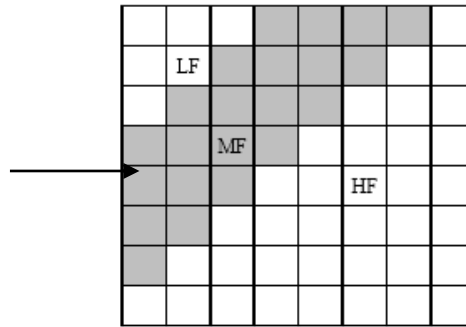
Gambar 6.7 Frekuensi Rendah

Penyisipan pesan pada frekuensi rendah dilakukan pada indeks [0, 1] dan [0, 2] dari koefisien DCT. Untuk mengetahui kualitas citra tersisip pesan yang telah disisipi pesan pada frekuensi rendah, dilakukan perhitungan MSE dan PSNR yang ditunjukkan pada Tabel 6.6:

Tabel 6.6 Frekuensi Rendah

Citra Awal	Citra Tersisip Pesan	MSE	PSNR
		2,8522	43,5788
		25,3345	34,0936
		4,5384	41,5616
		27,2866	33,7712


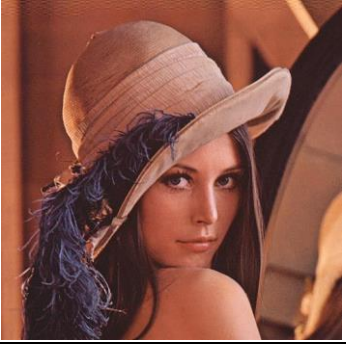


6.3.4.2. Penyisipan Pesan pada Frekuensi Menengah







Gambar 6.8 Frekuensi Menengah

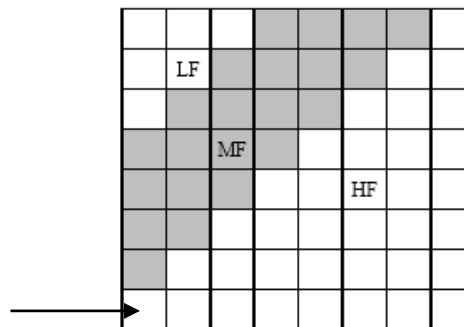
Percobaan kedua pesan disisipkan pada frekuensi menengah pada koefisien DCT pada citra awal dan dilakukan pada indeks $[3, 0]$ dan $[3, 1]$. Untuk mengetahui kualitas citra tersisip pesan yang telah disisipi pesan pada frekuensi menengah, dilakukan perhitungan MSE dan PSNR yang ditunjukkan pada Tabel 6.7:

Tabel 6.7 Frekuensi Menengah

Citra Awal	Citra Tersisip Pesan	MSE	PSNR
		2,7755	43,6973
		26,8193	33,8463

		4,5482	41,5524
		0,9253	48,4676



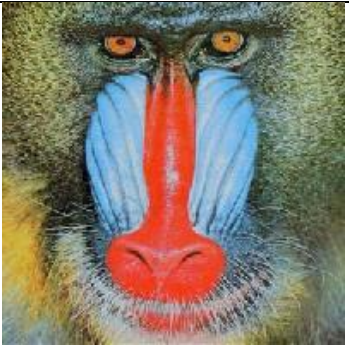





6.3.4.3. Penyisipan Pesan Pada Frekuensi Tinggi



Gambar 6.9 Frekuensi Tinggi

Percobaan ketiga pesan disisipkan pada frekuensi tinggi pada koefisien DCT pada citra awal dan dilakukan pada indeks [7, 6] dan [7, 7]. Untuk mengetahui kualitas citra tersisip pesan yang telah disisipi pesan pada frekuensi tinggi, dilakukan perhitungan MSE dan PSNR yang ditunjukkan pada Tabel 6.8:

