MENGGABUNGKAN TEKNIK STEGANOGRAFI *DISCRETE WAVELET TRANSFORM* DUA DIMENSI (2-D) DAN ALGORITMA KRIPTOGRAFI RSA PADA PERANCANGAN DAN ANALISIS KEAMANAN PESAN

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana Teknik
Program Studi Teknik Elektro Fakultas Sains dan Teknologi

Oleh :

ADE IBRAHIM

11455105224

PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU
PEKANBARU
2021

Lampiran Surat :
Nomor : Nomor 25/2021
Tanggal : 10 September 2021

SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini:

Nama : ADE IBRAHIM
NIM : 11455105224
Tempat/Tgl. Lahir : Pekanbaru, 14 September 1997
Fakultas/Pascasarjana : Sains dan Teknologi
Prodi : Teknik Elektro
Judul Disertasi/Thesis/Skripsi/Karya Ilmiah lainnya*:
Menggabungkan Teknik steganografi DISCRETE WAVELET TRANSFORM DUA DIMENSI (2-D) DAN ALGORITMA KRIPTOGRAFI RSA PADA PERANCANGAN DAN ANALISIS KEAMANAN PESAN

Menyatakan dengan sebenar-benarnya bahwa :

1. Penulisan Disertasi/Thesis/Skripsi/Karya Ilmiah lainnya* dengan judul sebagaimana tersebut di atas adalah hasil pemikiran dan penelitian saya sendiri.
2. Semua kutipan pada karya tulis saya ini sudah disebutkan sumbernya.
3. Oleh karena itu Disertasi/Thesis/Skripsi/Karya Ilmiah lainnya* saya ini, saya nyatakan bebas dari plagiat.
4. Apa bila dikemudian hari terbukti terdapat plagiat dalam penulisan Disertasi/Thesis/Skripsi/(Karya Ilmiah lainnya)* saya tersebut, maka saya besedia menerima sanksi sesua peraturan perundang-undangan.

Demikianlah Surat Pernyataan ini saya buat dengan penuh kesadaran dan tanpa paksaan dari pihak manapun juga.

Pekanbaru, 18 Februari 2022
Yang membuat pernyataan

ADE IBRAHIM
NIM : 11455105224

** pilih salah satu sasuai jenis karya tulis*

LEMBAR PERSETUJUAN

MENGGABUNGKAN TEKNIK STEGANOGRAFI *DISCRETE WAVELET TRANSFORM* DUA DIMENSI (2-D) DAN ALGORITMA KRIPTOGRAFI RSA PADA PERANCANGAN DAN ANALISIS KEAMANAN PESAN

TUGAS AKHIR

Oleh:

ADE IBRAHIM
11455105224

Telah diperiksa dan disetujui sebagai Laporan Tugas Akhir Program Studi Teknik Elektro di Pekanbaru, pada tanggal 21 Januari 2022

Ketua Program Studi	Pembimbing
Digitally signed by Zulfatri Aini Tanggal: 2022.02.11 15:06:04 WIB	Harris Simaremare 2022.02.08 09:50:45 +07'00'
Dr. Zulfatri Aini, ST., MT	**Dr. Harris Simaremare, ST., MT**
NIP. 19721021 200604 2 001	**NIP. 19830625 200801 1 008**

LEMBAR PENGESAHAN

MENGGABUNGKAN TEKNIK STEGANOGRAFI *DISCRETE WAVELET TRANSFORM* DUA DIMENSI (2-D) DAN ALGORITMA KRIPTOGRAFI RSA PADA PERANCANGAN DAN ANALISIS KEAMANAN PESAN

TUGAS AKHIR

Oleh:

ADE IBRAHIM
11455105224

Telah dipertahankan di depan Sidang Dewan Penguji
sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik
Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau
di Pekanbaru, pada tanggal 21 Januari 2022

Pekanbaru, 25 Januari 2022

Mengesahkan,

Ketua Program Studi

Digitally signed by Zulfatri Aini
Tanggal: 2022.02.11 15:06:35 WIB

Dr. Zulfatri Aini, ST., MT
NIP. 19721021 200604 2 001

Dekan

Dr. Hartono, M.Pd
NIP. 19640301 199203 1 003

DEWAN PENGUJI :

Ketua	**: Arif Marsal Lc., M.A**
Sekretaris	**: Dr. Harris Simaremare, ST., MT**
Anggota I	**: Abdillah, S.Si., MIT**
Anggota II	**: Oktaf Brillian Kharisma, ST., MT**

Abdillah Tanggal: 08-02-2022 12:03:10

Harris Simaremare 2022.02.08 09:50:20 +07'00'

LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL

Tugas akhir yang tidak diterbitkan ini terdaftar dan tersedia di Perpustakaan Universitas Islam Negeri Sultan Syarif Kasim Riau adalah terbuka untuk umum dengan ketentuan bahwa hak cipta pada penulis.Referensi kepustakaan di perkenankan dicatat, tetapi pengutipan atau ringkasan hanya dapat dilakukan seizin penulis dan harus disertai dengan kebiasaan ilmiah untuk mencantumkan sebutkan sumbernya.

Penggandaan atau penerbitan sebagian atau seluruh Tugas Akhir ini harus memperoleh izin dari Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Perpustakaan yang meminjamkan Tugas Akhir ini untuk anggotanya diharapkan untuk mengisi nama, tanda peminjaman dan tanggal pinjam.

LEMBAR PERNYATAAN

Dengan ini saya menyatakan bahwa di dalam Tugas Akhir ini tidak terdapat karya yang pernah diajukan oleh saya maupun orang lain untuk keperluan lain, dan sepanjang pengetahuan saya juga tidak memuat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain kecuali disebutkan dalam referensi dan di dalam daftar pustaka.

Saya bersedia menerima sanksi jika pernyataan ini tidak sesuai dengan yang sebenarnya.

Pekanbaru,15 Januari 2022

Yang membuat pernyataan,

ADE IBRAHIM

NIM :11455105224

HALAMAN PERSEMBAHAN

Dengan menyebut nama Allah yang maha pengasih lagi maha penyayang

Barang siapa Yang menghendaki kehidupan dunia, maka wajib baginya berilmu, dan barangsiapa yang menghendaki kehidupan akhirat, maka wajib baginya berilmu, dan barang siapa yang menghendaki keduanya, maka wajib baginya berilmu.

(HR Tirmidzi)

Terima Kasih Ya Allah…

Sembah sujud serta syukurku kepada-Mu ya Allah, zat yang Maha Pengasih namun tak pernah pilih kasih dan Maha Penyayang yang kasih sayang-Nya tak terbilang.Engkau zat yang Maha membolak-balikkan hati, teguhkanlah hati ini di atas agama-Mu ya Allah. Lantunan shalawat beriring salam penggugah hati dan jiwa, menjadi persembahan penuh kerinduan pada sosok panutan umat, pembangun peradaban manusia yang beradab Nabi Besar Muhammad SAW.

Niscaya Allah akan mengangkat (derajat) orang-orang yang beriman diantaramu

dan orang-orang yang diberi ilmu beberapa derajat.

(QS : Al-Mujadilah 11)

Ku persembahkan karya ini untuk Ayahanda tercinta, sosok pejuang dalam hidupku yang tak pernah mengenal kata lelah apalagi mengeluh serta Ibunda tersayang, malaikat tanpa sayap dalam hidupku yang tak kenal waktu siang dan malam selalu menjaga dan melindungi hingga aku bisa sampai seperti sekarang ini, Adik-adik tercinta, seluruh keluarga serta sahabat dan seluruh keluarga besar teknik elektro UIN SUSKA RIAU yang doanya senantiasa mengiringi setiap derap langkahku dalam meniti kesuksesan.

Dan katakanlah:"Ya Tuhan-ku, masukkan aku ketempat masuk yang benar dan keluarkanlah (pula) aku ketempat keluar yang benar dan berilah aku disisi-Mu kekuasaan yang dapat menolongku."

(QS: Al-Isra 80)

MENGGABUNGKAN TEKNIK STEGANOGRAFI DISCRETE WAVELET TRANSFORM DUA DIMENSI (2-D) DAN ALGORITMA KRIPTOGRAFI RSA PADA PERANCANGAN DAN ANALISIS KEAMANAN PESAN

ADE IBRAHIM

NIM :11455105224

Tanggal Sidang :21 Januari 2022

Program Studi Teknik Elektro Teknik Elektro
Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau
Jl. Soebrantas No. 155 Pekanbaru

ABSTRAK

Meningkatnya perkembangan teknologi komunikasi data tentunya berbanding lurus terhadap aspek keamanan dan kerahasiaan data. Berbagai cara dilakukan untuk melindungi informasi dari pihak yang tidak berhak. Beberapa cara tersebut adalah dengan menggunakan teknik steganografi dan teknik kriptografi. Seiring dengan berjalannya waktu, teknik steganografi mulai umum digunakan sehingga memungkinkan informasi rahasia yang disisipkan tersebut masih dapat dipecahkan oleh pihak lain. Hal ini tentu saja mengurangi tingkat keamanan dan kerahasiaan data. Untuk mengatasi masalah tersebut, pada penelitian ini teknik steganografi akan dikombinasikan dengan teknik kriptografi. Perancangan sistem yang dibangun dilakukan dengan menggabungkan penerapan metode algoritma kriptografi Rivest-Shamir-Adleman (RSA) dalam menyandikan pesan, serta penerapan metode steganografi citra dalam menyembunyikan pesan tersandi yang dihasilkan kedalam sebuah citra warna (RGB) dalam domain Discrete Wavelet Transform (DWT). Hasil yang akan diperoleh dari tugas akhir ini adalah sebuah citra yang memiliki pesan terenkripsi pada subband-subband frekuensi citra tersebut. Berdasarkan dari beberapa pengujian yang telah dilakukan pada sistem, telah diperoleh beberapa hasil performansi dengan nilai rata-rata meliputi Avalanche Effect sebesar 7.33 %, Peak Signal to Noise Ratio (PSNR) sebesar 62,3657%, Bit Error Rate (BER) dan Character Error Rate (CER) sebesar 0%.

Kata kunci : Discrete Wavelet Transform, Steganografi Citra, Algoritma Kriptografi RSA

COMBINED TWO DIMENSIONAL (2-D) DISCRETE WAVELET TRANSFORM STEGANOGRAPHY TECHNIQUES AND RSA CRYPTOGRAPHIC ALGORITHM IN MESSAGE SECURITY DESIGN AND ANALYSIS

ADE IBRAHIM

Student Number : 11455105224

Date of Final Exam :21 Januari 2022

Department of Electrical Engineering

Faculty of Science of Technology

State Islamic University of Sultan Syarif Kasim Riau

Soebrantas St. Number. 155 Pekanbaru

ASBTRACK

The increasing development of data communication technology is certainly directly proportional to the security and data confidentiality aspects. There are some ways to protect information from unauthorized parties. Some of these ways are using steganography techniques and cryptographic techniques. Nowadays steganography technique is commonly used, so it still allows the inserted secret information is solved by other parties. This problem may reduce the level of data security and confidentiality. To overcome these problem, in this study steganography techniques will be combined with cryptographic techniques. Design of the built system is done by combining the application of Rivest-Shamir-Adleman (RSA) cryptography algorithm in encrypting the message on application of the image steganography method in concealing the encrypted messages generated into a color image (RGB) on the Discrete Wavelet Transform (DWT). The results to be obtained from this project is an image that has an encrypted message on the subband frequency of the image. Based on some tests that have been done on the system, the result of average performances score is showed by Avalanche Effect of 7,33%, Peak Signal to Noise Ratio (PSNR) of 62,3657%, Bit Error Rate (BER) and Character Error Rate (CER) of 0%. The conclusion is, the implementation of a combination between RSA cryptographic methods and steganography of DWT transformation image is successfully performed.

Key Word : Discrete Wavelet Transform, Image Steganography, RSA Cryptography Algorithm

KATA PENGANTAR

Assalamu'alaikum Wr.Wb

Alhamdulillah, segala puji dan syukur penulis ucapkan kehadirat Allah SWT, yang telah mencurahkan rahmat dan hidayah-Nya kepada penulis sehingga penulis dapat menyelesaikan Tugas Akhir ini. Shalawat dan salam juga penulis haturkan kepada baginda Rasulullah SAW, sebagai seorang sosok pemimpin dan suri tauladan bagi seluruh umat di dunia yang patut di contoh dan di teladani bagi kita semua. Atas ridho Allah SWT penulis telah menyelesaikan Tugas Akhir ini dengan judul "**Menggabungkan Teknik Steganografi Discrete wavelet Transform Dua Dimensi (2-D) Dan Algoritma Kriptografi RSA pada Perancangan dan analisis Keamanan pesan**".

Melalui proses bimbingan dan pengarahan yang disumbangkan oleh orang-orang yang berpengetahuan, dorongan, motivasi, dan juga do'a orang-orang yang ada disekeliling penulis sehingga penulisan Tugas Akhir ini dapat diselesaikan dengan penuh kesederhanaan. Sudah menjadi ketentuan bagi setiap Mahasiswa yang ingin menyelesaikan studinya pada perguruan tinggi UIN SUSKA Riau harus membuat karya ilmiah berupa Tugas Akhir guna mencapai gelar sarjana.

Oleh sebab itu sudah sewajarnya penulis menyampaikan ucapan terima kasih sebesar-besarnya kepada :

1. Ayah yaitu Takim Napitupulu dan Ibu yaitu Asmurniati tercinta yang telah memberikan semangat, dukungan moril maupun materil dan doa kepada penulis serta keluarga besar penulis yang selalu mendoakan penulis.
2. Adik yaitu Abdul Manaf, Khaila Khalizah dan Kakak Sri rose junita yang telah memberikan semangat, dukungan moril maupun materil dan doa sehingga bisa menyelesaikan proposal Tugas Akhir ini.
3. Bapak Prof. Dr. Khairunnas M.Ag, selaku rektor UIN SUSKA Riau beserta kepada seluruh staf dan jajarannya.
4. Bapak Dr. Hartono, B.A, M.Pd selaku Dekan Fakultas Sains dan Teknologi UIN SUSKA Riau beserta kepada seluruh Pembantu Dekan, Staf dan jajarannya.
5. Ibu Dr. Zulfatri Aini S.T, M.T selaku ketua Program Studi Teknik Elektro Fakultas Sains dan Teknologi UIN SUSKA Riau.

6. Bapak Sutoyo, S.T, M.T selaku sekretaris program studi Teknik Elektro Fakultas Sains dan Teknologi UIN SUSKA Riau.
7. Bapak Aulia Ullah, S.T., M.Eng selaku Penasehat Akademik program studi Teknik Elektro Fakultas Sains dan Teknologi UIN SUSKA Riau.
8. Bapak Dr Haris Simaremare S.T., M.Eng selaku dosen pembimbing yang telah banyak meluangkan waktu serta pemikirannya dengan ikhlas dalam memberikan penjelasan dan masukan yang sangat berguna sehingga penulis menjadi lebih mengerti dalam menyelesaikan Tugas Akhir ini.
9. Bapak Abdillah, S.Si, MIT selaku penguji I dan Bapak Oktaf Brillian Kharisma, S.T., M.T selaku penguji II yang telah bersedia meluangkan waktu untuk memberikan kritikan dan saran yang sangat membangunkan terhadap penulis.
10. Bapak dan ibu dosen Program Studi Teknik Elektro yang telah memberikan bimbingan dan curahan ilmu sehingga bisa menyelesaikan Tugas Akhir ini.
11. Komputer 2014 serta teman-teman 2014 lainnya yang telah memberi dukungan kepada penulis dalam meyelesaikan Tugas Akhir ini serta teman-teman penulis yang lainnya yang tidak dapat penulis tulis satu persatu yang telah membantu dan memberi dorongan, motivasi dan sumbangan pemikiran dalam menyelesaikan Tugas Akhir ini.

Semoga bantuan yang telah diberikan baik moril maupun materil mendapat balasan pahala dari Allah SWT, dan sebuah harapan dari penulis semoga Tugas Akhir ini dapat bermanfaat bagi penulis dan para pembaca semua pada umumnya.

Semua kekurangan hanya datang dari penulis dan kesempurnaan hanya milik Allah SWT, hal ini dapat membuat penulis menyadari bahwa dalam pembuatan Tugas Akhir ini masih jauh dari kata kesempurnaan karena keterbatasan kemampuan, pengalaman dan pengetahuan penulis. Untuk ini penulis mengharap kritik dan saran dari semua pihak yang bersifat positif dan membangun demi kesempurnaan Tugas Akhir ini.

Pekanbaru, 15 Januari 2022

Penulis,

Ade Ibrahim

DAFTAR ISI

Halaman

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR SIMBOL

X dan Y = koordinat citra digital

M dan N = dimensi dari citra digital

S_{xy} = *stego* citra digital

C_{xy} = *cover* citra digital

DAFTAR SINGKATAN

DWT	= *Discrete Wavelet Transform*
BMP	= *Bitmap*
JPEG	= *Joint Photographic Group*
PNG	= *Portable Network Graphic*
MSE	= *Mean Square Error*
BER	= *Bit Error Rate*
CER	= *Character Error Rate*
PSNR	= *Peak Signal to Noise Ratio*
GUI	= *Graphical User Interface*

BAB I
PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi dan komunikasi telah berkembang pesat, memberikan pengaruh yang besar bagi kehidupan manusia sebagai makhluk sosial. Dengan komunikasi seseorang dapat berinteraksi dan bertukar informasi dengan orang lain, tetapi disisi lain komunikasi dapat menjadi ancaman yang dapat membahayakan informasi tersebut. Padahal kerahasiaan dan keamanan suatu informasi menjadi sesuatu yang sangat penting. Untuk itu, dibutuhkan suatu sistem yang mengamankan dan melindungi informasi tersebut. Sehingga dibuat suatu metode komunikasi yang dirancang sedemikian rupa supaya informasi yang disampaikan menjadi lebih aman. Salah satunya dengan menggunakan teknik steganografi. Steganografi merupakan suatu teknik menyembunyikan pesan ke dalam media lain sehingga keberadaan suatu pesan tidak diketahui oleh orang lain [1]. Terdapat beberapa metode pada steganografi, salah satunya adalah metode *Discrete Wavelet Transform* (DWT). Teknik ini membagi citra menjadi *subband-subband* yang memiliki 2 frekuensi tinggi dan frekuensi rendah.

Pengolahan citra adalah pemrosesan citra, khususnya menggunakan komputer untuk menghasilkan citra dari manipulasi citra sebelumnya, sehingga citra tersebut lebih mudah diinterpretasikan baik oleh manusia maupun mesin. Metode DWT merupakan konsep yang sederhana, dalam hal ini citra yang ditransformasikan didekomposisi terlebih dahulu menjadi sub-sub image sesuai dengan level (tingkatan) transformasi yang diinginkan. Pengolahan citra digital dalam bidang dekomposisi citra (*image compression)* berbasis transformasi *wavelet* (gelombang singkat) didasari bahwa koefisien-koefisien hasil proses transformasi *wavelet* bertujuan meminimalkan kebutuhan memori dalam merepresentasikan citra digital[2].