Tabel 6.8 Frekuensi Tinggi

Citra Awal	Citra Tersisip Pesan	MSE	PSNR
		2,7068	43,8062
		30,5758	33,2770
		4,4353	41,6615
		0,9019	48,5788

Berdasarkan ketiga percobaan di atas, didapatkan hasil perbandingan yang ditampilkan pada Tabel 6.9:

Tabel 6.9 Perbandingan Frekuensi

No	Frekuensi	Citra	MSE	PSNR
1	Frekuensi Rendah	Picture1.jpg	2,8522	43,5788
		Picture2.jpg	25,3345	34,0936
		Picture3.jpg	4,5384	41,5616
		Picture4.jpg	27,2866	33,7712
2	Frekuensi Menengah	Picture1.jpg	2,7755	43,6973
		Picture2.jpg	26,8193	33,8463
		Picture3.jpg	4,5482	41,5524
		Picture4.jpg	0,9253	48,4676
3	Frekuensi Tinggi	Picture1.jpg	2,7068	43,8062
		Picture2.jpg	30,5758	33,2770
		Picture3.jpg	4,4353	41,6615
		Picture4.jpg	0,9019	48,5788

Dari tabel di atas dapat dilihat bahwa penyisipan pesan yang dienkrpsi dengan panjang nilai biner 287 pada masing masing frekuensi memiliki nilai PSNR yang baik. Namun penyisipan pesan pada frekuensi tinggi memiliki nilai PSNR yang lebih tinggi dibandingkan dengan penyisipan pada frekuensi rendah dan frekuensi menengah.

Maka dari itu, penyisipan pesan dilakukan pada frekuensi tinggi. Dikarenakan penyisipan pada frekuensi tinggi memiliki nilai PSNR yang paling tinggi hingga 48,5788.

6.3.5. Analisis Perbandingan Penyisipan Pesan pada Sub Blok

Selanjutnya dilakukan pengujian pada proses penyisipan pesan. Pada proses ini dilakukan pengecekan nilai biner yang dapat disisipkan kedalam 1 sub blok yang sudah didapatkan koefisien DCT pada citra awal. Hasil pengujian dari penyisipan pesan yang dienkrpsi dengan panjang nilai biner 287 ditampilkan pada Tabel 6.10:

Tabel 6.10 Perbandingan penyisipan biner

No	Banyak Nilai Biner	Citra	MSE	PSNR
1	1	Picture1.jpg	2,6749	43,7139
		Picture2.jpg	21,1409	34,8795
		Picture3.jpg	4,5298	41,5699
		Picture4.jpg	0,9217	48,4848
2	2	Picture1.jpg	2,7068	43,8062
		Picture2.jpg	30,5758	33,2770
		Picture3.jpg	4,4353	41,6615
		Picture4.jpg	0,9019	48,5788
3	4	Picture1.jpg	2,7491	43,7387
		Picture2.jpg	24,3403	34,3675
		Picture3.jpg	4,5024	41,5945
		Picture4.jpg	0,9163	48,5099
4	8	Picture1.jpg	2,7378	43,7567
		Picture2.jpg	21,4583	34,8148
		Picture3.jpg	4,4853	41,6128
		Picture4.jpg	0,9123	48,5291

Pada hasil pengujian penyisipan nilai biner dengan menyisipkan nilai biner 1, 2, 4, dan 8 memiliki PSNR yang baik. Untuk panjang nilai biner 287 pada masing masing pengujian ketika dilakukan ekstraksi hasilnya baik dan nilai biner dapat kembali ke bentuk *ciphertext*. Namun untuk penyisipan lebih dari 2 nilai biner akan merubah kualitas dari citra sehingga jika dimasukkan pesan yang ketika dilakukan enkripsi menghasilkan panjang biner yang lebih dari 287 akan menghasilkan kualitas citra yang kurang baik.

Maka dari itu penyisipan akan menggunakan 2 nilai biner dikarenakan penyisipan dengan 2 nilai biner dapat menghasilkan kualitas citra yang baik dan ketika dilakukan proses ekstraksi nilai biner tidak banyak berubah.



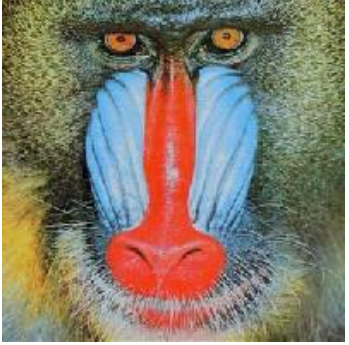
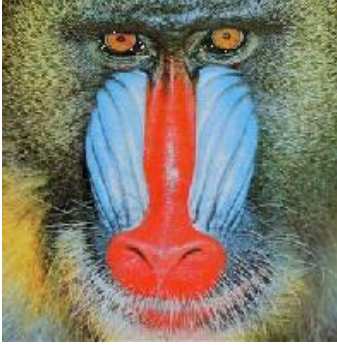




6.3.6. Analisis Perbandingan Penyisipan Nilai pada Koefisien DCT

6.3.6.1. Penyisipan pesan dengan nilai 10

Selanjutnya dilakukan pengujian pada proses penyisipan pesan, pada proses ini penyisipan pesan pada citra dilakukan dengan menambahkan nilai pada koefisien DCT pada indeks [7, 6] dan [7, 7]. Dengan ketentuan jika nilai biner 1

maka koefisien DCT ditambah 10, dan jika nilai biner 0 maka koefisien DCT dikurangi 10.







Tabel 6.11 Koefisien ditambah 10



Citra Awal	Citra Tersisip Pesan	Hasil
		<p><i>Ciphertext</i> = ?0?LeJtk??o~?F qd???>??,??, ?,?????7</p> <p>MSE = 0,1088 PSNR = 57,763</p>
		<p><i>Ciphertext</i> = ??C?S??3 ^?4s? ??- u6?T!???JH??</p> <p>MSE = 0,7133 PSNR = 49,597</p>
		<p><i>Ciphertext</i>= /??Eo????=???? mZ????^ ? • ??rl?)??-??</p> <p>MSE = 0,178 PSNR = 55,618</p>
		<p><i>Ciphertext</i>= ????????????? ?????????????5^ ?????</p> <p>MSE = PSNR =</p>

6.3.6.2. Penyisipan pesan dengan nilai 30

Pada proses ini penyisipan pesan pada citra dilakukan dengan menambahkan nilai pada koefisien DCT pada indeks [7, 6] dan [7, 7]. Dengan ketentuan jika nilai biner 1 maka koefisien DCT ditambah 30, dan jika nilai biner 0 maka koefisien DCT dikurangi 30 seperti ditunjukkan pada tabel 6.10:

Tabel 6.12 Koefisien ditambah 30



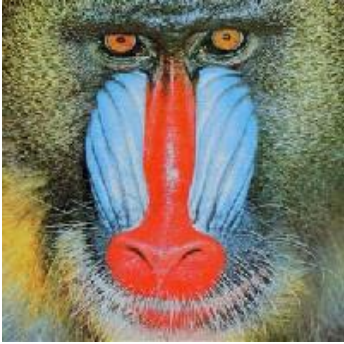
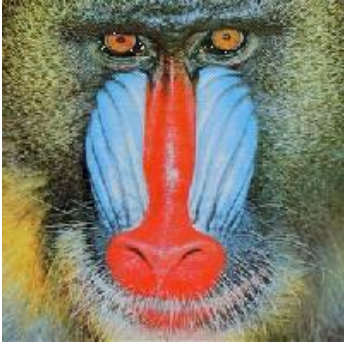
Citra Awal	Citra Tersisip Pesan	Hasil
		<p><i>Ciphertext</i> = JA=q Z*K?u?+ Y-c ?A([/~ ??w}? ?</p> <p>MSE = 0,990 PSNR = 48,171</p>
		<p><i>Ciphertext</i> =)+k]+ND?/s? f;?}xw • \%*?JaY?<</p> <p>MSE = 11,031 PSNR = 37,704</p>
		<p><i>Ciphertext</i>= La8)(:?.?HI^' ?JYsl- \V}?! ?~</p> <p>MSE = 1,622 PSNR = 46,028</p>