Citra hasil kompresi yang baik adalah yang cocok dengan kebutuhan pengiriman dan penyimpanan[3]. Biasanya Transmisi ini banyak dilakukan dengan cara pengiriman dalam jaringan. Citra ditransmisikan dalam jaringan untuk menyampaikan informasi terhadap orang yang membutuhkannya. Sebuah citra yang memiliki banyak informasi atau kapasitas besar akan ditransmisikan dalam jaringan dengan membutuhkan sumber daya yang lebih besar[4]. Kompresi citra digital merupakan sebuah cara pembuktian untuk menentukan teknik kompresi mana yang lebih baik untuk digunakan. Salah satunya teknik kompresi

dengan metode *Discrete Wavelet Transform (DWT). Discrete Wavelet Transform (DWT)* terbagi lagi menjadi 2 diantaranya adalah *Discrete Wavelet Transform* 1-dimensi *(DWT 1-D)* dan *Discrete Wavelet Transform* 2-dimensi (DWT 2-D). Pada teknik DWT 2-D dekomposisi perataan dan pengurangan sama dengan teknik *Discrete Wavelet Transform* (DWT) biasa, hanya saja proses dekomposisinya dilakukan dalam dua tahap[2].

Saat ini teknik steganografi DWT 2-D mulai umum digunakan. Namun teknik pengungkapan informasinya telah diketahui khalayak banyak sehingga dianggap kurang efektif sebagai metode komunikasi yang aman. Untuk itu kita mampu meningkatkan keamanan teknik ini dengan menambahkan algoritma kriptografi [5]. Algoritma kriptografi ialah suatu teknik yang memiliki aspek seperti kerahasiaan, otektikasi, dan integritas. Dengan aspek-aspek tersebut maka diharapkan setiap orang yang ingin bertukar informasi dapat terjamin keamanan nya [6].

Banyak algoritma kriptografi yang dapat digunakan dalam penyandian informasi, salah satunya adalah kriptografi Rivest-Shamir-Adleman (RSA) kunci asimetris. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor faktor prima. Selama pemfaktoran bilangan besar menjadi faktor prima belum ditemukan algoritma yang tepat. Maka selama itu pula keamanan algoritma RSA tetap terjamin [7]. Dengan menambahkan algoritma kriptografi RSA pada teknik steganografi DWT 2-D, diharapkan kerahasian informasi akan tetap terjaga meskipun orang lain mampu mengungkap informasi tersebut dari media steganografi.

Berdasarkan latar belakang diatas maka penulis mengadakan penelitian serta pengujian dengan melakukan penggabungan metode kriptografi dan metode steganografi dalam meningkatkan keamanan proses pertukaran pesan. Penerapan metode kriptografi yang dipilih ialah algoritma kriptografi RSA dengan metode steganografi citra *Discrete wavelet Transform 2-dimensi* (DWT 2-D) dalam menambahkan variasi model keamanan yang baru.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan diatas, maka beberapa rumusan masalah dari penelitian ini yaitu :

1. Bagaimana mengenkripsi pesan teks menggunakan algoritma kriptografi RSA?
2. Bagaimana proses penyisipan pesan pada media citra digital menggunakan metode steganografi *Discrete Wavelet Transform Two Dimension (2-D DWT)*?

3. Bagaimana Proses steganografi dan kriptografi sehingga dapat meningkatkan keamanan pesan?
4. Bagaimana performansi dari sistem steganografi dan kriptografi yang akan dibuat?

1.3 Tujuan Penelitian

Adapun tujuan dari penyusunan tugas akhir ini adalah sebagai berikut :

1. Melakukan proses enkripsi pesan teks menggunakan algoritma kriptografi RSA.
2. Melakukan proses penyisipan pesan teks kedalam media citra digital menggunakan metode steganografi *Discrete Wavelet Transform Two Dimension (2-D DWT).*
3. Menggabungkan teknik steganografi dan kriptografi untuk meningkatkan keamanan pesan rahasia.
4. Menguji performansi dan menganalisis sistem steganografi dan kriptografi yang dibuat.

1.4 Batasan Masalah

Beberapa Batasan masalah pada penelitian ini antara lain adalah sebagai berikut :

1. Penelitian ini hanya dilakukan sebatas proses enkripsi pesan teks menggunakan algoritma kriptografi RSA.
2. Penelitian ini merupakan proses penyisipan pesan teks kedalam media citra digital menggunakan metode steganografi *Discrete Wavelet Transform Two Dimension (2-D DWT).*
3. Citra digital yang digunakan adalah citra digital berwarna dengan format 24-bit (BMP).
4. Pesan yang disisipkan berupa teks dengan format (*.txt).
5. Parameter Performansi dalam penelitian ini menggunakan PSNR, MOS, BER, dan CER.

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah dapat meningkatkan keamanan dan memperkuat suatu file dalam proses pengiriman suatu informasi yang memiliki sifat rahasia, agar tidak dapat diterima dan dipecahkan oleh pihak lain yang tidak berhak menerimanya. Penelitian ini menggunakan Matlab yang berguna memudahkan pengirim dan penerima dalam mengirim dan menerima informasi.

BAB II
TINJAUAN PUSTAKA

2.1 Studi Literatur

Pada tugas akhir ini dilakukan *studi literatur* yang mengumpulkan teori-teori dari sumber-sumber yang terkait dengan penelitian ini. Sumber-sumber pada penelitian ini bisa berasal dari buku, jurnal, *paper*, ataupun dari sumber-sumber yang terkait dan mendukung pada penelitian ini. Penelitian tentang menggabungkan teknik steganografi *discrete wavelet transform* 2 *dimensi* (2-D DWT) dan algoritma kriptografi *RSA* pada perancangan dan analisis keamanan pesan bertujuan untuk meningkatkan kemananan pesan dan kerahasian data yang lebih *kompleks.* Berikut dipaparkan beberapa penelitian terdahulu tentang keamanan dan kerahasian data.

Penelitian yang berjudul Pedekomposisian Citra Digital dengan Algoritma DWT. Pada penelitian ini proses dekomposisi menggunakan algoritma DWT. Adapun dekomposisi citra berbasis DWT menggunakan sistem perhitungan dengan dekomposisi dengan arah baris dan dekomposisi dengan arah kolom. Proses dekompsosisi menggunakan rumus perataan dan pengurangan dengan menghitung nilai rata-rata dua pasang data. Hasil penelitian ini menunjukan bahwa dekomposisi citra digital berbasis DWT diharapkan mampu menjadi salah satu algoritma yang bermanfaat dalam teknologi pencitraan untuk dekomposisi citra digital, karna pada algoritma ini citra dekomposisi yang dihasilkan merupakan hasil dari empat pembagian *subband* yaitu : *Low-Low* (LL), *subband High-Low* (HL), *subband Low-High* (LH), *subband High-High* (HH)[2].

Penelitian yang sama juga dilakukan oleh Aditya Mahmud Faza, dkk (2016) yang berjudul Analisis Kinerja Kompresi Citra Digital dengan Komparasi DWT, DCT, dan *Hybrid* (DWT-DCT). Hasil Penenlitian terhadap penerapan transformasi DCT, transformasi DWT, dan *Hybrid* sebagai penggabungan dari kedua transformasi sebelumnya dalam proses kompresi data citra digital. Proses kompresi dilakukan untuk menekan komsumsi sumber daya memori, mempercepat proses transmisi citra digital. Proses kompresi yang dilakukan dapat menghasilkan nilai *mean square, peak signal to noise ratio* dan waktu yang dibutuhkan untuk proses kompresi dari masing-masing transformasi. Nilai tersebut sebagai parameter untuk tahap komparasi kompresi sehingga pengguna dapat menentukan kinerja kompresi dari setiap transformasi dan dapat menentukan jenis transformasi yang paling baik digunakan untuk proses

citra digital. Penelitian menghasilkan algoritma DCT, DWT, dan *Hybrid* (DCT-DWT) dapat mengkompresi *file* citra digital dengan ekstensi *file* JPG. Semakin rendah nilai MSE menandakan semakin baik kualitas citra terkompresi. Hasil anilisis pada algoritma DWT memiliki nilai MSE terendah dengan nilai rata-rata MSE sebesar 28,02 dB, dan DCT memiliki nilai terendah kedua dengan nilai MSE sebesar 28,55 dB. Semakin tinggi nilai PSNR, maka semakin baik kualitas citra terkompresi. Hasil analisis mengenai nilai PSNR dihasilkan algoritma DWT memiliki nilai PSNR tertinggi dengan nilai rata-rata PSNR sebesar 28,58 dB, dan DCT memiliki nilai tertinggi kedua dengan nilai PSNR sebesar 36,03 dB. Tingkat efensiensi memori *file* hasil kompresi dakur dari besarnya rasio kompresi yang dihasilkan. Semakin besar nilai rasio kompresi menandakan semakin efesien memori hasil kompresi. Algoritma *Hybrid* memiliki nilai rata-rata rasio kompresi tertinggi dengan nilai sebesar 70,51 %. Besarnya persentase kompresi *Hybrid* menjadikan hasil kompresi memiliki ukuran *file* paling kecil. Lamanya waktu yang dibutuhkan kompresi menandakan proses kinerja kompresi yang membutuhkan energi besar. Dalam kasus perbandingan algoritma, waktu terpendek menjadi waktu terbaik untuk dipilih. Waktu terpendek untuk proses kompresi ditunjukan oleh algoritma DWT dengan rata-rata waktu kompresi 2,71 Detik. Waktu tercepat kedua didapat oleh algoritma *Hybrid* dengan rata-rata waktu kompresi 2,94 Detik[4].

Dalam penelitian berjudul Pengaman Data User Pada *Database* Menggunakan Kriptografi *Rivest-Shamir-Adleman* (RSA) dan *Cipher Block Chaining* (CBC) memaparkan hasil penelitiannya dengan menggunakan salah satu teknik *SQL Injection* yaitu memanfaatkan *bug* yang ada *URL,* pencuri memungkinkan untuk melihat isi dari *database*, jika tidak dilakukan pengamanan maka data tersebut dapat bebas dilihat oleh orang tersebut. Penelitian ini telah membangun sistem pengamanan terhadap data user dengan metode RSA dan CBC. Pengaman ini memungkinkan isi data terenkripsi sehingga pada saat sistem diretas dengan *SQL Injection* maka data tidak dapat dipahami, data yang telah dienkripsi dapat dideskripsi kembali untuk menjaga keaslian data. Hasil dari pengujian yang telah dilakukan, sistem ini berhasil melakukan enkripsi pada data tabel sehingga data menjadi karakter acak yang tidak dapat dipahami. Pada saat *database* diretas dengan *SQL Injection* pun data yang tampil adalah data yang telah terenkripsi. Pengujian dilakukan untuk melihat hasil dari proses pengamanan, pengujian yang dilakukan adalah melakukan pengamanan terlebih dahulu terhadap tabel user pada salah satu sitem, kemudian sistem tersebut diretas untuk melihat hasil pengamanan yang dilakukan[5].

Selanjutnya penelitian yang berjudul Perancangan dan Analisis Keamanan Pesan Menggunakan Teknik Steganografi *Discrete Wavelet Transform* (DWT) dan *Algoritma kriptografi Rivest-Shamir-Adleman* (RSA). Pada penelitian ini teknik *steganografi* akan dikombinasikan dengan teknik *kriptografi*. Perancangan sistem yang dibangun dilakukan dengan menggabungkan penerapan metode algoritma *kriptografi* RSA dalam menyandikan pesan, serta menerapkan metode steganografi citra dalam menyembunyikan pesan tersandi yang dihasilkan tadi dalam sebuah citra warna (RGB) dalam domain DWT. Hasil yang akan diperoleh dari tugas akhir adalah sebuah citra yang memiliki pesan terenkrepsi pada *subband-subband* frekuensi citra tersebut. Berdasarkan dari beberapa pengujian yang telah dilakukan pada sistem, telah diperoleh beberapa hasil performasi dengan nilai rata-rata meliputi *Avalanche Effect* sebesar 43,486 %, *Peak Signal To Noise Ratio* (PSNR) sebesar 67,2755 %, *Bit Error Rate* (BER) sebesar 0 %, *Character Error Rate* (CER) sebesar 0 %[8].

Selanjutnya Penelitian yang Berjudul Analisis Kriptografi Simetris AES dan Kriptografi Asimetris RSA pada enkripsi citra digital. Proses enkripsi dan dekripsi pada algoritma RSA dipengaruhi oleh pasangan kunci. Dimana, semakin besar kunci publik maka akan semakin lama proses enkripsi, dan semakin besar kunci privat maka akan semakin lama proses dekripsi. Proses enkripsi dan dekripsi pada algoritma AES dipengaruhi oleh panjang kunci. Dimana semakin panjang kunci yang digunakan maka akan semakin banyak putaran yang dilalui dan semakin lama proses enkripsi dan dekripsi berlangsung. Besarnya dimensi plain image yang digunakan mempengaruhi waktu enkripsi dan dekripsi. Semakin besar dimensi citra, maka akan semakin besar waktu yang dibutuhkan, hal ini dikarenakan oleh jumlah pixel yang ada juga akan semakin besar sehingga menambahkan waktu proses enkripsi dan dekripsi. Algortima RSA menghasilkan nilai MSE=0 dan PSNR=inf. Ini menandakan bahwa tidak terjadi perubahan dan error antara gambar asli dengan gambar hasil dekripsi. Sedangkan algoritma AES menghasilkan nilai MSE rata-rata=9,7 dengan PSNR=36,56 dB. Algoritma RSA lebih unggul pada kualitas dekripsi dan kecepatan proses enkripsi dan dekripsi, sedangkan algoritma AES lebih unggul pada kualitas enkripsi[9].

2.2 Kriptografi

Kata kriptografi berasal dari bahasa Yunani. Dalam bahasa Yunani kriptografi terdiri dari dua buah kata yaitu *cryptos* dan *graphia*. Kata *crypto* berarti rahasia (*secret*) sedangkan *graphia* berarti tulisan. Sehingga makna dari kriptografi adalah tulisan rahasia. Menurut terminologinya kriptografi adalah ilmu yang mempelajar tentang bagaimana menjaga kerahasiaan suatu pesan, agar isi pesan yang disampaikan tersebut aman sampai ke penerima pesan. Dalam ilmu kriptografi suatu pesan yang telah disandikan disebut dengan *plaintext*, sedangkan pesan yang telah disandikan sehingga tidak bermakna lagi yang bertujuan agar pesan tidak dapat dibaca oleh pihak yang tidak berhak disebut *chipertext*. Lalu dalam ilmu kriptografi terdapat istilah enkripsi dan deskripsi. Enkripsi adalah proses menyandikan *plaintext* menjadi *chipertext*. Sedangkan proses mengembalikan *chipertext* menjadi *plaintext* semula disebut sebagai Deskripsi[19].

Aspek-aspek keamanan didalam kriptografi adalah sebagai berikut :

1. Kerahasiaan adalah layanan yang digunakan untuk menjaga isi informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka atau mengupas informasi yang telah disandi.
2. Integritas adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain ke dalam data yang sebenarnya.
3. Autentikasi adalah berhubungan dengan identifikasi atau pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keasliannya, isi datanya, waktu pengirimannya, dan lain lain.
4. Non-repudiasi adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/terciptanya suatu informasi oleh yang mengirimkan/membuat.

Sistem kriptografi (*cryptosystem*) adalah kumpulan yang terdiri dari algoritma, semua *plaintext* dan *chiphertext*. Hybrid *cryptosystem* yaitu kombinasi kriptografi dengan menggabungkan algoritma simetris dengan algoritma asimetris atau dengan *public key* dengan *private key*[20].

2.2.1 Kriptografi Simetri (*symetric cryptosystem*)

Algoritma simetris atau disebut juga algoritma Kriptografi konvensional adalah algoritma yang menggunakan kunci yang sama untuk proses enkripsi dan proses deskripsi. Algoritma kriptografi simetris dibagi menjadi 2 kategori yaitu algoritma aliran (stream Chipers) dan algoritma blok (Block Chipers). Pada algoritma aliran, proses penyandiannya berorientasi pada satu bit atau satu byte data. Sedang pada algoritma blok, proses penyandiannya berorientasi pada sekumpulan bit atau byte data (perblok). Contoh algoritma kunci Simetris adalah DES (Data Encryption Standard), Blowfish, TwoFish, MARS, IDEA, 3DES (DES diaplikasikan 3 kali), AES (Advanced Encryption Standard) yang bernama asli Rijndael.

2.2.2 Kriptografi Asimetri (*Assymetric cryptosystem*)

Dalam kriptografi asimetris digunakan dua buah kunci yang berbeda dalam proses enkripsi dan dekripsinya. Suatu kunci yang disebut kunci public (*public key*) yang dapat dipublikasikan, sedangkan kunci yang lain disebut kunci privat (*private key*) yang harus dirahasiakan. Contoh dari sistem ini antara lain RSA *Scheme* dan Merkie Hellman *Scheme*.

Kriptografi asimetris dikembangkan para pakar kriptografi untuk menanggulangi kesulitan distribusi kunci pada kriptografi kunci simetris. Distribusi kunci pada kriptografi kunci asimetris sangat mudah, karna kunci enkripsi bersifat *public* atau umum maka distribusi kunci dapat dilakukan dijalur mana saja bahkan jalur yang ingin diamankan sekalipun[21].

Terdapat banyak algoritma yang dikembangkan pakar-pakar kriptografi untuk algoritma kunci asimetris, diantaranya : Algoritma RSA, Algoritma McElice, Algoritma Rabin, Algoritma Knapsack, Algoritma LUC, Algoritma El Gamal. Kelebihan dari kriptografi asimetris adalah :

1. Hanya kunci privat yang perlu dijaga kerahasiaannya oleh entitas yang berkomunikasi. Tidak ada kebutuhan mengirim kunci privat sebagimana pada kunci simetris.
2. Pasangan kunci privat dan kunci public tidak perlu diubah dalam jangka waktu yang sangat lama.
3. Dapat digunakan dalam pengamanan pengiriman kunci simetris
4. Beberapa algoritma kunci public dapat digunakan untuk memberi tanda tangan digital pada pesan.

Kelemahan dari kriptografi asimetris adalah :

1. Proses enkripsi dan dekripsi umumnya lebih lambat dari algoritma simetris, karna menggunakan bilangan yang lebih besar dan operasi bilangan yang besar.
2. Ukuran *chipertext* lebih besar daripada *plaintext*
3. Ukuran kunci relatif lebih besar daripada kunci simetris

2.2.2.1 Kriptografi RSA

Algoritma RSA pertama kali ditemukan oleh Ron Rivest, Adi Shamir, dan Leonard Adleman. RSA singkatan dari huruf depan tiga orang yang menemukannya pada tahun 1977 di MIT (*Massachussets Institute of Technology*). Pada tahun 1983, *Massachussets Institute of Technology* menerima hak paten atas sebuah makalah yang berjudul "*Cryptography Communication System And Method*" yang mengaplikasikan pengguna algoritma kriptografi RSA[22]. Dari sekian banyak algoritma kriptografi kunci-publik yang pernah dibuat, algoritma yang popular adalah algoritma RSA. Langkah dalam algoritma RSA adalah membuat pasangan kunci yaitu kunci *public* dan kunci *private*. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Pemfaktoran yang dilakukan memperoleh kunci *private*. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan algoritma yang mangkus, maka selama itu pula keamanan algoritma RSA tetap terjamin[23].

Pada algoritma RSA terdapat tiga langkah utama yaitu *keygeneration* (pembangkit kunci), *enkripsi* dan *dekripsi*.kunci pada RSA mencangkup dua buah kunci, yaitu *public key* dan *private key*. *Public key* digunakan untuk melakukan enkripsi, dan dapat diketahui oleh orang lain. Sedangkan *private key* tetap dirahasiakan dan digunakan untuk melakukan dekripsi. Sistem kriptografi yang baik adalah sistem kriptografi yang memang dirancang sedemikian rupa sehingga sulit dipecahkan. Secara teori sebuah metode kriptografi dengan sebuah kunci akan dapat dipecahkan dengan mencoba semua kemungkinan rangkaian kunci. Satu-satunya cara yang diketahui untuk mendobrak sandi RSA adalah dengan mencoba satu-persatu berbagai kombinasi kunci dengan istilah *brute force attack*. Sebenarnya keamanan dari RSA banyak bergantung dari ukuran kunci yang digunakan yaitu dalam bit. Jadi semakin panjang ukuran kunci maka semakin sulit untuk dipecahkan. Algoritma RSA memiliki besaran sebagai berikut :

Tabel 2.1 Besaran Kriptografi Algoritma RSA[20]

1	p dan q bilangan prima	Rahasia
2	n = p x q	Tidak Rahasia
3	Φ(n) = (p-1)(q-1)	Rahasia
4	e (kunci enkripsi)	Tidak Rahasia
5	d (kunci dekripsi)	Rahasia
6	m (plainteks)	Rahasia
7	c (cipherteks)	Tidak Rahasia

RSA adalah suatu blok sandi rahasia tempat teks asli dan teks rahasia merupakan bilangan bulat antara 0 dan n-1 untuk beberapa n. Enkripsi dan dekripsi berasal dari beberapa bentuk berikut ini, untuk beberapa blok teks asli M dan blok teks rahasia C.

$C = M' \bmod n$

$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$

Blok pengirim maupun penerima harus mengetahui nilai e dan n, dan hanya penerima saja yang mengetahui nilai d. ini merupakan algoritma enkripsi kunci umum dengan kunci umum sebesar KU = {e,n} dan kunci khusus sebesar KR = {d,n}. agar algoritma ini bisa memenuhi syarat sebagai enkripsi kunci umum yang baik, maka harus memenuhi ketentuan-ketentuan sebagai berikut :

1. Kemungkinan menemukan nilai e, d, n sedemikian rupa sehingga $M^{ed} = \bmod n$ untuk semua $M < n$
2. Relative mudah menghitung M^e dan C^d untuk semua nilai $M < n$
3. Tidak mudah menghitung menentukan d, yang diberi e dan n

Dua ketentuan pertama bisa terpenuhi dengan mudah. Sedangkan ketentuan ketiga baru bisa terpenuhi untuk nilai e dan n yang besar.

Pembangkitan kunci

1. Memilih dua bilangan prima p, q. bilangan ini harus cukup besar (minimal 100 digit).

2. Menghitung n = p.q. bilangan n disebut *parameter security* (sebaiknya $p \neq q$, sebab jika p = q maka n = p^2 sehingga p dapat diperoleh dengan menarik akar pangkat dua dari n).
3. Menghitung $\varphi(n) = (p-1)(q-1)$.
4. Memilih bilangan bulat e dengan algoritma Euclid yaitu gcd $(\varphi(n),e)=1$; dimana $1 < e < \varphi(n)$.
5. Menghitung d dengan rumus $d = e-1 \bmod \varphi(n)$ atau $e, d, \equiv 1 \pmod{\varphi(n)}$. Perhatikan bahwa $e \cdot d \equiv 1 \pmod{\varphi(n)}$ ekivalen dengan $e \cdot d = 1 + k\ \varphi(n)$, sehingga secara sederhana d dapat dihitung dengan $d = (1 + k \cdot \varphi(n)) / e$
6. Kunci umum (kunci public) adalah KU = {e,n}
7. Kunci pribadi (kunci privat) adalah KR = {d,n}

Catatan : n tidak bersifat rahasia, sebab ia diperlukan dalam perhitungan enkripsi dan dekripsi

Enkripsi :

B mengenkripsi message M untuk A, yang harus dilakukan B :

1. Teks asli dengan syarat M < n
2. Ambil kunci public A yang otentik (n,e)
3. Representasikan message sebagai integer M dalam interval [0,n-1]
4. Teks rahasia didapat dari $C = M^e \pmod n$
5. Kirim C ke A

Dekripsi :

Untuk mendekripsi, A melakukan :

1. Gunakan kunci pribadi d untuk menghasilkan M
2. Teks rahasia adalah C
3. Teks asli didapat dari $M = C^d \pmod n$

2.3 Sistem Pengkodean Karakter

Komputer merupakan salah satu penemuan terbesar dalam peradaban manusia. Dengan adanya komputer berbagai aspek kegiatan bisa dilakukan dengan mudah. Tidak hanya itu, komputer juga bisa melakukan hal-hal yang sulit dikerjakan bagi manusia. Lantas tahukah Anda bagaimana komputer bekerja? Komputer bekerja dengan bahasa pemrograman yang rumit dan kompleks. Komputer sebenarnya memiliki bahasa tersendiri yang berbeda dengan manusia. Agar tetapi sebenarnya bahasa komputer hanya terdiri dari angka 0 dan 1. Bahasa dalam komputer dapat mengekspresikan setiap nilai numerik sebagai terjemahan biner yang tidak lain adalah operasi matematika sederhana. Komputer sendiri juga memiliki kode-kode khusus yang merujuk pada suatu karakter[25].

Gambar 2.1 Sistem Pengkodean Karakter

Dalam sistem komputer ada tiga kode karakter yang bisa kita pelajari, antara lain ASCII, EBCDIC dan Unicode. Berikut penjelasan singkat ketiganya.

2.3.1 EBCDIC (*Extended Binary Code Decimal Interchange Code*)

Extended Binary Code Decimal Interchange Code atau EBCDIC adalah standar yang dikembangkan oleh IBM di tahun 1950-an. Standar ini memakai 8 bit untuk setiap kode. EBCDIC pertama kali digunakan pada IBM System/360. Standar EBCDIC digunakan pada berbagai komputer mainframe.

2.3.2 UNICODE

Unicode adalah standar yang lebih baru dibandingkan dengan dua standar di atas. Standar Unicode dinyatakan dengan 16 bit untuk sebuah karakter. Oleh karenanya standar ini mencakup hingga 65.536 karakter. Dengan metode tersebut ada banyak sekali simbol dalam bahasa yang dimasukkan, seperti bahasa Arab atau Cina.

2.3.3 ASCII (*American Standard Code For Information Interchange*)

ASCII (*American Standard Code For Information Interchange*) adalah Kode Standar Amerika untuk Pertukaran Informasi yang merupakan suatu standar internasional dalam kode huruf dan simbol seperti Hex dan Unicode tetapi ASCII lebih bersifat universal. Kode ASCII sebenarnya memiliki komposisi bilangan biner sebanyak 7 bit. Namun, ASCII disimpan sebagai sandi 8 bit dengan menambakan satu angka 0 sebagai bit significant paling tinggi. Bit tambahan ini sering digunakan untuk uji prioritas. Karakter control pada ASCII dibedakan menjadi 5 kelompok sesuai dengan penggunaan yaitu berturut-turut meliputi *logical communication*, *Device control*, *Information separator*, *Code extention*, dan *physical communication*. Kode ASCII ini banyak dijumpai pada papan ketik (*keyboard*) komputer atau instrumen-instrumen digital. Jumlah kode ASCII adalah 255 kode. Kode ASCII 0-127 merupakan kode ASCII untuk manipulasi teks; sedangkan kode ASCII 128-255 merupakan kode ASCII untuk manipulasi grafik. Kode ASCII sendiri dapat dikelompokkan lagi kedalam beberapa bagian:

1. Kode yang tidak terlihat simbolnya seperti Kode 10(Line Feed), 13(Carriage Return),8(Tab), 32(Space)
2. Kode yang terlihat simbolnya seperti abjad (A..Z), numerik (0..9), karakter khusus (~!@#$%^&*()_+?:”{})
3. Kode yang tidak ada di keyboard namun dapat ditampilkan. Kode ini umumnya untuk kode-kode grafik.

Dalam pengkodean kode ASCII memanfaatkan 8 bit. Pada saat ini kode ASCII telah tergantikan oleh kode UNICODE (*Universal Code*). UNICODE dalam pengkodeannya memanfaatkan 16 bit sehingga memungkinkan untuk menyimpan kode-kode lainnya seperti kode bahasa Jepang, Cina, Thailand dan sebagainya. Pada papan keyboard, aktifkan *numlock*, tekan tombol ALT secara bersamaan dengan kode karakter maka akan dihasilkan karakter tertentu. Misalnya: ALT + 44 maka akan muncul karakter koma (,). Mengetahui kode-kode ASCII sangat bermanfaat misalnya untuk membuat karakter-karakter tertentu yang tidak ada di keyboard. Tabel berikut berisi karakter-karakter ASCII . Dalam sistem operasi Windows dan MS-DOS, pengguna dapat menggunakan karakter ASCII dengan menekan tombol Alt+[nomor nilai ANSI (desimal)]. Sebagai contoh, tekan kombinasi tombol Alt+87 untuk karakter huruf latin "W" kapital.

Tabel 2.2 Karakter-Karakter ASCII[25].

ASCII table

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(nul)	0	0000	0x00	(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
(soh)	1	0001	0x01	!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
(stx)	2	0002	0x02	"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
(etx)	3	0003	0x03	#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
(eot)	4	0004	0x04	$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
(enq)	5	0005	0x05	%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
(ack)	6	0006	0x06	&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
(bel)	7	0007	0x07	'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(bs)	8	0010	0x08	(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
(ht)	9	0011	0x09	)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
(nl)	10	0012	0x0a	*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
(vt)	11	0013	0x0b	+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
(np)	12	0014	0x0c	,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
(cr)	13	0015	0x0d	-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
(so)	14	0016	0x0e	.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
(si)	15	0017	0x0f	/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
(dle)	16	0020	0x10	0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
(dc1)	17	0021	0x11	1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
(dc2)	18	0022	0x12	2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
(dc3)	19	0023	0x13	3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
(dc4)	20	0024	0x14	4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
(nak)	21	0025	0x15	5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
(syn)	22	0026	0x16	6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
(etb)	23	0027	0x17	7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
(can)	24	0030	0x18	8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
(em)	25	0031	0x19	9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
(sub)	26	0032	0x1a	:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
(esc)	27	0033	0x1b	;	59	0073	0x3b	[	91	0133	0x5b	{	123	0173	0x7b
(fs)	28	0034	0x1c	<	60	0074	0x3c	\	92	0134	0x5c	|	124	0174	0x7c
(gs)	29	0035	0x1d	=	61	0075	0x3d	]	93	0135	0x5d	}	125	0175	0x7d
(rs)	30	0036	0x1e	>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
(us)	31	0037	0x1f	?	63	0077	0x3f	_	95	0137	0x5f	(del)	127	0177	0x7f

2.4 Analisis dan Pembahasan Bilangan Prima

Bilangan prima (Prime Number) adalah bilangan yang habis dibagi oleh angka 1 dan dirinya sendiri. Bilangan prima merupakan bilangan asli. Bilangan asli adalah bilangan bulat positif yang bukan nol. Contohnya dari 1, 2, 3, ... , tak terhingga positif. Bilangan prima dimulai dari angka 2 dan berlanjut ke angka seterusnya. Bilanga 2 hanya dapat difaktorkan menjadi 2 dan 1 (2 = 2 x 1). Bilangan 2 adalah bilangan prima terkecil dan satu satunya bilangan prima genap[26].

2.4.1 Algoritma Bilangan Prima

Ketika kita akan mencari sebuah bilangan prima, misalnya kita akan menentukan apakah angka 5 adalah bilangan prima atau tidak. Maka langkah pertama yang kita lakukan adalah mencari sisa bagi (mod) angka 5 dengan angka 2, Apabila habis dibagi 2 maka bilangan 5 bukan bilangan prima. Kemudian kita bagi kembali dengan angka 3 dan terakhir dengan angka 4. Dari sisa pembagian 2, 3, dan 4, bilangan 5 tidak habis dibagi. Dan hanya habis dibagi oleh dirinya sendiri. Jadi 5 termasuk bilangan prima.

2.4.2 Algoritma Bilangan Prima dalam Pemrograman

Kita coba terapkan algoritma diatas kedalam sebuah logika pemrogaman.

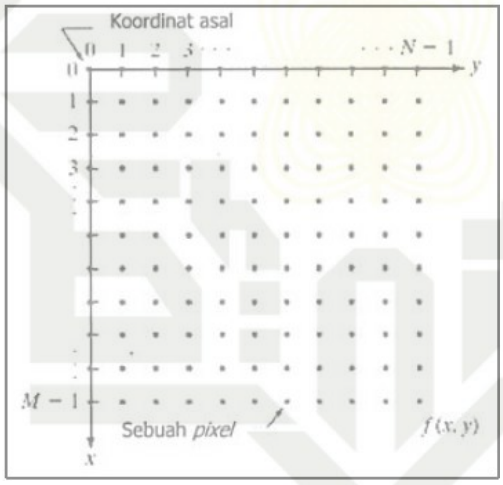
1. Pertama kita definisikan inputan misalnya dengan x = 5.
2. Kemudian kita bisa mendefinisikan jika 2 adalah bilangan prima.
3. Kemudian kita akan melakukan sisa pembagian (mod) x dengan bilangan 2 sampai x-1. Jadi 5 akan dibagi oleh 2, 3, dan 4.
4. Apabila hasil mod (sisa bagi) x dengan setiap bilangan x-1 adalah sama dengan 0, maka bilangan tersebut bukanlah bilangan prima.
5. Jika tidak, maka hasilnya adalah bilangan tersebut adalah bilangan prima.

2.5 Citra Digital

Citra merupakan fungsi terus menerus *(continue)* atas intensitas cahaya pada bidang dua dimensi. Sumber cahaya menerangi objek, objek memantulkan kembali seluruh atau sebagian berkas cahaya kemudian ditangkap oleh alat optis atau elektro-optis[12]. Citra didefenisikan sebagai fungsi $f(x,y)$ dimana x dan y adalah sebuah koordinat pada bidang dan amplitude dari f pada pasangan koordinat adalah intensitas atau sebuah tingkatan keabu-abuan dari suatu citra

pada titik tersebut. Jika *x*, *y*, dan nilai intensitas dari *f* tersebut bernilai diskrit, sehingga, citra tersebut dinamakan citra *digital*[3].

Pada jurnal lainnya defenisi citra adalah suatu representasi, kemiripan, atau imitasi dari suatu obyek atau benda. Sebuah citra mengandung informasi tentang obyek yang di representasikan. Citra dapat dikelompokkan menjadi citra tampak dan citra tak tampak. Untuk dapat dilihat mata manusia, citra tak tampak harus dirubah menjadi citra tampak, misalnya dengan menampilkannya dimonitor, dicetak dikertas dan sebagainya. Salah satu citra tak tampak adalah citra digital. Citra dapat juga didefenisikan sebagai gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses sampling. Gambar analog dibagi menjadi N baris dan M kolom sehingga menjadi gambar diskrit. Persilangan antara baris dan kolom tertentu disebut dengan *piksel* [13]. Citra adalah gambar pada bidang dua dimensi. Citra dibentuk dari persegi empat yang teratur sehingga jarak horizontal dan vertikal antara *piksel* satu dengan yang lain adalah sama pada seluruh bagian citra. Indeks *x* bergerak kebawah dan indeks *y* bergerak kekanan.



Gambar 2.2 Kordinat Pada Citra Digital[14].

Citra digital dapat diwakili oleh sebuah matriks yang terdiri dari baris (*N*) dan kolom (*M*) dimana perpotongan antara kolom dan baris disebut *piksel*, yaitu elemen terkecil dari sebuah citra. *Piksel* mempunyai dua parameter, yaitu *f*(*x*,*y*) pada kordinat citra (*x*.*y*) merupakan besar dari intensitas atau warna dari *piksel* pada titik itu[3].

N = jumlah baris $0 \leq x \leq N - 1$.

M= jumlah kolom $0 \leq y \leq M - 1.$

L = maksimal warna intensitas $0 \leq f(x,y) \leq L - 1.$

Citra digital yang berukuran *N* x *M* dapat dinyatakan dengan matriks yang berukuran *N* baris dan *M* kolom seperti pada gambar berikut :

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ f(1,0) & f(1,1) & \cdots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.3 Matriks berukan N x M [3].

2.6 Format File Citra

Sebuah format *file* harus dapat menyatukan kualitas citra, ukuran *file* dan kompabilitas dengan berbagai aplikasi. Format *file* citra standar yang digunakan saat ini terdiri dari beberapa jenis. Format-format ini digunakan untuk menyimpan citra dalam sebuah *file*. Setiap format memiliki karakteristik masing-masing. Ini adalah contoh format umum yaitu : *Bitmap* (.bmp), *tagged image format* (.tif,tiff), *Portable Network Graphics* (.png), JPEG (.jpg), dll. Ada dua jenis format *file* citra yang sering digunakan dalam pengolahan citra, yaitu citra bitmap dan citra vektor[3].