		<p><i>Ciphertext</i>= *%*"*+((x?xr* ?(*X"-?("Z'm- u-?j3 ?L</p> <p>MSE = 0,330 PSNR = 52,945</p>
---	--	--

6.3.6.3. Penyisipan pesan dengan nilai 50

Pada proses ini penyisipan pesan pada citra dilakukan dengan menambahkan nilai pada koefisien DCT pada indeks [7, 6] dan [7, 7]. Dengan ketentuan jika nilai biner 1 maka koefisien DCT ditambah 50, dan jika nilai biner 0 maka koefisien DCT dikurangi 50 seperti ditunjukkan pada tabel 6.11:

Tabel 6.13 Koefisien ditambah 50

Citra Awal	Citra Tersisip Pesan	Hasil
		<p><i>Ciphertext</i> = 993,425,1386,2 071,1806,1943, 633,280,</p> <p>MSE = 2,70681 PSNR = 43,806</p>
		<p><i>Ciphertext</i> = 99s(27,1786,20 71,18<V,q943,6 33,280</p> <p>MSE = 30,575 PSNR = 33,277</p>

		<p><i>Ciphertext</i>= 993,425,1386,2 071,1806,1943, 633,280,</p> <p>MSE = 4,435 PSNR = 41,661</p>
		<p><i>Ciphertext</i>= 993,425,1386,2 071,1806,1943, 633,280,</p> <p>MSE = 0,901 PSNR = 48,578</p>

Pada hasil ketiga pengujian diatas dapat disimpulkan bahwa, semakin kecil nilai yang disisipkan pada koefisien DCT maka nilai tersebut akan hilang ketika dilakukan ekstraksi. Oleh Karena itu dipilih nilai 50 karena dapat melakukan ekstraksi dengan hasil yang baik dan dapat didekripsi menjadi sebuah pesan.

6.3.7. Analisis Ketahanan Citra Tersisip Pesan

Setelah melakukan pengujian dengan menghitung PSNR dari citra tersisip oleh pesan, selanjutnya dilakukan analisa citra tersisip pesan terhadap manipulasi citra. Manipulasi citra yang dilakukan adalah *resize* citra yang telah tersisip oleh Pesan 1 dengan panjang biner *ciphertext* 287. Pengujian ini bertujuan untuk menguji seberapa tahan pesan yang disisipkan menggunakan Algoritma RSA dan Metode *Discrete Cosine Transform* terhadap manipulasi citra. Berikut merupakan hasil pengujian *resize* seperti ditunjukkan pada tabel 6.14:

Tabel 6.14 Pengujian Resize

Citra Tersisip Pesan	Citra <i>Resize</i>	<i>Ciphertext</i>
 512 X 512	 400 X 400	???BS?58????_?? co???9\$?07?
 200 X 200	 207 X 207	6?&`d????i???J?I? ??!???&?6v???7?G M?[
 400 X 400	 403 X 403	280,99?????????hl2 7???????\$gg43S?? ????:??
 1024 X 768	 1026 X 769	[993,425,??????? ????????????????3 ?280,

Dari hasil pengujian *resize* citra tersisip pesan dapat disimpulkan bahwa penyisipan pesan setelah dilakukan *resize* menghasilkan kualitas *ciphertext* yang kurang baik. Pada citra tersisip pesan dengan ukuran 1024x768 di *resize* menjadi 1026x769 menghasilkan *ciphertext* yang sama pada nilai awalnya. Sedangkan untuk citra lainnya, nilai *ciphertext* tidak kembali ke bentuk semula. Untuk itu, *resize* citra kurang tepat untuk melakukan ekstraksi *ciphertext*.