Pada citra bitmap ini sering disebut juga citra raster, citra bitmap ini menyimpan data kode citra secara digital dan lengkap (cara penyimpanannya adalah per *pixel*). Citra bitmap ini dipresentasikan dalam bentuk matriks atau dipetakan dengan menggunakan bilangan biner atau sistem bilangan yang lain. Citra ini memiliki kelebihan untuk memanipulasi warna, tetapi untuk mengubah objek lebih sulit. Tampilan bitmap mampu menunjukkan kehalusan gradasi bayangan dan warna dari sebuah gambar. Tetapi apabila tampilan diperbesar maka tampilan dimonitor akan tampak pecah-pecah. Contoh format *file* citra antara lain adalah BMP, GIFF, TIF, WPG, IMG, dll. Sedangkan pada format *file* citra vektor merupakan vektor yang dihasilkan dari perhitungan matematis dan tidak terdapat *pixel*, yaitu data yang tersimpan dalam bentuk vektor posisi, dimana yang tersimpan hanya informasi vektor posisi dengan bentuk sebuah fungsi. Pada citra vektor, mengubah warna lebih sulit dilakukan, tetapi membentuk objek dengan cara mengubah nilai lebih mudah. Oleh karena itu, bila citra diperbesar atau diperkecil, kualitas citra relative tetap baik dan tidak berubah. Citra vektor biasanya dibuat menggunakan aplikasi-aplikasi seperti *Paint, corelDRAW, Adobe Illustrator, Autocad, dll*[3].

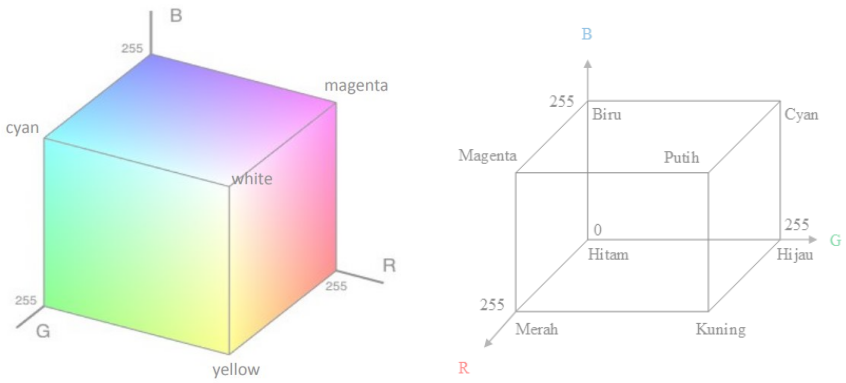
2.6.1 Citra Berwarna

Warna secara utuh bergantung pada sifat pantulan (*reflectance)* suatu objek. Warna yang dilihat merupakan yang dipantulkan, sedangkan yang lain diserap. Sehingga sumber sinar perlu diperhitungkan, begitu pula sifat alami sistem visual manusia ketika menangkap suatu warna. Model warna merupakan cara standar untuk menspesifikasi suatu warna tertentu, dengan mendefenisikan suatu sistem kordinat 3D, dan suatu ruang bagian yang mengandung semua warna yang dapat dibentuk ke dalam suatu model akan berhubungan ke suatu titik tunggal dalam suatu ruang bagian yang didefenisikannya. Citra berwarna, atau biasa dinamakan RGB merupakan jenis citra yang menyajikan warna dalam bentuk 3 komponen yaitu, R (merah), G (hijau), B (biru). Setiap komponen warna menggunakan 8 bit (nilainya berkisar antara 0 sampai dengan 255). Dengan demikian, kemungkinan warna yang bisa disajikan mencapai 255 x 255 x 255 atau 16.581.375 warna. Pada tabel 3 dapat dilihat contoh warna dan nilai R, G, dan B

Tabel 2.3 Warna dan Nilai Penyusun Warna[3].

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255

Perlu diketahui juga juga bahwa, sebuah warna tidak hanya dinyatakan dengan komposisi R, G, dan B tunggal. Pada tabel 1 terlihat bahwa warna merah mempunyai R=255, G=0, B=0. Namun, komposisi R = 244, G = 1, B = 1 juga berwarna merah. Sementara itu, Gambar 2.4 menunjukkan kombinasi warna RGB dan pemetaan warna dalam ruang berdimensi tiga. Dan gambar 2.5 menunjukkan keadaan suatu citra dan representasi warnanya.



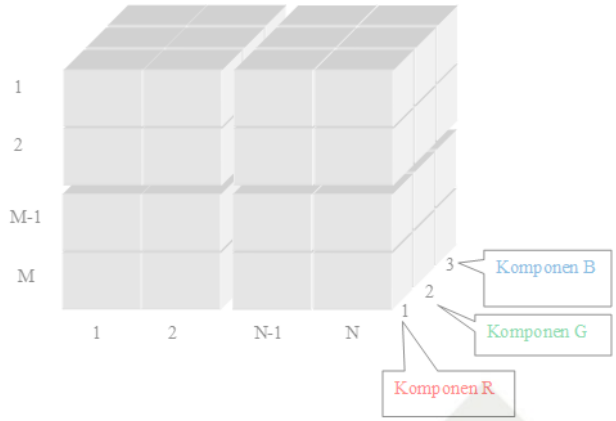
Gambar 2.4 Kombinasi warna RGB dan pemetaan RGB dalam ruang dimensi 3 [3].

Gambar 2.5 Citra berwarna dan representasi warnanya setiap *pixel* dinyatakan dengan nilai R, G, dan B[3].

Melalui matlab, citra bewarna dapat dibaca dengan **imread**. Contohnya :

```
>> gambar = imread('citra.jpg');
>> size(gambar)

ans =

   960   960     3
```

Perhatikan, ketika dicoba mengenakan **size** pada **gambar**, hasilnnya menunjukkan **gambar** merupakan larik berdimensi tiga, dengan dimensi ketika berisi tiga buah nilai. Hal inilah yang membedakan dengan citra berskala keabuan. Secara umum, larik hasil pembacaan citra berwarna dapat digambarkan seperti berikut.



Gambar 2.6 Hasil pembacaan citra berwarna

Dimensi ketiga menyatakan komponen R, G, B. Indeks pertama menyatakan komponen R, indeks kedua menyatakan komponen G, dan indeks ketiga menyatakan komponen B. Berikut adalah cara untuk mendapatkan komponen R, G dan B pada larik **gambar** diatas :

```
> R = gambar(:,:,1);
> G = gambar(:,:,2);
> B = gambar(:,:,3);
```

2.7 *Discrete Wavelet Transform* (DWT)

Wavelet adalah suatu konsep yang relative baru dikembangkan. Kata *wavelet* sendiri diberi Jean Morlet dan Alex Grossmann diawal tahun 1980-an, dan berasal dari bahasa prancis, *ondelette* yang berarti gelombang kecil. Kata *onde* yang berarti gelombang kemudian diterjemahkan ke bahasa inggris menjadi *wave*, lalu digabungkan dengan kata aslinya sehingga terbentuk kata baru *wavelet*[15]. Dalam penelitian lainnya dikemukakan pendapat tentang wavelet merupakan alat analisis yang biasa digunakan untuk menyajikan data atau fungsi operator kedalam komponen-komponen frekuensi yang berlainan, dan kemudian mengkaji setiap komponen dengan suatu resolusi yang sesuai dengan skalanya [16].

Wavelet adalah sinyal yang lokal dalam waktu dan skala pada umumnya memiliki bentuk yang tidak teratur. Sebuah *wavelet* adalah gelombang durasi efektif terbatas yang memiliki nilai rata-rata nol. Istilah ”*wavelet*” berasal dari fakta bahwa *wavelet* mengintegrasikan ke nol, *wavelet* gelombang atas dan kebawah pada sumbu. Banyak *wavelet* juga menampilkan properti ideal untuk representasi sinyal kompak : *orthogonality*. Properti ini memastikan bahwa data tidak lebih terwakili, sebuah sinyal dapat didekomposisikan menjadi banyak bergeser pada

skala representasi dari ibu *wavelet* asli. Sebuah *wavelet* transformasi dapat digunakan untuk dekomposisi sinyal ke *wavelet* komponen. Setelah ini dilakukan koefesien *wavelet* dapat hancur untuk menghapus beberapa detail. *Wavelet* memiliki keuntungan besar untuk dapat memisahkan rincian halus dalam sinyal. *Wavelet* sangat kecil dapat digunakan untuk mengisolasi rincian yang akan sangat baik disinyal, sementara *wavelet* yang sangat besar dapat mengidentifikasi rincian kasar. Transformasi *wavelet* dibagi menjadi dua bagian besar, yaitu Transformasi *wavelet* kontinu (*Continous Wavelet Transform/CWT)* dan Transformasi *Wavelet* Diskrit (*Discrete Wavelet Transform/DWT)* diturunkan dari *mother wavelet* melalui transisi/pergeseran dan penskalaan/kompresi. *Mother wavelet* digunakan dalam transformasi *wavelet*, karena *mother wavelet* menghasilkan semua fungsi *wavelet* yang digunakan dalam transformasi melalui transisi dan penskalaan, maka *mother wavelet* juga akan menunjukan karakteristik dari transformasi *wavelet* yang dihasilkan[17].

Transformasi wavelet mempunyai penerapan yang luas pada aplikasi pengolahan citra. Ada berbagai jenis transformasi *wavelet*, diantaranya adalah *Discrete Wavelet Transform* (DWT) 1-dimensi (1-D) dan (DWT) 2-dimensi (2-D). Proses transformasi *wavelet* secara konsep cukup sederhana. Citra semula yang ditransformasi dibagi (didekomposisi) menjadi 4 *sub image* baru untuk menggantikannya. Setiap *sub image* berukuran ¼ kali citra asli. 3 *sub image* pada posisi atas kanan, bawah kiri, dan bawah kanan akan tampak seperti versi kasar dari citra asli karena berisi komponen dari frekuensi tinggi pada citra asli. Sedangkan untuk 1 *sub image* atas kiri tampak seperti citra asli dan tampak lebih halus, karena berisi komponen frekuensi rendah dari citra asli. *Sub image* tersebut dibagi seperti semula lagi menjadi 4 *sub image* baru. Proses demikian dapat diulangi seterusnya. Sesuai dengan tingkatan transformasi yang diinginkan. Pada gambar dibawah ini terdapat contoh transformasi wavelet pada citra

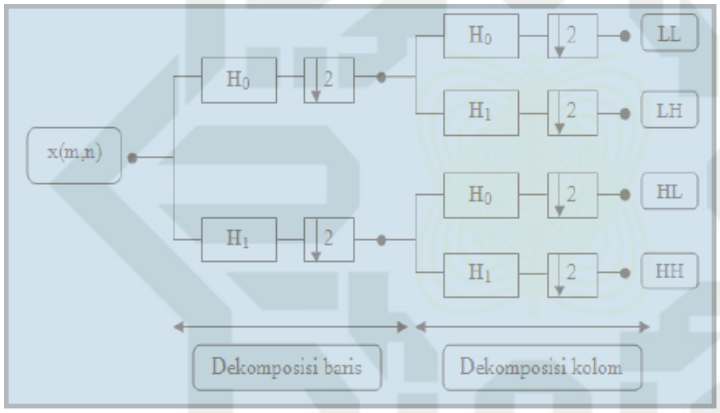
Gambar 2.7 Transformasi Wavelet pada Citra [18].

Gambar diatas hanya dilakukan 1 level dekomposisi, dimana citra asli dibagi menjadi 4 *sub image* baru, *image* yang berada pada bagian kiri atas terlihat lebih halus karena berisi komponen frekuensi rendah, sedangkan 3 *image* lainnya tampak lebih buram karena berisi komponen frekuensi tinggi. Masing-masing *sub image* tersebut mengandung ¼ dari nilai *image* asli[18].

Untuk citra 2 dimensi dekomposisi perataan dan pengurangan sama dengan citra 1 dimensi, hanya saja proses dekomposisi dilakukan dalam 2 tahap, yaitu :

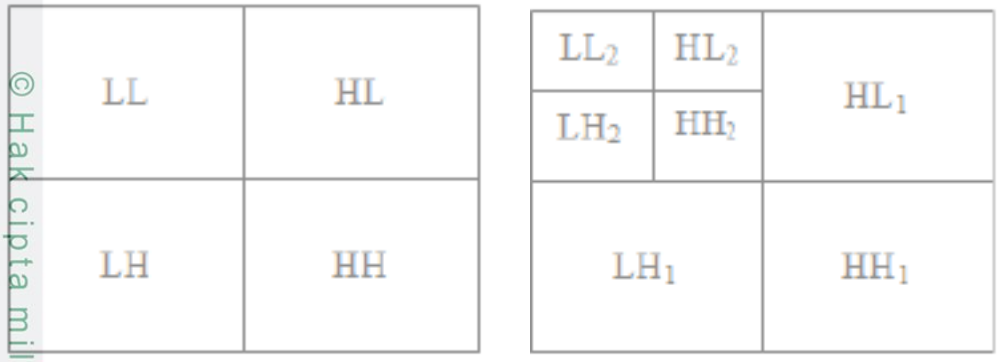
1. Tahap pertama proses dekomposisi dilakukan pada seluruh baris,
2. Tahap kedua pada citra hasil tahap pertama dilakukan proses dekomposisi dalam arah kolom

Di dalam proses dekomposisinya transformasi wavelet diskrit dua dimensi dilakukan dengan memproses baris dan kolom secara terpisah, yang dapat diilustrasikan dengan gambar berikut ini.



Gambar 2.8 *Two Dimensial DWT*[3].

Pada gambar 2.8 LL menyatakan bagian koefisien yang diperoleh melalui proses tapis *Low pass* dilanjutkan dengan *Low pass*, citra pada bagian ini mirip dan merupakan versi lebih halus dari citra aslinya sehingga koefisien pada bagian ini sering disebut dengan komponen aproksimasi. LH menyatakan bagian koefisien yang diperoleh melalui proses tapis *Low pass* kemudian dilanjutkan dengan *High pass*, koefisien pada bagian ini menunjukan citra tepi dalam arah horizontal. HL menyatakan bagian yang diperoleh melalui proses *High pass* kemudian dilanjutkan dengan *Low pass*, koefisien pada bagian ini menunjukkan citra tepi dalam arah vertical, dan HH menyatakan proses yang diawali dengan *High pass*, koefisien menunjukan citra tepi dalam arah diagonal. Ketiga komponen LH, HL, dan HH disebut komponen detil.



Gambar 2.9 Dekomposisi transformasi wavelet level 1 dan level 2[3].

2.8 Mode Penyisipan Substitusi

Mode penyisipan subtitusi merupakan metode penyisipan yang mudah dan cepat. Prinsip kerjanya ialah dengan merubah nilai value pada subband DWT berdasarkan bit sisip yang akan disisipkan[24].

Tabel 2.4 Prinsip kerja Penyisipan Subtitusi

Bit Sisip Pesan	Nilai *Subband* DWT	Penyisipan
0	Nilai *subband* < 0	Nilai *subband* tetap
	Nilai *subband* > 0	Nilai *subband* × (-1)
	Nilai *subband* = 0	Nilai *subband* – 1
1	Nilai *subband* < 0	Nilai *subband* × (-1)
	Nilai *subband* > 0	Nilai *subband* tetap
	Nilai *subband* = 0	Nilai *subband* + 1

Berdasarkan Tabel 2.4, untuk bit sisip 0 maka nilai value pada subband DWT dibuat negatif dan untuk bit sisip 1 maka nilai value pada subband DWT dibuat positif untuk mendapatkan bit stego. Sedangkan pada proses ekstraksi, jika nilai value subband DWT bernilai positif, maka nilai bit dekripsinya bernilai 1. Sebaliknya, jika nilai value subband DWT bernilai negatif, maka nilai bit dekripsinya bernilai 0.

2.9 Steganografi

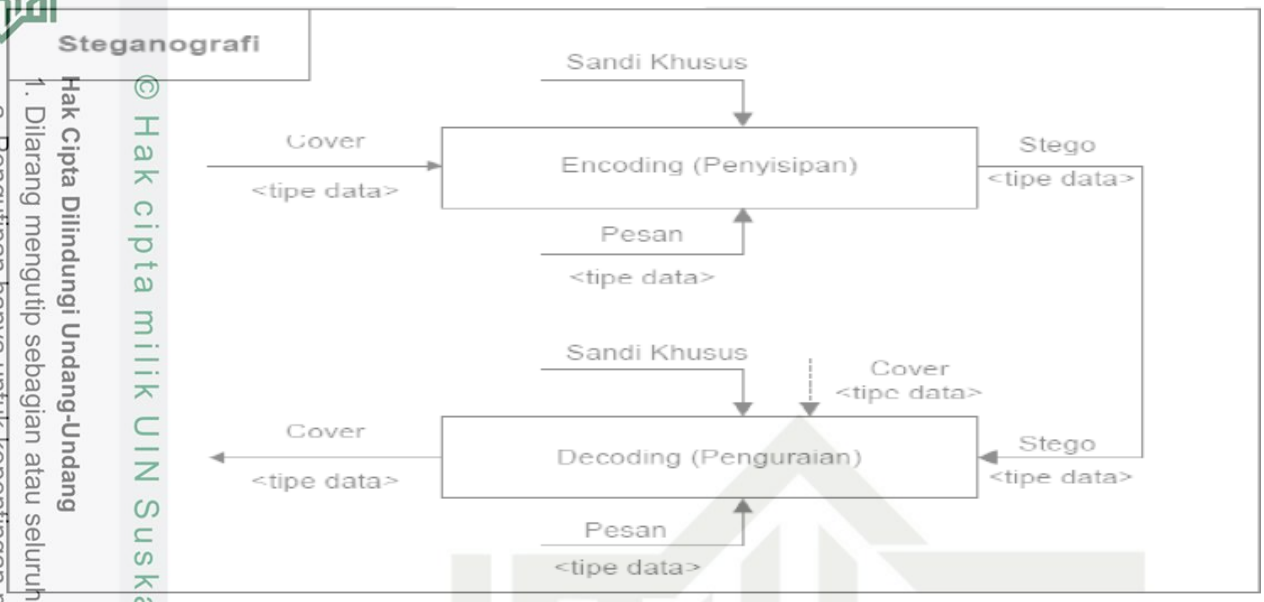
Kata steganografi pada awalnya berasal dari kata *steganos*, *steganos* sendiri sebenarnya merupakan kata dari bahasa Yunani. Lebih lengkapnya *steganos* memiliki arti penyamaran atau penyembunyian dan *graphein* atau *graptos* memiliki arti tulisan. Pengertian steganografi yang cukup sering digunakan dalam pembelajaran metodologi sejarah adalah "menulis tulisan yang tersembunyi atau terselubung". Jadi steganografi adalah teknik menyembunyikan atau menyamarkan keberadaan pesan rahasia dalam suatu media penampungnya sehingga orang lain tidak menyadari adanya pesan didalam media tersebut.

Steganografi sudah digunakan sejak 2500 tahun yang lalu, untuk kepentingan politik, militer, diplomatik, serta untuk kepentingan pribadi. Teknik steganografi konvesional berusaha merahasiakan komunikasi dengan cara menyembunyikan pesan atau mengkamuflasekan pesan. Sesungguhnya prinsip dasar dalam steganografi lebih dikonsentrasikan pada kerahasian komunikasi nya bukan pada datanya[10].

Kini istilah steganografi termasuk penyembunyian data digital dalam berkas-berkas (*file)* komputer. Contohnya, si pengirim mulai dengan berkas gambar biasa, lalu mengatur warna setiap pixel ke-100 untuk menyesuaikan suatu huruf dalam alphabet (perubahannya begitu halus sehingga tidak ada seorangpun yang menyadarinya jika ia tidak benar-benar memperhatikannya).

Pada umumnya, pesan steganografi muncul dengan rupa lain seperti gambar, artikel, daftar belanja, atau pesan-pesan lainnya. Pesan yang tertulis ini merupakan tulisan yang menyelubungi atau menutupi. Contohnya, suatu pesan bisa disembunyikan dengan menggunakan tinta yang tidak terlihat diantara garis-garis yang kelihatan.

Teknik steganografi meliputi banyak sekali metode komunikasi untuk menyembunyikan pesan rahasia (teks atau gambar) didalam berkas-berkas lain yang mengandung teks, *image*, bahkan *audio* tanpa menunjukkan ciri-ciri perubahan yang nyata atau terlihat dalam kualitas dan struktur dari berkas semula. Metode ini termasuk tinta yang tidak tampak, *microdots*, pengaturan kata, tanda tangan digital, jalur tersembunyi, dan komunikasi spektrum lebar. Ringkasnya, steganografi adalah teknik menanamkan *embedded message* pada suatu *cover object*, dimana hasilnya berupa *stego object.* Adapun proses steganografi selengkapnya ditunjukkan pada Gambar 2.10.



Gambar 2.10 Proses Steganografi[10].

Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Dalam prakteknya, kebanyakan pesan disembunyikan dengan membuat perubahan tipis terhadap data digital lain yang isinya tidak akan menarik perhatian dari penyerang potensial, sebagai contoh sebuah gambar yang terlihat tidak berbahaya. Perubahan ini bergantung pada kunci (sama pada kriptografi) dan pesan untuk disembunyikan. Orang yang menerima gambar kemudian dapat menyimpulkan informasi tersembunyi dengan cara mengganti kunci yang benar ke dalam algoritma yang digunkanan. Pada metode steganografi cara ini sangat berguna jika digunakan pada cara steganografi komputer karena banyak format berkas digital yang dapat dijadikan media untuk menyembunyikan pesan. Format yang biasa digunakan di antaranya :

1. Format *image* : bitmap (bmp), gif, pcx, jpeg, dan lain lain.
2. Format *audio* : wav, voc, mp3, dan lain lain.
3. Format lain : teks *file*, html, pdf, dan lain lain.

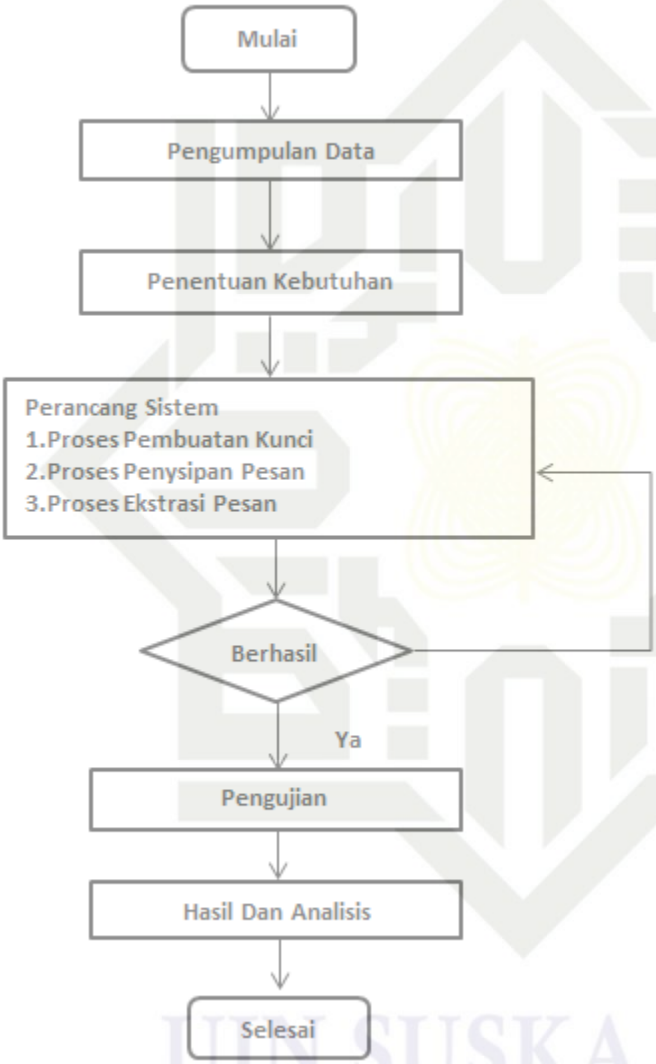
Kelebihan steganografi jika dibandingkan dengan kriptografi adalah pesan-pesannya tidak menarik perhatian orang lain. Pesan-pesan berkode dalam kriptografi yang tidak disembunyikan, walaupun tidak dapat dipecahkan, akan menimbulkan kecurigaan. Seringkali, steganografi dan kriptografi digunakan secara bersamaan untuk menjamin keamanan pesan rahasianya. Sebuah pesan steganografi (plaintext), biasanya pertama-

tama dienkripsikan traditional, yang menghasilkan *chipertext* kemudian *covertext* dimodifikasi dalam beberapa cara sehingga berisi *chipertext*, yang menghasilkan *stegotext*. Contohnya, ukuran huruf, ukuran spasi, jenis huruf, atau karakteristik *cover text* lain nya dapat dimanipulasi untuk membawa pesan tersembunyi; hanya penerima (yang harus mengetahui teknik yang digunakan) dapat membukan pesan dan mendekripsikannya[11].

BAB III
METODOLOGI PENELITIAN

3.1 Alur Tahapan Penelitian

Pada bab ketiga ini akan dibahas tahapan dan langkah-langkah penulis dalam melakukan penelitian mulai dari awal sampai akhir penelitian. Alur tahapan penelitian dapat dilihat dari gambar 3.1 berikut



Gambar 3.1 Alur Tahapan Penelitian

3.2 Pengumpulan Data

3.2.1 Studi Literatur

Pada tahap ini penulis melakukan pengumpulan data dan informasi melalui literatur buku, internet dan karya ilmiah lainnya seperti jurnal ilmiah, tesis dan disertasi. Tujuan digunakannya studi literatur ini yaitu untuk mencari data-data mengenai teknik steganografi *discrete wavelet transform* dua dimensi dan algoritma kriptografi RSA pada perancangan dan analisis keamanan pesan.

3.3 Penentuan Kebutuhan Sistem

Dalam perancangan sistem pengamanan pesan teks pada citra digital berdasarkan steganografi menggunakan DWT, dibutuhkan beberapa spesifikasi dari perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan dalam penelitian tugas akhir ini.

3.3.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk mengimplementasikan sistem steganografi yang telah dirancang adalah sebagai berikut :

1. *System Model* : Laptop Dell Latitude E5420
2. *Processor* : Intel®Core ™ i7-2630Qm @ 2.00GHz (8CPUs),~2.Ghz
3. *Memory* : 2048 MB RAM
4. *Harddisk* : 500 GB HDD

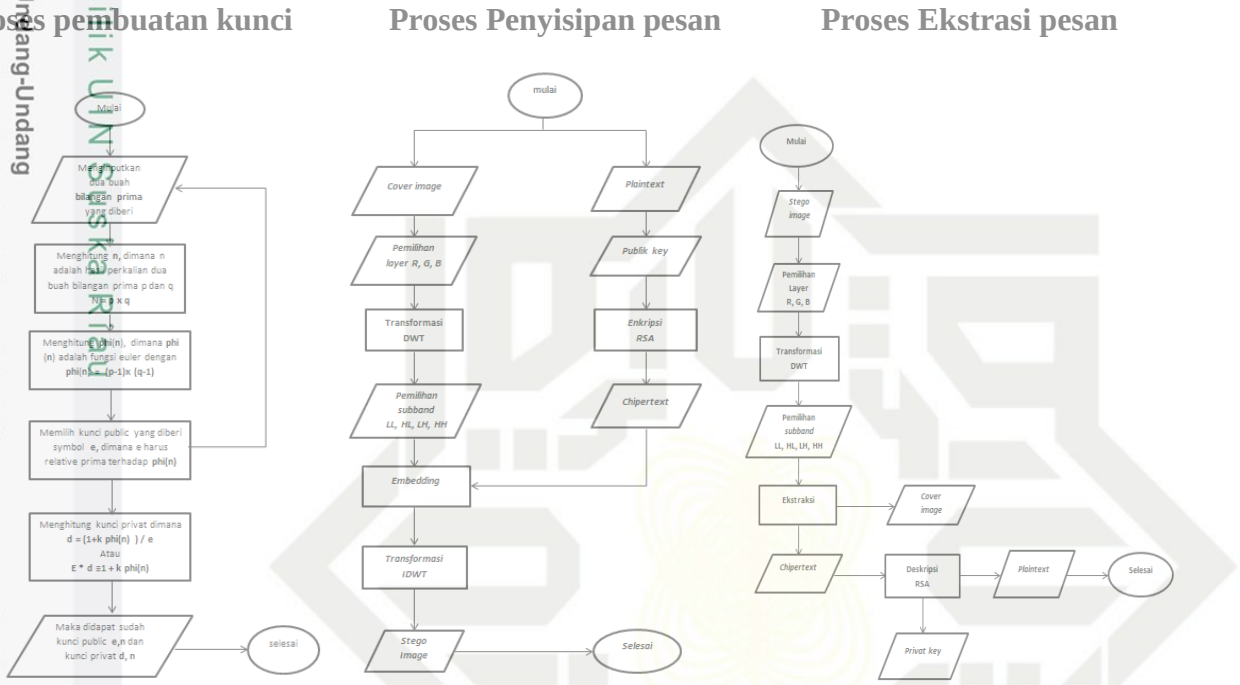
3.3.2 Spesifikasi Perangkat Lunak

Spesifikasi Perangkat lunak yang digunakan untuk mengimplementasikan sistem steganografi yang telah dirancang adalah sebagai berikut:

1. *Operating system* : Windows 7 Ultimate 64-Bit (6.1, Build 7600)
2. *Programing tool* : MATLAB (R2020a)
3. Paint untuk membuat ukuran gambar sesuai kebutuhan
4. Microsoft office excel 2010 untuk mengolah data hasil pengujian sistem.

3.4 Perancangan dan Pemodelan Sistem

Perancangan sistem dijelaskan dengan diagram alir dibawah ini. Secara garis besar sistem terbagi menjadi tiga buah proses. Pada Gambar 3.1, ditunjukan proses pembuatan kunci kriptografi algoritma kriptografi RSA, proses penyisipan pesan (*embbeding*) dan proses ekstraksi pesan (*retrieval*) pada sistem.



Gambar 3.1 Perancangan dan Pemodelan sistem

Pada Gambar 3.1, input yang digunakan dalam sistem adalah pesan berupa teks dan cover citra berupa citra warna (RGB). Berikut ini dijelaskan algoritma dalam perancangan sistem.

3.4.1 Proses Enkripsi RSA

Pada proses ini, pesan berupa teks dienkripsi menggunakan kriptografi RSA. Prinsip kerja algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Selama bilangan tersebut tidak dapat difaktorkan, selama itu pula keamanan algoritma RSA terjamin. Langkah pertama dalam algoritma RSA ini adalah membangkitkan pasangan kunci. Sebagaimana yang telah diketahui, kunci publik bersifat tidak rahasia atau diketahui oleh umum, sedangkan kunci rahasia bersifat rahasia hanya diketahui oleh pihak yang ingin mendekripsi.

Berikut adalah langkah – langkah dalam membangkitkan pasangan kunci :

1. Pertama pilih dua buah bilangan prima p dan q secara acak.
2. Hitung nilai n = p*q
3. Hitung ϕ (n) = (p - 1) (q - 1).
4. Pilih kunci publik e yang memiliki syarat bahwa e harus relatif prima terhadap ϕ (n).
5. Membangkitkan kunci privat dengan persamaan :

$$d = \frac{1+k\ \phi\ (n)}{\mathrm{e}}$$

Hasil dari algoritma diatas adalah pasangan kunci publik (e, n) dan pasangan kunci rahasia (d, n) [10]. Setelah didapat *public key* dan *private key*, proses selanjutnya adalah proses enkripsi.

Langkah-langkah dari proses enkripsi adalah sebagai berikut :

1. Langkah pertama adalah mengambil nilai e dan n dari proses pembangkitan kunci.
2. Masukan teks yang akan dienkripsi.
3. Ubah teks yang akan dienkripsi kedalam bentuk *decimal* sesuai tabel ASCII.
4. Setelah itu enkripsi menggunakan pasangan kunci publik.

3.2.2 Proses Penyisipan

Pada proses ini dilakukan penyisipan menggunakan *Discrete Wavelet Transform* (DWT). DWT mendekomposisi satu level daerah warna yang sudah dipilih. Keluaran dari DWT ini adalah *subband-subband* berupa HH, HL, LH, dan LL [11].

Proses penyisipan ini dijelaskan sebagai berikut:

1. Cover citra yang berupa citra RGB terlebih dahulu dipisahkan daerahnya berdasarkan daerah warna yang dipilih, kemudian diambil layer yang diinginkan sebagai media penyisipan. Dipilih layer *red* dikarenakan paling bagus dilihat dari nilai PSNR.
2. Pesan rahasia yang acak dilakukan proses penyisipan ke dalam citra. Kemudian dilakukan proses DWT untuk mendapatkan tempat penyisipan.
3. Sub-band yang di pakai adalah LH, dikarenakan pada sub-band LH paling tahan terhadap serangan.
4. Kemudian pada proses penyisipan kita mendapatkan nilai value pada LH. Untuk bit sisip 0 maka nilai value pada LH dibuat negatif dan untuk bit sisip 1 maka nilai value pada LH dibuat positif untuk mendapatkan bit stego, dijelaskan pada subbab 2.5.
5. Setelah proses penyisipan selesai, layer tersebut di Invers DWT dan dilakukan proses pembulatan terhadap nilai pikselnya.

6. Untuk tahap akhirnya layer tersebut digabungkan kembali dengan dua layer lainnya sehingga kembali terbentuk citra RGB dan disebut sebagai citra cover.
7. Proses penyisipan yang dilakukan bersifat blind, sehingga pada saat proses ekstraksi tidak dibutuhkan citra aslinya sebagai pembanding terhadap citra stegonya.

3.4.3 Proses Ekstraksi

Pada proses ini, pesan akan diambil kembali dari citra stego. Proses ekstraksi pesan merupakan proses kebalikan dari proses penyisipan. Proses ekstraksi dijelaskan sebagai berikut

1. Pertama, baca stego *image*.
2. Setelah itu kita mengambil salah satu layer dari citra stego di mana tempat pesan itu disisipkan, setelah itu layer tersebut di *Discrete Wavelet Transform* (DWT).
3. Kemudian dilakukan ekstraksi dengan cara pengambilan bit dari layer tersebut yang merupakan *chipertext* yang dikirimkan, dijelaskan pada subbab 2.5.
4. Bit tersebut diproses untuk diubah kembali menjadi karakter yang isinya adalah pesan rahasia.

3.4.4 Proses Dekripsi

Pada proses ini, *cipherteks* yang didapat akan didekripsi agar dapat diketahui isi pesan rahasianya. Proses dekripsi dijelaskan sebagai berikut :

1. Pertama, baca chipertext.
2. Kemudian masukkan kunci privat yang telah didapat.
3. Lakukan proses dekripsi, dijelaskan pada subbab. 2.6.2.
4. Pesan (plaintext) dihasilkan.

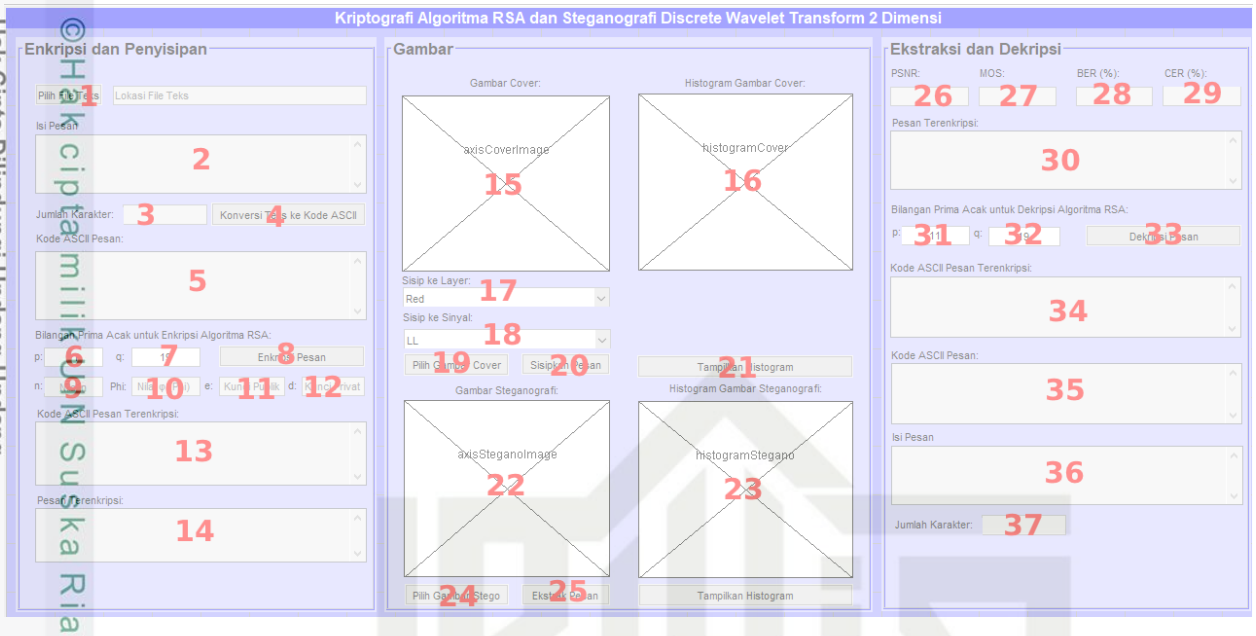
3.4.5 Perancangan *Graphical User Interface (GUI)*

Setelah program berhasil dijalankan, langkah selanjutnya adalah perancangan GUI. GUI dibangun dengan objek grafik, tombol, panel, text dan menu agar mudah digunakan. Kemampuan grafis yang baik pada GUI akan memudahkan dalam proses enkripsi, penyisipan, ekstraksi, dan dekripsi serta menghitung parameter PSNR (*Peak Signal to Noise Ratio*), MOS (*Mean Opinion Score*), BER (*Bit Error Rate*) dan CER (*Character Error Rate*).