BAB VII PENUTUP

7.1. Kesimpulan

Penyisipan watermark menggunakan metode *Discrete Cosine Transform* telah dilakukan dengan menggunakan bahasa pemrograman Visual VB.NET. Berdasarkan hasil pengujian yang telah dilakukan, dapat ditarik kesimpulan:

- a. Citra hasil dari penggunaan Algoritma RSA dan Metode *Discrete Cosine Transform* dapat menyisipkan pesan teks dan tidak dapat dikenali secara kasat mata.
- b. Proses ekstraksi pesan dapat dilakukan dengan memasukkan panjang nilai biner.
- c. Untuk mendapatkan hasil ekstraksi pesan yang baik digunakan pesan dengan hasil enkripsi dengan nilai panjang biner tidak lebih dari 300.
- d. Pesan yang terekstraksi tidak akan bisa kembali kedalam bentuk asli jika nilai dekripsi yang dimasukkan berbeda.
- e. Penyisipan pesan pada frekuensi rendah, menengah, dan tinggi menghasilkan kualitas citra atau nilai PSNR yang baik, namun pada rentang frekuensi tinggi memiliki nilai PSNR yang lebih tinggi dibandingkan dengan frekuensi rendah atau menengah.
- f. Semakin banyak nilai biner yang disisipkan pada citra, maka nilai MSE akan semakin tinggi dan nilai PSNR akan semakin rendah.

7.2. Saran

Berdasarkan penelitian yang diperoleh, ada beberapa saran untuk pengembangan sistem lebih lanjut, sebagai berikut:

- a. Pada penelitian ini menggunakan aplikasi yang berbasis komputer, untuk penelitian selanjutnya diharapkan aplikasi ini dapat dikembangkan dalam bentuk aplikasi mobile (Android).

DAFTAR PUSTAKA

- [1] *Pengertian Kriptografi* [Online]
Tersedia
informatika.stei.itb.ac.id/~rinaldi.../Kriptografi/Bab1_Pengantar%20Kriptografi.pdf [4 Januari 2017]
- [2] J.Elbert Adam, “Understanding and Applying Cryptography and Data Security”, New York, Taylor and Francis Group LLC, 2009. [Online]
Tersedia <https://goo.gl/Kn8yFS> [28 April 2017]
- [3] Agustina Reza dan Andrie Rosa, “*Penyisipan Watermark Dengan Menggunakan Metode Discrete Cosine Transform*”, Malang, Politeknik Negeri Malang, 2015.
- [4] Arifin Rian dan Oktovian Lucki Tri, “*Implementasi Kriptografi Dan Steganografi Menggunakan Algoritma RSA Dan Metode LSB*”, Malang, Universitas Negeri Malang, 2013. [Online] Tersedia
jurnalonline.um.ac.id/data/.../artikel0B100FED42A690D821D5E3F94707B07C.PDF [19 Desember 2016]
- [5] Munir Rinaldi, “*Algoritma RSA dan El Gamal*”, Bandung, Institut Teknologi Bandung, 2004. [Online]
Tersedia
informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/Algoritma%20RSA.pdf [2 Januari 2017]
- [6] Munir Rinaldi. “*Steganografi dan Watermarking*”, Bandung. Institut Teknologi Bandung, 2004. [Online] Tersedia
informatika.stei.itb.ac.id/~rinaldi.munir/.../Steganografi%20dan%20Watermarking.pdf [28 Desember 2016]
- [7] Madenda Sarifudin, “*PENGOLAHAN CITRA DAN VIDEO DIGITAL*”, Jakarta, Erlangga, 2015.