Berikut ini adalah *flowchart* dari perancangan GUI

Gambar 3.2 *Flowchart* Perancangan *GUI*

Berikut adalah tampilan dari perancangan GUI



Gambar 3.3 Rancangan GUI

Keterangan gambar diatas adalah sebagai berikut :

1. Tombol yang berfungsi untuk menginput *file* teks yang akan enkripsi.
2. *Form* yang berfungsi untuk menampilkan isi pesan dari *file* teks.
3. *Form* yang berfungsi untuk menampilkan jumlah karakter pesan.
4. Tombol yang berfungsi untuk mengubah pesan teks ke kode ASCII.
5. *Form* yang berfungsi untuk menampilkan kode ASCII pesan.
6. *Form* yang berfungsi untuk menginputkan variabel p.
7. *Form* yang berfungsi untuk menginputkan variabel q.
8. Tombol yang berfungsi untuk mengenkripsi pesan.
9. *Form* yang berfungsi untuk menampilkan variabel n.
10. *Form* yang berfungsi untuk menampilkan variabel Phi.
11. *Form* yang berfungsi untuk menampilkan variabel e (Kunci Publik).
12. *Form* yang berfungsi untuk menampilkan variabel d (Kunci Privat).
13. *Form* yang berfungsi untuk menampilkan kode ASCII pesan terenkripsi.
14. *Form* yang berfungsi untuk menampilkan pesan terenkripsi.
15. *Axis* yang berfungsi untuk menampilkan gambar cover.
16. *Axis* yang berfungsi untuk menampilkan histogram gambar cover.

17. *Form* yang berfungsi untuk menginputkan layer penyisipan.
18. *Form* yang berfungsi untuk menginputkan sinyal penyisipan.
19. Tombol yang berfungsi untuk menginputkan gambar cover.
20. Tombol yang berfungsi untuk menyisipkan pesan.
21. Tombol yang berfungsi untuk menampilkan histogram gambar cover.
22. *Axis* yang berfungsi untuk menampilkan gambar stego.
23. *Axis* yang berfungsi untuk menampilkan histogram gambar stego.
24. Tombol yang berfungsi untuk menginputkan gambar stego.
25. Tombol yang berfungsi untuk mengekstrak pesan.
26. *Form* yang berfungsi untuk menampilkan PSNR.
27. *Form* yang berfungsi untuk menampilkan MOS.
28. *Form* yang berfungsi untuk menampilkan BER.
29. *Form* yang berfungsi untuk menampilkan CER.
30. *Form* yang berfungsi untuk menampilkan pesan terenkripsi
31. *Form* yang berfungsi untuk menginputkan variabel p.
32. *Form* yang berfungsi untuk menginputkan variabel q.
33. Tombol yang berfungsi untuk dekripsi pesan.
34. *Form* yang berfungsi untuk menampilkan kode ASCII pesan terenkripsi.
35. *Form* yang berfungsi untuk menampilkan kode ASCII pesan asli.
36. *Form* yang berfungsi untuk menampilkan pesan asli.
37. *Form* yang berfungsi untuk menampilkan jumlah karakter pesan asli.

3.5 *Source Code* Inti Aplikasi

Dalam ilmu komputer, *source code* atau kode program adalah suatu rangkaian pernyataan atau deklarasi yang ditulis dalam bahasa pemrograman komputer yang dapat dibaca oleh manusia. Berikut adalah *source code* inti dari aplikasi :

3.5.1 *Pre-processing*

a. Membaca *File* Teks

```
[fileName, pathName]=uigetfile('*.txt','Select Text File');
fullTxtPathName=strcat(pathName,fileName);
pesan=fileread(fullTxtPathName);
```

Source code di atas digunakan untuk membaca data *file* teks kemudian menyimpan nama *file* teks pada variable fileName dan lokasi *file* teks pada variabel pathName.

b. Konversi Teks ke Kode ASCII

```
pesan=get(handles.teksPesanForm,'String');
asciiPesan=double(char(pesan));
```

Source code di atas digunakan untuk mengkonversi teks pesan ke kode ASCII kemudian disimpan pada variabel asciiPesan.

3.5.2 Enkripsi Pesan

```
pesan=get(handles.teksPesanForm,'String');
panjangPesan=length(pesan);
p=get(handles.pEnkripsiForm,'String');
q=get(handles.qEnkripsiForm,'String');
[n,Phi,e,d]=inisialisasi(str2double(p),str2double(q));
asciiPesan=double(char(pesan));
for j=1:panjangPesan
    cipher(j)=crypt(asciiPesan(j),n,e);
end
```

Source code di atas digunakan untuk mengenkripsi ASCII pesan Fungsi inisialisasi berfungsi untuk mencari nilai n, Phi, e, dan d untuk enkripsi RSA dari variabel p dan q yang sudah dimasukkan sebelumnya. Fungsi crypt berfungsi untuk enkripsi ASCII pesan menjadi ASCII pesan terenkripsi menggunakan variabel n dan e, kemudian hasilnya disimpan ke variabel cipher.

3.5.3 Pilih Gambar *Cover*

```
[fileName,pathName]=uigetfile('*.bmp;*.jpg;*.tiff','Select Cover Image
File');
fullImgPathName=strcat(pathName,fileName);
gambarCover=imread(fullImgPathName);
imwrite(gambarCover, 'cover_image.bmp');
imshow(gambarCover);
```

Source code di atas digunakan untuk membaca data file gambar cover kemudian menyimpan nama *file* gambar cover pada variable fileName dan lokasi *file* gambar cover pada variabel pathName.

3.5.4 Penyisipan Pesan

a. Mengambil Teks Enkripsi RSA

```
fid=fopen('teks_enkripsi.txt','r');
pesanEnkripsi=fread(fid,[1,inf],'char');
panjangPesan=length(pesanEnkripsi);
intPesanEnkripsi=uint16(pesanEnkripsi);
fclose(fid);
```

Source code di atas digunakan untuk membuka file teks enkripsi kemudian mengubah pesan terenkripsi menjadi kode ASCII dan disimpan di variabel intPesanEnkripsi.

b. Konversi Teks Enkripsi RSA ke Biner

```
panjangBitPesan=16*panjangPesan;
bitPesan=dec2bin(panjangBitPesan,16);
for i=1:panjangPesan
    pesanBiner=dec2bin(intPesanEnkripsi(i),16);
    bitPesan=[bitPesan pesanBiner];
end
```

Source code di atas digunakan untuk membaca membuat bit pesan dari kode ASCII pesan terenkripsi dan disimpan kedalam variabel bitPesan. 16 bit pesan pertama adalah panjang pesan dalam bit, kemudian setiap 1 kode ASCII pesan terenkripsi akan dikonversi menjadi 16 bit untuk disisipkan.

c. Proses Memisahkan *Layer* RGB dan DWT (*Discrete Wavelet Transform*)

```
fullImgPathName=imread('cover_image.bmp');
redLayer=fullImgPathName(:,:,1);
greenLayer=fullImgPathName(:,:,2);
blueLayer=fullImgPathName(:,:,3);
if(strcmp(selectedLayer,'red')), usedLayer=redLayer;
elseif(strcmp(selectedLayer,'green')),usedLayer=greenLayer;
else, usedLayer=blueLayer;
end
[LL,LH,HL,HH]=dwt2(usedLayer,'haar');
```

Source code di atas digunakan untuk membaca data *file* gambar *cover* kemudian memisahkan *layer* gambar menjadi *red*, *green*, dan *blue*. Setelah *layer* dipisah kemudian *layer*

yang dipilih akan dilakukan proses DWT 2 Dimensi sehingga *layer* dipisah menjadi sinyal LL (Low-Low), LH (Low-High), HL (High-Low) dan HH (High-High).

d. Proses Penyisipan Bit Pesan

```
for i=1:panjangBitPesan+16
    if(bitPesan(i)=='0')
        if(SignalSebaris(i)>0.0001)
            SignalSebaris(i)=-SignalSebaris(i);
        elseif(SignalSebaris(i)<-0.0001)
            SignalSebaris(i)=SignalSebaris(i);
        else
            SignalSebaris(i)=-1.5;
        end
    elseif(bitPesan(i)=='1')
        if(SignalSebaris(i)<-0.0001)
            SignalSebaris(i)=abs(SignalSebaris(i));
        elseif(SignalSebaris(i)>0.0001)
            SignalSebaris(i)=SignalSebaris(i);
        else
            SignalSebaris(i)=1.5;
        end
    end
end
```

Source code di atas digunakan untuk menyisipkan bit pesan dengan cara *looping* ke dalam sinyal yang dipilih. Untuk bit pesan 0, jika sinyal bernilai positif atau mendekati 0 maka nilai sinyal diubah menjadi negatif. Untuk bit pesan 1, jika sinyal bernilai negatif atau mendekati 0 maka nilai sinyal diubah menjadi positif dan disimpan ke dalam variabel Signal Sebaris.

e. Proses *Inverse* DWT dan Penggabungan *Layer* RGB

```
ukuranCover=size(fullImgPathName);
if(strcmp(selectedSignal,'LL'))
    IDWT=idwt2(EMBED,LH,HL,HH,'haar',ukuranCover);
elseif(strcmp(selectedSignal,'LH'))
    IDWT=idwt2(LL,EMBED,HL,HH,'haar',ukuranCover);
elseif(strcmp(selectedSignal,'HL'))
    IDWT=idwt2(LL,LH,EMBED,HH,'haar',ukuranCover);
```

```
else
    IDWT=idwt2(LL,LH,HL,EMBED,'haar',ukuranCover);
end

if(strcmp(selectedLayer,'red'))
    stegoImage=cat(3,IDWT,greenLayer,blueLayer);
elseif(strcmp(selectedLayer,'green'))
    stegoImage=cat(3,redLayer,IDWT,blueLayer);
else
    stegoImage=cat(3,redLayer,greenLayer,IDWT);
end
imwrite(uint8(stegoImage),'stego_image.bmp');
axes(handles.axisSteganoImage);
imshow(stegoImage);
```

Source code di atas digunakan untuk menggabungkan kembali sinyal yang telah disisipkan bit pesan terenkripsi dengan 3 sinyal lainnya menggunakan fungsi idwt 2 (*Inverse* DWT 2-Dimensi) sehingga menjadi layer R, G, atau B (tergantung layer yang dipilih) dan disimpan ke variabel IDWT. Setelah proses idwt 2 varibel IDWT digabungkan dengan 2 layer warna lainnya dan disimpan ke variabel stegoImage. variabel stegoImage disimpan menjadi gambar bitmap dengan nama stego_image, kemudian ditampilkan ke pengguna menggunakan fungsi imshow.

f. Proses Menghitung Nilai PSNR dan MOS

```
nilaiPsnr=psnr(stegoImage,fullImgPathName);
if(nilaiPsnr>37), nilaiMos='Excellent';
elseif(nilaiPsnr>=31), nilaiMos='Good';
elseif(nilaiPsnr>=25), nilaiMos='Fair';
elseif(nilaiPsnr>20), nilaiMos='Poor';
else, nilaiMos='Bad'; end
```

Source code di atas digunakan mencari nilai PSNR antara gambar *cover* dan gambar stego menggunakan fungsi psnr dan hasilnya disimpan ke dalam variabel nilaiPsnr. Nilai MOS didapatkan dari rentang nilai PSNR dan disimpan ke dalam variabel nilaiMos.

3.5.5 Pilih Gambar Stego

```
[fileName,pathName]=uigetfile('*.bmp;*.jpg;*.tiff','Select Stego Image
File');
fullImgPathName=strcat(pathName,fileName);
gambarStego=imread(fullImgPathName);
imwrite(gambarStego, 'stego_image.bmp');
imshow(gambarStego);
```

Source code di atas digunakan untuk membaca data *file* gambar stego kemudian menyimpan nama *file* gambar stego pada variable fileName dan lokasi *file* gambar cover pada variabel pathName.

3.5.6 Ekstrak Pesan

a. Proses Memisahkan *Layer* RGB dan DWT (*Discrete Wavelet Transform*)

```
fullImgPathName=imread('stego_image.bmp');
redLayer=fullImgPathName(:,:,1);
greenLayer=fullImgPathName(:,:,2);
blueLayer=fullImgPathName(:,:,3);
if(strcmp(selectedLayer,'red')), usedLayer=redLayer;
elseif(strcmp(selectedLayer,'green')),usedLayer=greenLayer;
else, usedLayer=blueLayer;
end
[LL,LH,HL,HH]=dwt2(usedLayer,'haar');
```

Source code di atas digunakan untuk membaca data *file* gambar stego kemudian memisahkan *layer* gambar menjadi *red*, *green*, dan *blue*. Setelah *layer* dipisah kemudian *layer* yang dipilih akan dilakukan proses DWT 2 Dimensi sehingga *layer* dipisah menjadi sinyal LL (Low-Low), LH (Low-High), HL (High-Low) dan HH (High-High).

b. Proses Ekstrak Bit Pesan

```
panjangBitPesanBit=[];
for i=1:16
    if(SignalSebaris(i)<0)
        panjangBitPesanBit=[panjangBitPesanBit '0'];

    elseif(SignalSebaris(i)>0)
        panjangBitPesanBit=[panjangBitPesanBit '1'];
    else
```

```
        panjangBitPesanBit=[panjangBitPesanBit '0'];
    end
end
panjangBitPesan=bin2dec(panjangBitPesanBit(1:16));
bitPesan=[];
for i=1:panjangBitPesan
    if(SignalSebaris(i+16)<-0.0001)
        bitPesan=[bitPesan '0'];
    elseif(SignalSebaris(i+16)>0.0001)
        bitPesan=[bitPesan '1'];
    else
        bitPesan=[bitPesan '0'];
    end
end
```

Source code di atas digunakan untuk mengekstrak panjang bit pesan dengan cara *looping* ke dalam sinyal yang dipilih. Jika sinyal bernilai negatif maka bit untuk panjang bit pesan adalah 0. Jika sinyal bernilai positif maka bit untuk panjang bit pesan adalah 1 dan disimpan ke dalam variabel panjangBitPesan. Setelah itu dilakukan pembacaan sinyal yang dipilih berdasarkan panjang bit pesan. Jika sinyal bernilai negatif maka bit pesan adalah 0. Jika sinyal bernilai positif maka bit pesan adalah 1 dan disimpan ke dalam variabel bitPesan.

c. Proses Konversi Bit Pesan ke Kode ASCII Terenkripsi

```
pesanEktraksi=[];
indexAscii=1;
for i=1:16:panjangBitPesan
    pesanDesimal=bin2dec(bitPesan(i:i+15));
    indexAscii=indexAscii+1;
end
```

Source code di atas digunakan untuk mengubah bit pesan menjadi kode ASCII Terenkripsi dan disimpan ke dalam variabel pesanDesimal.

d. Proses Menghitung BER (*Bit Error Rate*)

```
bitpesanAsli=getappdata(0,'bitPesan');
bitError=0;
for i=1:panjangBitPesan
    if(str2double(bitPesan(i))~=str2double(bitpesanAsli(i+16)))
        bitError=bitError+1;
```

```
    end
end
ber=(bitError/panjangBitPesan)*100;
```

Source code di atas digunakan untuk mencari *bit error rate*. Bit pesan asli akan dibandingkan menggunakan *looping* dengan bit pesan ekstrak. Jika ditemukan bit tidak sama maka akan disimpan dalam bentuk jumlah kedalam variabel bitError. Setelah *looping* selesai bit *error* dihitung menggunakan rumus *bit error rate*.

3.5.7 Dekripsi Pesan

```
p=get(handles.pEnkripsiForm,'String');
q=get(handles.qEnkripsiForm,'String');
fid=fopen('teks_ekstraksi.txt','r');
pesanEnkripsi=fread(fid,[1,inf],'char');
panjangPesan=length(pesanEnkripsi);
asciiPesanEnkripsi=double(char(pesanEnkripsi));
[n,Phi,e,d]=inisialisasi(str2double(p),str2double(q));
for j=1:panjangPesan
    decipher(j)=crypt(asciiPesan(j),n,d);
end
pesanDekripsi=char(decipher);
```

Source code di atas digunakan untuk dekripsi pesan terenkripsi menjadi pesan asli. Teks pesan terenkripsi diubah menjadi ASCII pesan terenkripsi, kemudian nilai n, Phi, e, dan d dicari menggunakan fungsi inisialisasi. ASCII pesan terenkripsi di dekripsi menjadi ASCII pesan menggunakan fungsi crypt dan disimpan ke variabel decipher. ASCII pesan diubah menjadi karakter menggunakan fungsi char dan disimpan ke dalam variabel pesanDekripsi.

3.6 Pengujian Sistem

Dalam dunia telekomunikasi, pertukaran informasi jarak jauh akan sangat rentan terhadap gangguan selama proses pengirimannya, baik yang disengaja maupun yang tidak disengaja. Gangguan yang tidak disengaja dapat berupa *noise*, interferensi, maupun redaman pada saluran transmisi. Sedangkan gangguan yang disengaja tentu saja dilakukan oleh pihak-pihak yang tidak bertanggung jawab sebelum data tersebut sampai ke tangan penerima. Hal ini menyebabkan data yang diterima tidak lagi sama dengan yang dikirim oleh pengirim, atau bahkan mengalami kerusakan.

Pada tugas akhir ini dilakukan simulasi pengujian sistem steganografi dengan *noise Gaussian* dan *noise Salt & Pepper*. Hasilnya kemudian dianalisis apakah sistem sudah bekerja dengan baik atau belum dengan menggunakan parameter parameter tertentu.

3.6.1 Noise Gaussian

Dalam pentransmisiannya, citra dapat saja mengalami gangguan yang berupa derau *uniform* atau *noise*. *Noise* merupakan sinyal yang tidak diharapkan karena bersifat mengganggu serta kehadirannya tidak bisa ditentukan (acak). Namun, *noise* hampir selalu ada dalam setiap sistem telekomunikasi, khususnya pada sistem transmisi. Terdapat berbagai jenis *noise* yang sering menyerang citra, salah satunya yang sering muncul adalah *noise Gaussian*. *Noise Gaussian* merupakan model *noise* yg mengikuti distribusi normal standar dengan rata-rata nol dan standar deviasi 1. Efek dari *noise* ini adalah munculnya titik-titik berwarna yg jumlahnya sama dengan persentase *noise*. Hal ini tentunya dapat menyebabkan kesalahan pada informasi yang diterima. Pada tugas akhir ini, dilakukan pengujian pada sistem dengan mengubah-ubah intensitas *noise* untuk mengetahui tingkat ketahanan citra steganografi.

3.6.2 Noise Salt & Pepper

Noise Salt & Pepper ialah model *noise* seperti taburan garam serta lada yang membagikan warna putih serta gelap pada titik yang terserang noise.

3.7 Performansi Sistem

Evaluasi terhadap kualitas sistem ini dicoba dengan evaluasi obyektif.

3.7.1 Penilaian Obyektif

Analisis kualitas hasil dari penyisipan teks pada *image cover* berdasarkan penilaian obyektif dilakukan dengan mengukur nilai parameter PSNR (*Peak Signal to Noise Ratio*), CER (*Character Error Rate*) dan BER (*Bit Error Rate*). Sedangkan penilaian objektif metode kriptografi dilakukan dengan mengukur *Avalanche Effect* dan *Brute Force Attack*.

a. *Peak Signal to Noise Ratio(PSNR)*

PSNR merupakan nilai perbandingan antara harga maksimum dari intensitas citra terhadap *error* citra yaitu MSE. Lebih jelasnya, MSE adalah nilai yang menyatakan rata-rata kuadrat *error*, dalam hal ini *error* menyatakan selisih antar citra dimana kedua citra yang dibandingkan memiliki ukuran yang sama. Oleh karena itu, sebelum dapat menghitung PSNR suatu citra harus menghitung nilai MSE terlebih dahulu. Untuk

menghitung nilai MSE digunakan persamaan berikut:

$$MSE = \frac{1}{M.N}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(g(x,y) - f(x,y))^2 \quad (3.1)$$

Sementara penghitungan nilai PSNR dilakukan dengan menggunakan persamaan berikut:

$$PSNR = 20\log_{10}\left[\frac{255}{\sqrt{MSE}}\right] \quad (3.2)$$

PSNR yang semakin besar menandakan bahwa kualitas citra semakin bagus, hal ini karena *error* antara kedua citra semakin kecil.

b. *Character Error Rate (CER)*

Character Error Rate (CER) adalah perbandingan jumlah karakter yang *error* dengan total karakter. CER merupakan parameter pengujian yang digunakan untuk melihat kualitas pesan yang disisipkan. Penggunaan parameter BER 26 tidak cukup apabila tidak disertakan dengan pengujian terhadap parameter CER. Hal ini dikarenakan, nilai BER yang rendah belum berarti menghasilkan CER yang rendah juga. Berikut rumus untuk menghitung CER:

$$CER = \frac{Jumlah\ karakter\ salah}{Jumlah\ karakter\ keseluruhan} \times 100\% \quad (3.3)$$

c. *Bit Error Rate (BER)*

Bit Error Rate (BER) merupakan parameter pengukuran obyektif juga yang dipakai untuk mengatur ketepatan data hasil ekstraksi. Dengan cara menghitung persentase bit yang salah dari hasil ektraksi dengan bit keseluruhan. Secara matematis, nilai BER dapat dihitung dengan:

$$BER = \frac{Jumlah\ bit\ error}{Jumlah\ bit\ keseluruhan} \times 100\% \quad (3.4)$$

d. *Avalanche Effect*

Avalanche Effect menunjukan persentase perhitungan rasio jumlah bit beda antara perubahan bit-bit *plaintext* ke dalam bit-bit *ciphertext* terhadap jumlah bit total. Secara matematis penilaian *Avalanche Effect* ditunjukan pada persamaan 3.5

$$Avalanche\ effect = \frac{Jumlah\ Bit\ Beda}{Jumlah\ Bit\ Total}\ x\ 100\ \% \quad (3.5)$$

Semakin tinggi nilai Avalanche Effect yang dihasilkan maka kekuatan ciphertext semakin kuat.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan analisa, menggabungkan teknik Steganografi *Discrete Wavelet Transform* Dua Dimensi (2-D) dan Algoritma Kriptografi RSA pada perancangan dan analisis keamanan pesan, dapat diambil kesimpulan sebagai berikut:

1. Steganografi pada citra digital dengan menerapkan teknik Steganografi *Discrete Wavelet Transform* 2-Dimensi dapat melakukan penyembunyian data teks kedalam berkas gambar.
2. Steganografi dapat digunakan untuk proses penyembunyian suatu data pesan teks dalam berkas citra digital.
3. Metode RSA dapat digunakan untuk mengenkripsi pesan yang akan disisipkan.
4. Berdasarkan dari beberapa pengujian yang telah dilakukan pada sistem, telah diperoleh beberapa hasil performansi dengan nilai rata-rata meliputi Avalanche Effect sebesar 7.33 %, Peak Signal to Noise Ratio (PSNR) sebesar 62,3657%, Bit Error Rate (BER) dan Character Error Rate (CER) sebesar 0%.

5.2 Saran

Beberapa hal yang disarankan dalam menggabungkan teknik Steganografi *Discrete Wavelet Transform* Dua Dimensi (2-D) dan Algoritma Kriptografi RSA pada perancangan dan analisis keamanan pesan ini adalah sebagai barikut:

1. Aplikasi ini menggunakan berkas citra digital dengan format .BMP, diharapkan kedepannya aplikasi dapat dikembangkan menggunakan berkas citra digital dengan format lain.
2. Dalam penyisipan data, aplikasi ini menggunakan metode *Discrete Wavelet Transform* 2-Dimensi, dan disarankan untuk mengembangkannya dengan metode lain seperti metode Masking and Filtering, Transformation untuk membandingkan metode yang tepat untuk steganografi.

DAFTAR PUSTAKA

[1] Wahana Komputer, *Memahami model enkripsi dan security data*, Andi: Yogyakarta, 2003.

[2] Sriani, Triase, and Khairuna, *pendekomposisian citra digital dengan algoritma DWT*, J. ILKOM, vol. 01, no. 01, ISSN: 2598-6341, 2017.

[3] W. Sutoyo T, Mulyanto E, Suhartono V, Nurhayati O. D, *Teori Pengolahan Citra Digital*, Andi: Yogyakarta, 2009.

[4] A. M. Faza, C. Slamet, D. Nursantika, *Analisis kinerja kompresi citra digital dengan komparasi DWT, DCT, dan Hybrid (DWT-DCT)*, JOIN, vol.1, no.1, ISSN 2527-9165, 2016.

[5] M. Tezar, *Implementasi dan Analisis Keamanan Teks Menggunakan Teknik Steganografi LSB dan Algoritma Kriptografi Relative Displacement Cipher*, Telkom University, Bandung, 2016.

[6] I. tropiana, G. abdillah, and R. ilyas, *pengamanan data user pada database menggunakan kriptografi rivest shamir adleman dan cipher block chaining*, SENTIKA, ISSN: 2089-9815, Yogyakarta, 2019.

[7] Y. D. Arifani, *jurnal enkripsi dan deskripsi kriptografi metode RSA*, Universitas PGRI Ronggolawe, Tuban, 2016.

[8] T. I. Saputra, Fauziah, N. Hayati, *Implementasi Discrete Wavelet Transform Pada Aplikasi Kompresi Citra Medis*, Jurnal Infomedia vol. 4 No.2: E-ISSN 2548 – 1180, Universitas Nasional, Desember 2019.

[9] G. G. Putri, W. Styoroni, R. D. Rahayani, Analisis Kriptografi Simetris AES dan Kriptografi Asimetris RSA pada Enkripsi Citra Digital, Ethos: 197-207, Politeknik Caltek Riau, Vol. 6, 2018.

[10] B. A. Wijaya, *Analasis Steganografi Video Menggunakan Kombinasi Algoritma Discrete Cosinetransform (DCT-2D) dan Discrete Wavelet Transform (DWT)*, Universitas Sumatera Utara, Medan, 2018.

[11] T. P. Utomo, *Steganografi gambar dengan Metode Least Significant Bit untuk proteksi komunikasi pada media online*, Universitas Islam Negeri Sunan Gunung Djati, Bandung, 2012.

[12] Sutarno, *Analisis Perbandingan Transformasi Wavelet Pada Pengenalan Citra Wajah*, GENERIC, Vol. 5 No.2: 15-21, 2010.

[13] M. I. Sikki, *Pengenalan wajah Menggunakan K-Nearest Neighbour Dengan Praproses Transformasi Wavelet*, PARADIGMA Vol. **x** No. **2**: 159-172, 2009.

[14] H. Beni, *Analisis Bicubic dan Bilinear Menggunakan Metode Discrete Wavelet Transform Pada Super Resolusi*, Universitas Dian Nuswantoro, Semarang, 2016

[15] C. E. Bire, *Denoising pada Citra Menggunakan Transformasi Wavelet*, seminar Nasional Teknologi Informasi & Komunikasi Terapan, 487-493, 2012.

[16] Hidayat, *Restorasi Bar Codes 2-D pada Citra Hasil Kamera Menggunakan Metode Wavelet*, Prosiding SNATIF Ke-1 : 233-240, 2014.

[17] Budiman A, *Kompresi citra medis menggunakan metode wavelet*, Agri-tek Volume 14: 80-87, 2013

[18] Meta, I. Fitri, I. Agustina, *Optimalisasi penggunaan Transformasi Wavelet Haar pada Citra Digital munggunakan Matlab*, Proseedings SNIT Hal. A-339-346, 2011.

[19] Ariyus, Dony, *Pengantar ilmu kriptografi : Teori Analisis dan Implementasi. Penerbit Andi, Yogyakarta, 2008.*

[20] Y. Fauziah, *Pengaman Pesan dalam Editor Text menggunakan Hybrid Cryptosystem*, Seminar Nasional Informatika, SemNasIF ISSN : 1979-2328, Yogyakarta, 24 Mei 2008.

[21] Sumarno, I. Gunawan, H. S. Tambunan, *Analisis Kinerja Kombinasi Algoritma Messege-Digest Algortihm 5 (MD5), Rivest Shamir Adleman (RSA) Dan Rivest Cipher 4 (RC4) Pada Keamanan E-Dokumen*, Jusikom Prima Vol. 2 No. 1, E-ISSN : 2580-2879, Juli 2018.

[22] Y. Kurniawan, *Kriptografi keamanan internet dan jaringan komunikasi*, Cetakan Pertama, Informatika, Bandung, 2004.

[23] H. Listiyono, *Implementasi Algoritma Kunci Publik Pada Algoritma RSA.* Dinamika Informatika-Vol.1 No.2, ISSN : 2085-3343, September 2009.

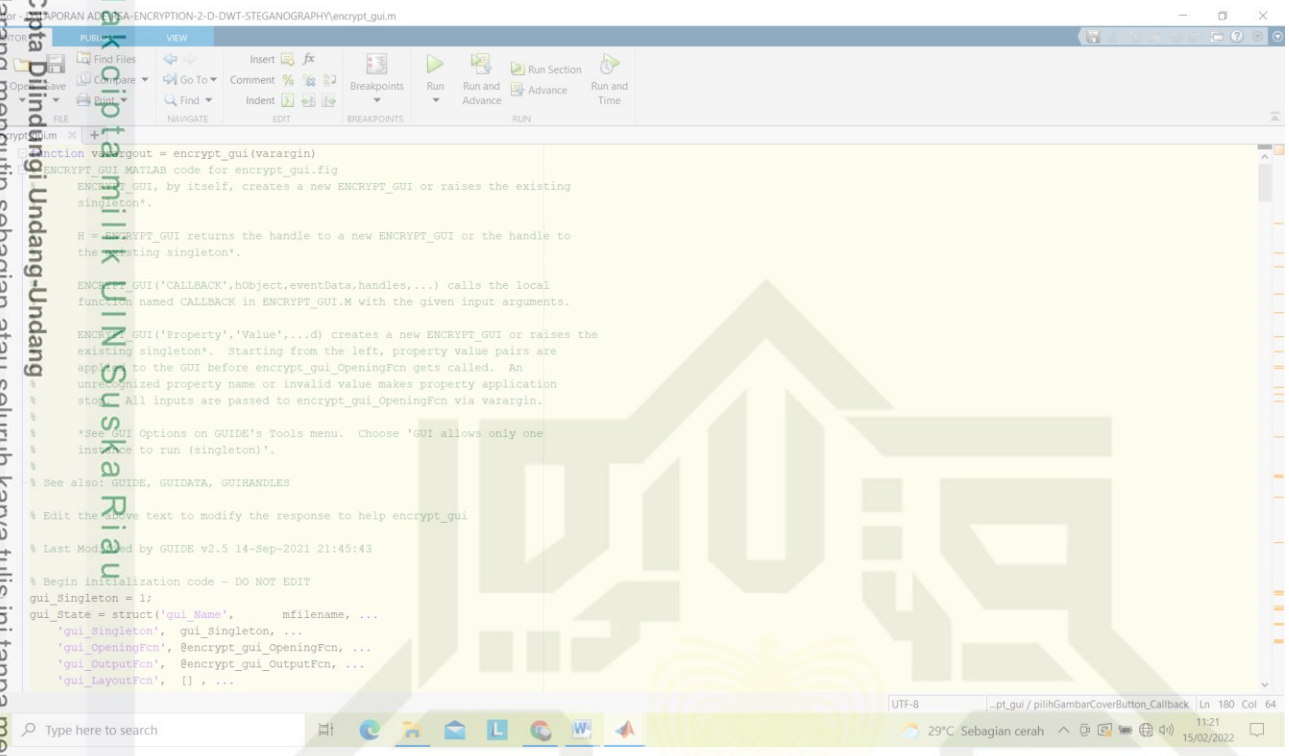
[24] A. Putri, M. Subali, *Implementation Technique With Steganography LSB Method in Digital Images*. Undergraduate Program, Faculty of Computer Science, Gunadarma University, 2009

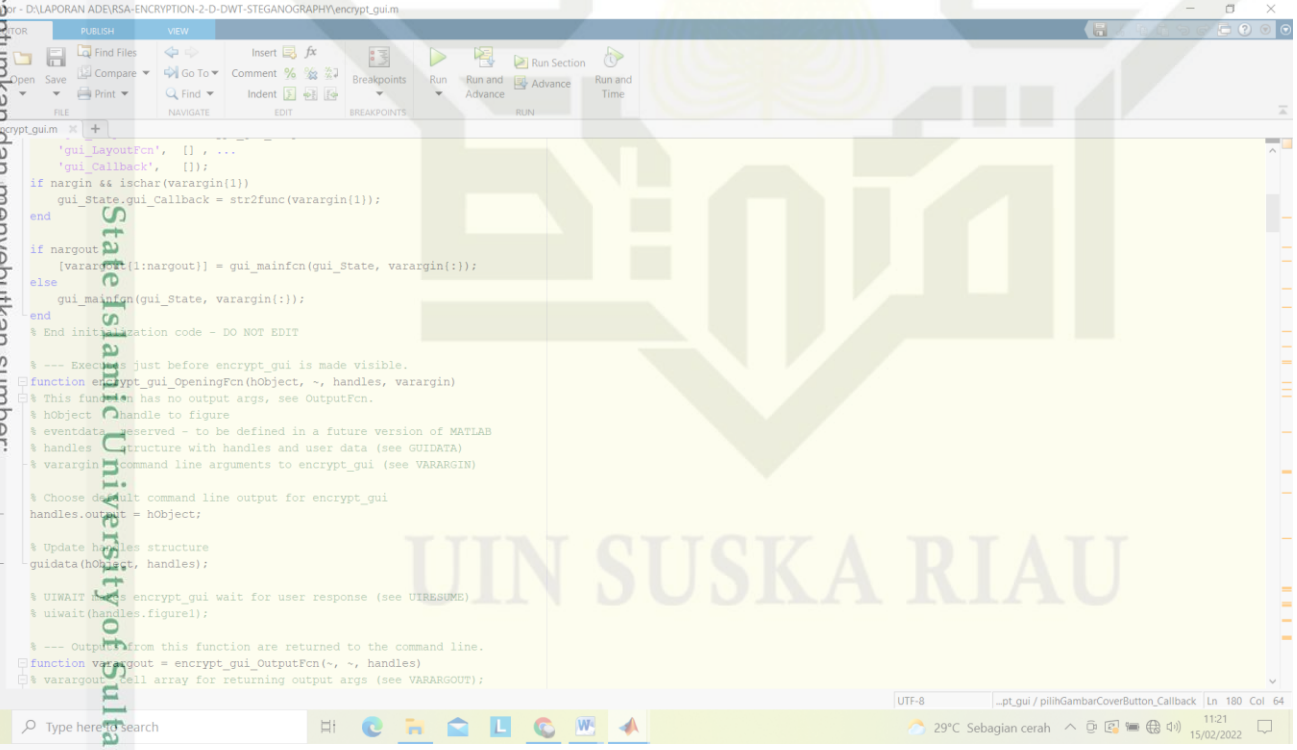
[25] P. Cop, R. A. Putra, *Sistem Keamanan E-Voting Menggunakan Algoritma KODE ASCII*, ejurnal.bsi Vol. 1 No. 1, E-ISSN : 2550-0120, 1 Februari 2015

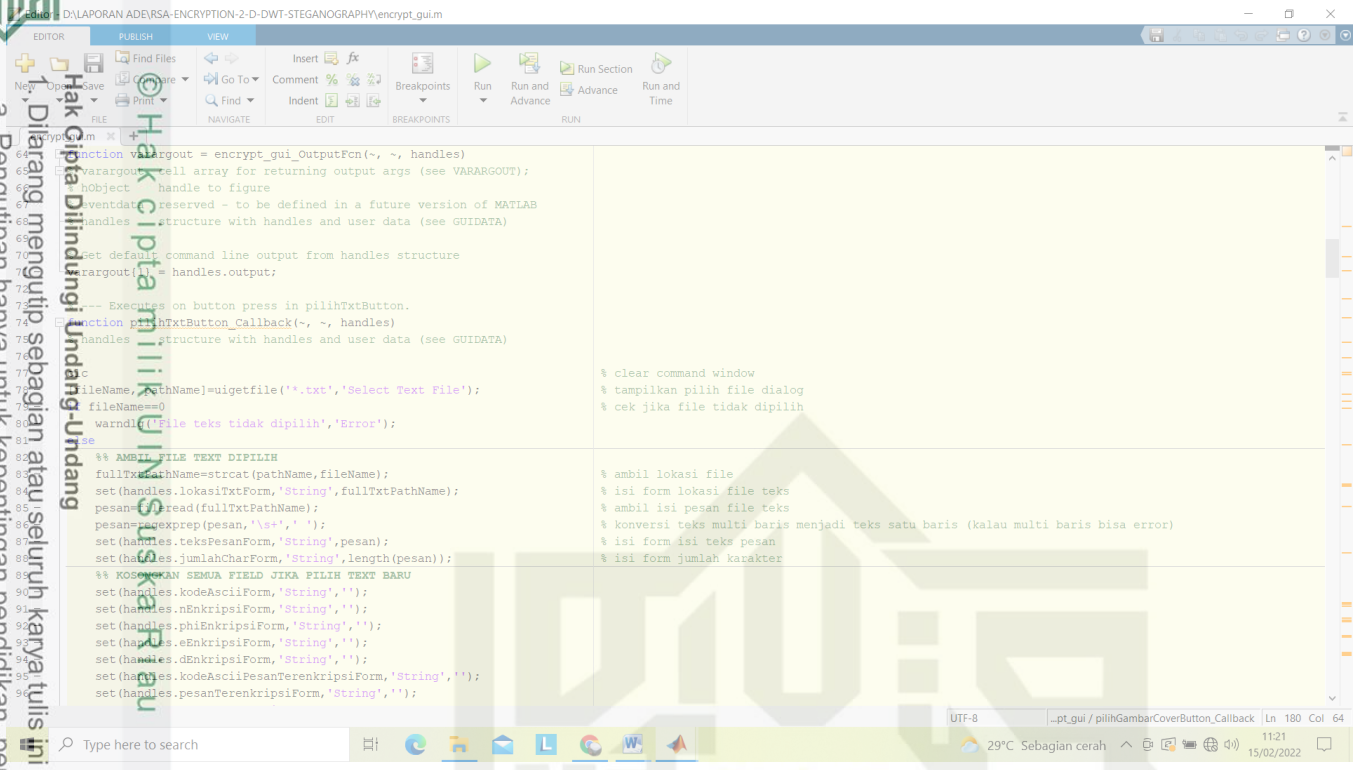
[26] Yanita, *Pengantar Teori Bilangan*, FMIPA Matemetika Universitas Andalas, Padang, 22 Februari 2014.