DAFTAR LAMPIRAN

Lampiran 1 Kode Program Penyisipan

```
// ENKRIPSI .....
Dim i As Double
    Dim hasilEnkripsi As String

    For i = 1 To Len(txtPlaintext.Text)
        hasilEnkripsi = hasilEnkripsi &
Mult(CInt(Asc(Mid(txtPlaintext.Text, i, 1))) -
        , Key(1), Key(3)) & ", "

    Next i
    txtEncrypt.Text = hasilEnkripsi
    txtCipher.Text = hasilEnkripsi

    Dim Temp As String
    Dim Builder As New
System.Text.StringBuilder
    For Each Character As Byte In
System.Text.ASCIIEncoding.ASCII.GetBytes(txtEncrypt.Text)
    Builder.Append(Convert.ToString(Character,
2).PadLeft(8, "0"))
        Builder.Append("")
    Next
    Temp = Builder.ToString
    txtEncrypt.Text = Temp

    Dim text As String = txtEncrypt.Text
    Dim arrayBinari() As Char
    arrayBinari = text.ToCharArray
    Dim jmlArrayBinary As Integer =
arrayBinari.Length() - 1

    txtPanjangbin.Text =
jmlArrayBinary.ToString
    MessageBox.Show("Pesan Terenkripsi",
"Success", MessageBoxButtons.OK)

    //Penyisipan.....

    Dim x As Integer = 0
```

```

- 1      For i As Integer = 0 To subBlokCitra.Count
        blockImage = subBlokCitra(i)
        Dim citraDCT As Double(,)
        Dim citraIDCT As Double(,)

        DCT = New RumusDCT(blockImage, block)
        citraDCT = DCT.DCT(blockImage)

        Dim arrayBinari() As Char
        Dim text As String = txtEncrypt.Text
        arrayBinari = text.ToCharArray

        Dim jumlahArrayBinary As Integer =
arrayBinari.Count()

        Dim v As Integer = 6
        Dim h As Integer = 7
        If (jumlahArrayBinary > i) AndAlso (x <
jumlahArrayBinary) Then

            For z As Integer = 0 To 1
                If arrayBinari(x) = "1" Then
                    citraDCT(h, v) =
citraDCT(h, v) + 30
                ElseIf arrayBinari(x) = "0"
Then
                    citraDCT(h, v) =
citraDCT(h, v) - 30
                End If
                x += 1
                v += 1
            Next
        End If

        Dim subDCT As Bitmap
        subDCT = DCT.tampilDCTV2(citraDCT)
        citraIDCT = DCT.IDCT(citraDCT)
        PictureBox2.Image = subDCT

        subImgWatermarked =
DCT.tampilDCTV2(citraIDCT)

        subImgWatermarked.RotateFlip(RotateFlipType.Rotate90Fl
ipX)

        subBlokCitra(i) = subImgWatermarked

```

Lampiran 2 Kode Program Ekstraksi

```

//Ekstraksi.....
For i As Integer = 0 To subBlokWatermark.Count - 1
    blokWatermark = subBlokWatermark(i)
    Dim matrixEkripsi As Double(,)

    DCT = New RumusDCT(blokWatermark,
block)

    matrixEkripsi = DCT.DCT(blokWatermark)

    Dim v As Integer = 6
    Dim h As Integer = 7
    If (jmlArrayBinary > i) AndAlso (x <
jmlArrayBinary) Then

        For z As Integer = 0 To 1
            If matrixEkripsi(h, v) >= 0
Then
                'indexBinary(x) = "1"
                listIndexBinary.Add("1")
            ElseIf matrixEkripsi(h, v) < 0
Then
                'indexBinary(x) = "0"
                listIndexBinary.Add("0")
            End If
            x += 1
            v += 1

        Next
    End If

    Next
    x = 0

    Dim indexBinary() As String =
listIndexBinary.ToArray()
    For i = 0 To indexBinary.Length - 1
        txtBinary.Text += indexBinary(i)

    Next

    Dim Val As String = Nothing
    Dim Characters As String =
System.Text.RegularExpressions.Regex.Replace(txtBinary.
Text, "[^01]", "")
    Dim ByteArray((Characters.Length / 8) - 1)
As Byte

```

```

        For Index As Integer = 0 To
ByteArray.Length - 1

        Next

        Val =
System.Text.ASCIIEncoding.ASCII.GetString(ByteArray)

        encrypt.Text = Val

// Dekripsi.....

    Dim nilai_ciphertext As Double
        Dim pisah_ciphertext As Double
        Dim z As Int16
        Dim hasilDekripsi As String
        For z = 1 To Len(encrypt.Text) - 1

            pisah_ciphertext = InStr(z,
encrypt.Text, ",")

            nilai_ciphertext =
Val(Mid(encrypt.Text, z, pisah_ciphertext))

            hasilDekripsi = hasilDekripsi &
Chr(Mult(nilai_ciphertext, nilaiD.Text, nilaiN.Text))

            z = pisah_ciphertext

        Next z
        decrypt.Text = hasilDekripsi

```

Lampiran 3 Lembar Bimbingan Pembimbing I



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



NO SKRIPSI: 18.

LEMBAR BIMBINGAN SKRIPSI 2016/2017

JUDUL : Implementasi Kriptografi dan Steganografi Pada Citra Digital Menggunakan Algoritma
RSA dan Metode Discrete Cosine Transform

Nama : Dovie Yudhawiratama

NIM : 1341180106

No.	Tanggal	Materi Bimbingan	Tanda Tangan	
			Mahasiswa	Dosen
1.	22/2/17	Pengenalan		
2.	1/3/17	Progress Program		
3.	8/3/17	Pengarah Program		
4.	15/3/17	Presentasi Skripsi		
5.	22/3/17	Pemunculan Proses Enkripsi		
6.	30/3/2017	Presentasi Skripsi		
7.	5/04/2017	Pemunculan Progress		
8.	12/04/17	Pembelajaran Metode Skripsi		
9.	19/04/17	Presentasi Skripsi		
10.	26/04/17	Bab I dan Bab II		
11.	3/05/17	Bab III dan Bab IV		
12.	17/05/17	Revisi Bab III dan Bab IV		
13.	31/05/17	Bab V, Bab VI, dan Bab VII		
14.				
15.				
16.				
17.				
18.				
19.				

Malang, 1 Maret 2017
Dosen Pembimbing Skripsi,

Dr. Eng. Rosa Andrie A., S.T., M.T.
NIP. 19801010 200501 1 001

Lampiran 4 Lembar Bimbingan Pembimbing II



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
Jl. Sockarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



NO SKRIPSI: 18

LEMBAR BIMBINGAN SKRIPSI 2016/2017

JUDUL : Implementasi Kriptografi dan Steganografi Pada Citra Digital Menggunakan Algoritma
RSA dan Metode Discrete Cosine Transform

Nama : Dovie Yudhawiratama

NIM : 1341180106

No.	Tanggal	Materi Bimbingan	Tanda Tangan	
			Mahasiswa	Dosen
1.	8/3/2017	Pengantar		
2.	23/3/2017	Enkripsi		
3.	5/4/2017	Reversan		
4.	12/4/2017	Latihan belahang, Rumus Mersenne		
5.	20/4/2017	Latihan belahang		
6.	20/4/2017	bab 3		
7.	4/5/2017	bab 3 & bab 4		
8.	15/5/2017	bab 5		
9.	29/5/2017	bab 6 - 7		
10.	2/6/2017	Revisi akhir		
11.				
12.				
13.				
14.				
15.				
16.				
17.				
18.				
19.				

Malang, 23 Maret 2017
Dosen Pembimbing II Skripsi,

Ariana Ferti Syafiandini, S.Kom., M.Kom.
NIP.

Lampiran 5 Lembar Revisi Penguji I



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



FORM REVISI SKRIPSI

No. Skripsi: 18

Nama Mahasiswa : Dovie Yudhawiratama NIM: 1341180106
Tanggal Ujian : 8-6-2017
Judul : IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI
PADA CITRA DIGITAL MENGGUNAKAN ALGORITMA
RIVEST SHAMIR ADLEMAN (RSA) DAN METODE
DISCRETE COSINE TRANSFORM (DCT)

NO	SARAN PERBAIKAN	PARAF
	2	

Malang, 8/6-2017
Dosen Penguji,
(.....)

FORM VERIFIKASI:

Laporan Akhir telah diperbaiki sesuai dengan saran perbaikan dari dosen penguji.

PENGUJI/PEMBIMBING	NAMA	TTD	TANGGAL
Penguji	FALSAH R	<i>[Signature]</i>	8/6-2017
Pembimbing 1	Dr Eng Rosa Andria A., ST.MT.	<i>[Signature]</i>	8/6-2017
Pembimbing 2	Arida Ferti Syakrandini, S.Kom	<i>[Signature]</i>	10/6/2017

Lampiran 6 Lembar Revisi Penguji II



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
JL. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



FORM REVISI SKRIPSI

No. Skripsi: 18

Nama Mahasiswa : Doxie Yudhawiratama NIM: 1341180106

Tanggal Ujian : 8-6-2017

Judul : IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI
PADA CITRA DIGITAL MENGGUNAKAN ALGORITMA
RIVEST SHAMIR ADLEMAN (RSA) DAN METODE
DISCRETE COSINE TRANSFORM (DCT)

NO	SARAN PERBAIKAN	PARAF
-	Coba kita kecilkan dan maka	/
-	Tanggal sub pixel warna gambar dan gambar + Ref: Bhs	/

Malang,

Dosen Penguji,

(.....)
FORM VERIFIKASI:

Laporan Akhir telah diperbaiki sesuai dengan saran perbaikan dari dosen penguji.

PENGUJI/PEMBIMBING	NAMA	TTD	TANGGAL
Penguji			12/6 - 2017
Pembimbing 1	Dr. Eng. Rosa Andrie A, ST, MT		17/6 - 2017
Pembimbing 2	Andi Ferti Syah andini, ST, MT		18/6/2017

Lampiran 7 Lembar Verifikasi



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
 Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



No. Skripsi : 18

FORM VERIFIKASI

ABSTRAK BAHASA INGGRIS DAN TATA TULIS BUKU SKRIPSI

Nama Mahasiswa : Dovie Yudhawiratama **NIM** : 13411801060
Tanggal Ujian : 8 Juni 2017
Judul : Implementasi Kriptografi dan Steganografi Pada Citra Digital
 Menggunakan Algoritma Rivest Shamir Adleman (RSA) dan Metode
 Discrete Cosine Transform (DCT)

NO	BAGIAN YANG DIVERIFIKASI	NAMA VERIFIKATOR	TANGGAL VERIFIKASI	TTD
1	Abstrak Berbahasa Inggris	Dr. Eng. Rosa Andrie Asmara, S.T., M.T.	7-8-2017	
2	Tata Tulis Buku Skripsi	Arida Ferti Syafiandini, S.Kom., M.Kom	7/8/2017	

Lampiran 8 Profil Penulis



Nama: Dovie Yudhawiratama

Tempat Tanggal Lahir: Malang, 12 September 1994

Alamat: Jl. Bendungan Sigura Gura Barat no 25, Malang

No HP: 085755602288

Email: doveyudha@gmail.com

RIWAYAT PENDIDIKAN

2001-2007 SDN Percobaan 1, Malang

2007-2010 SMPN 13, Malang

2011-2013 SMAN 8, Malang

2013-2017 Politeknik Negeri Malang