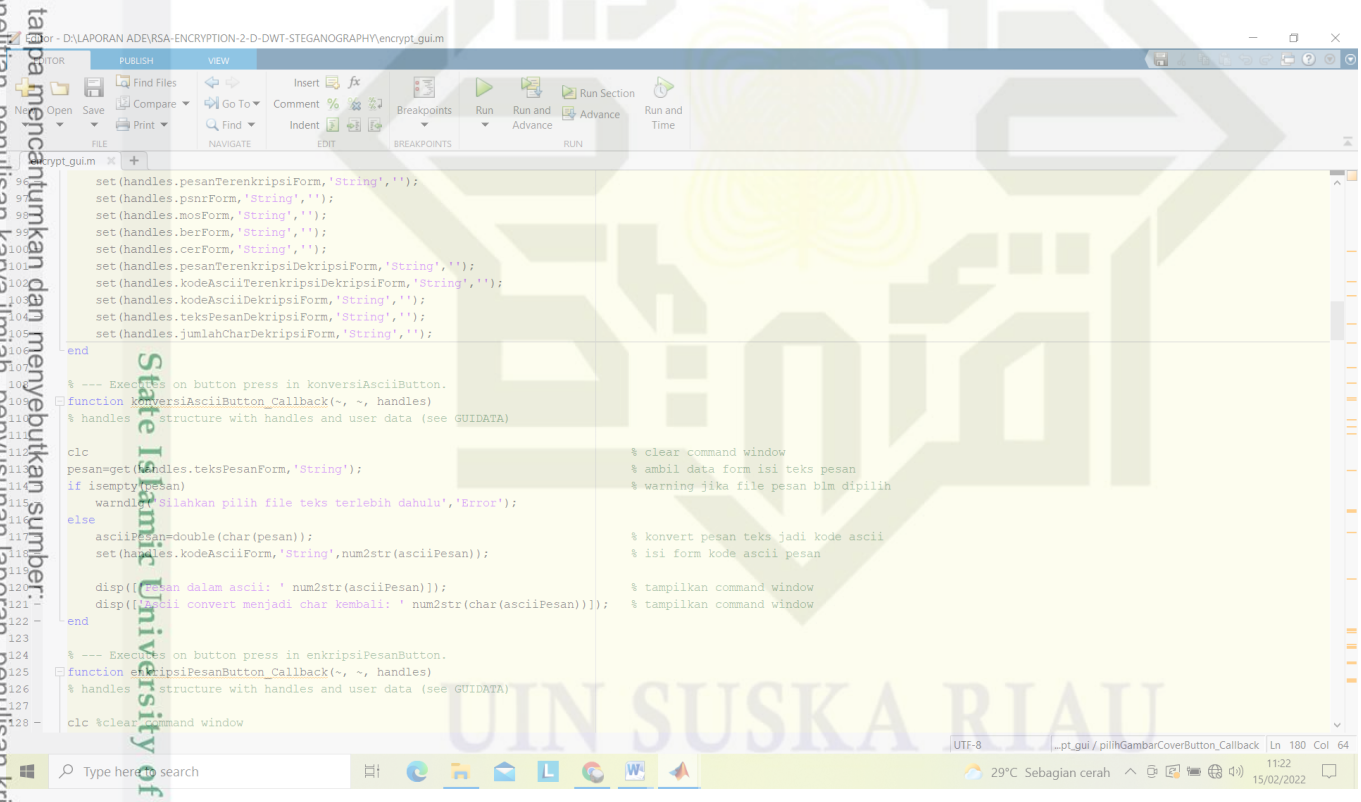
LAMPIRAN A

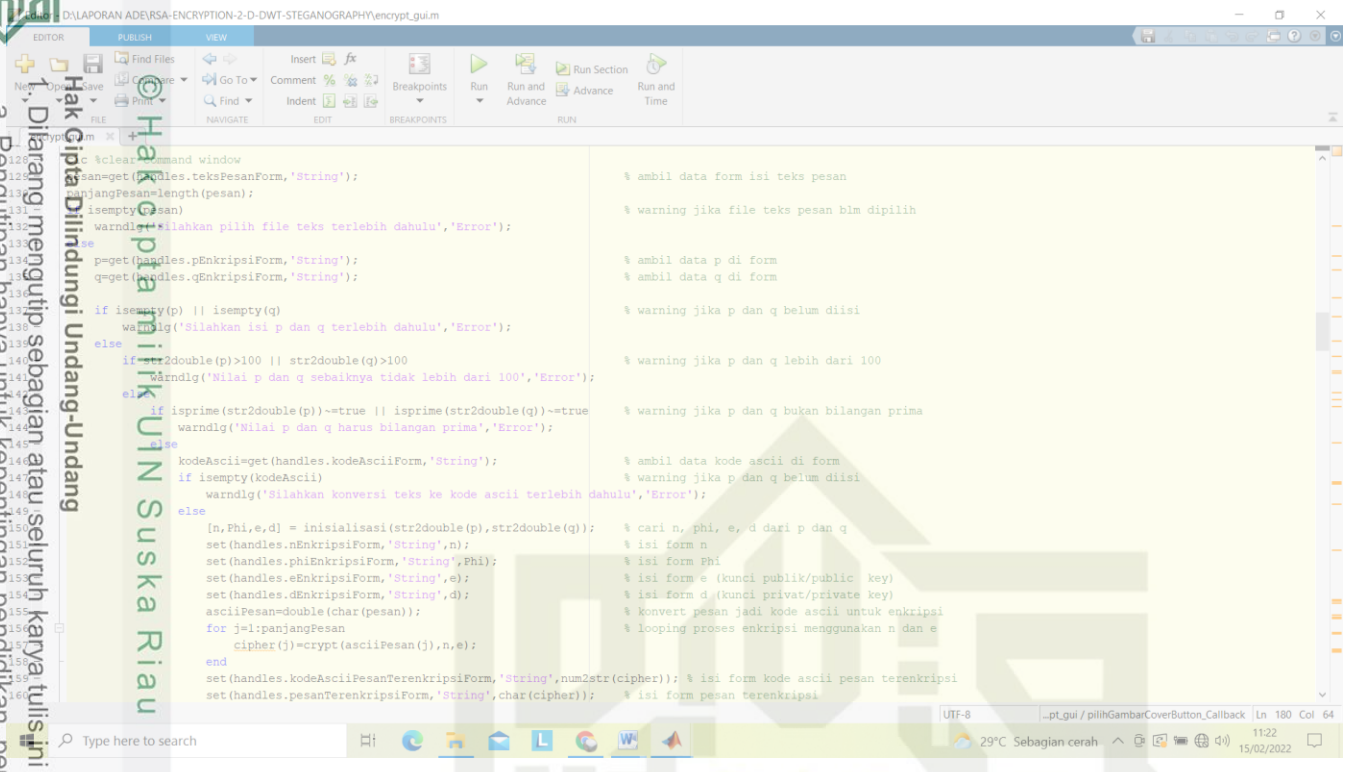
LIST PROGRAM



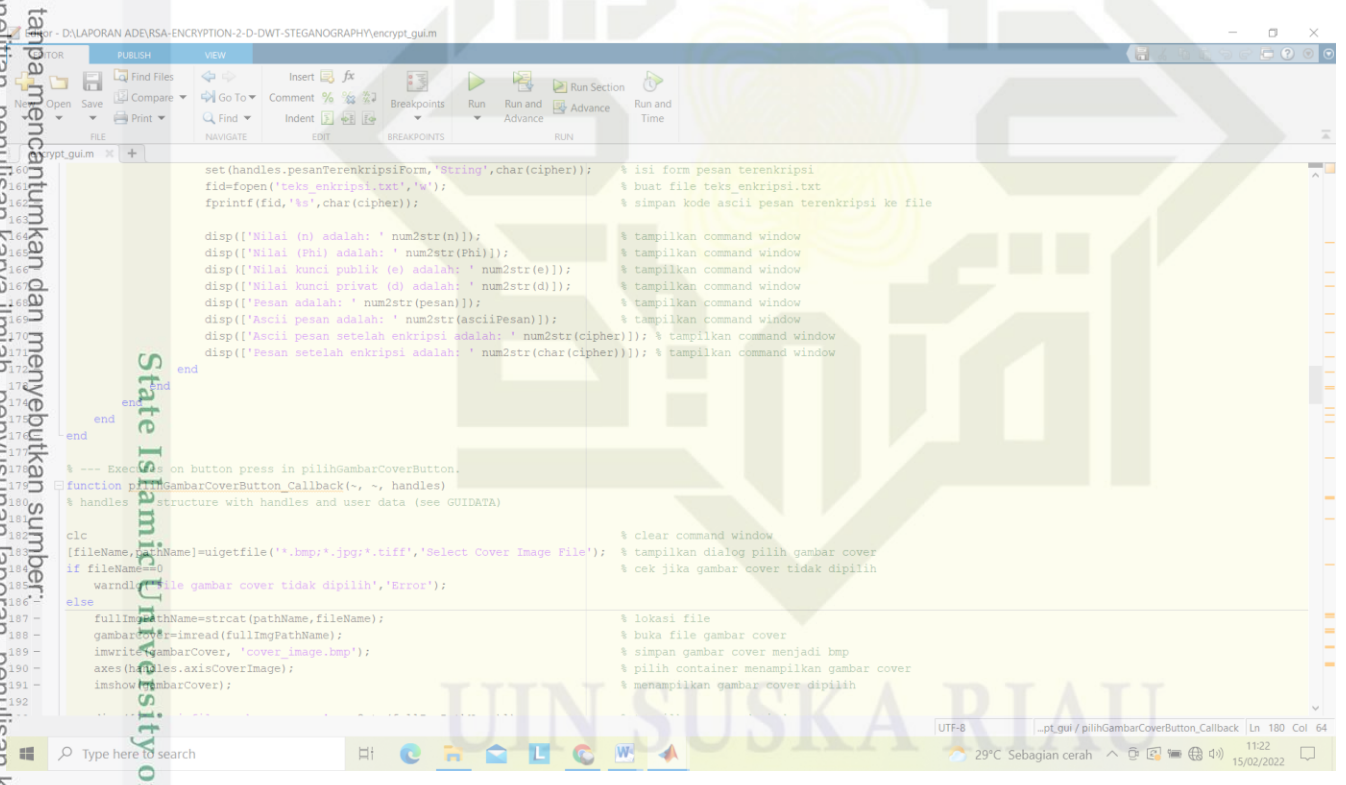




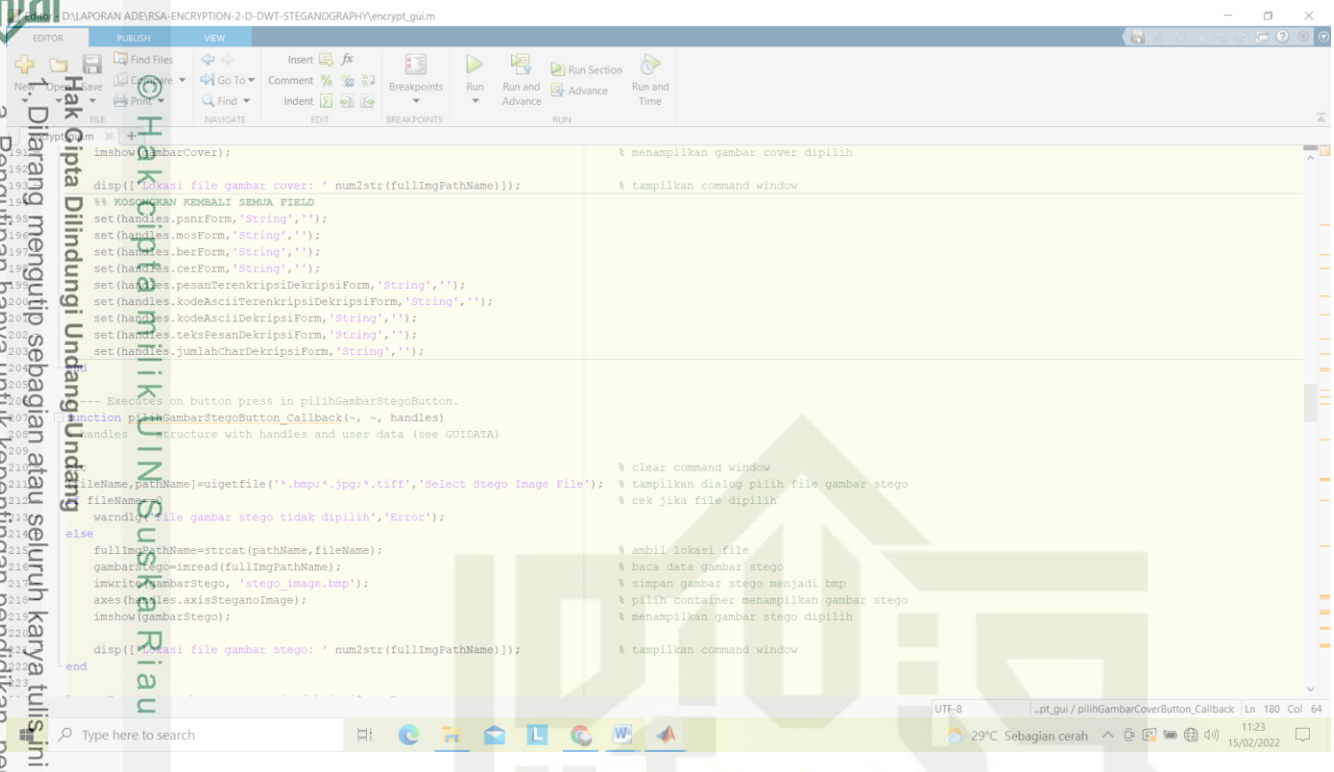


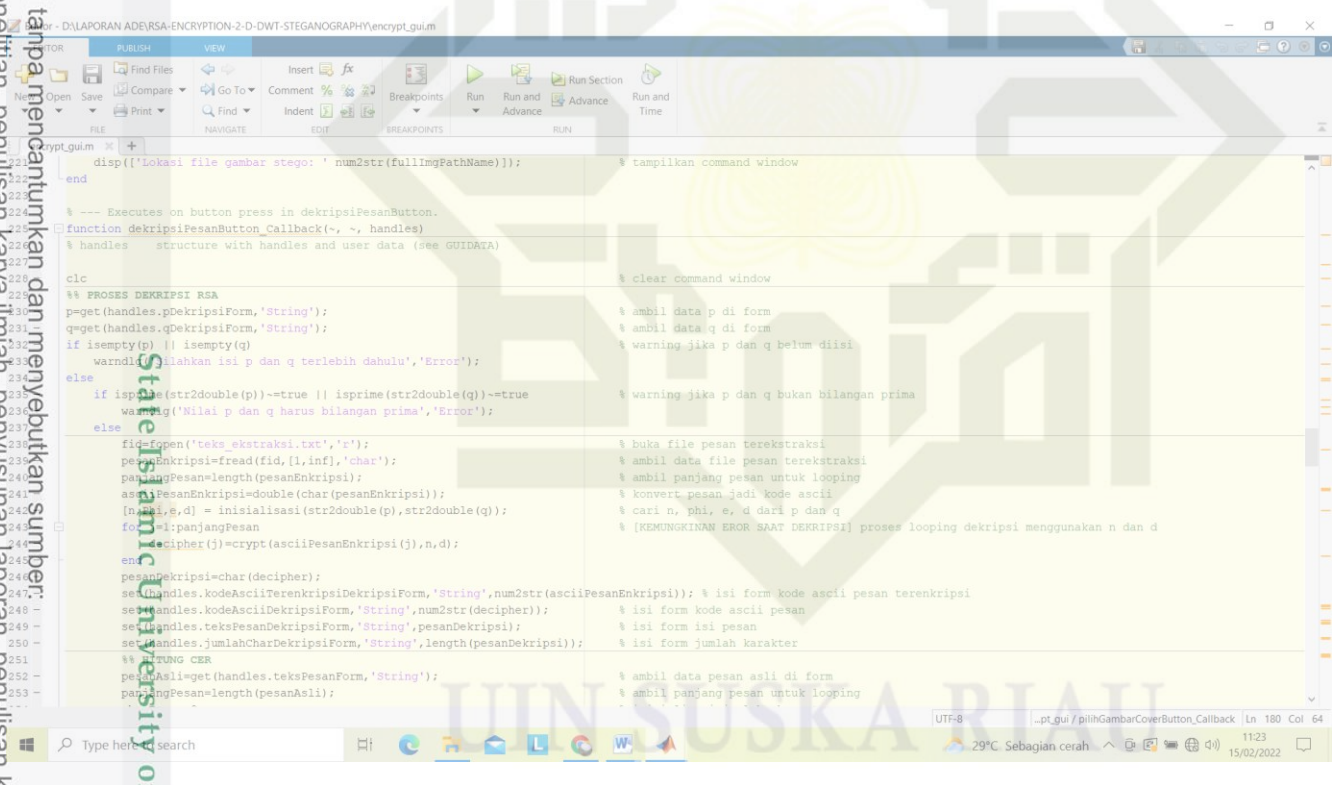


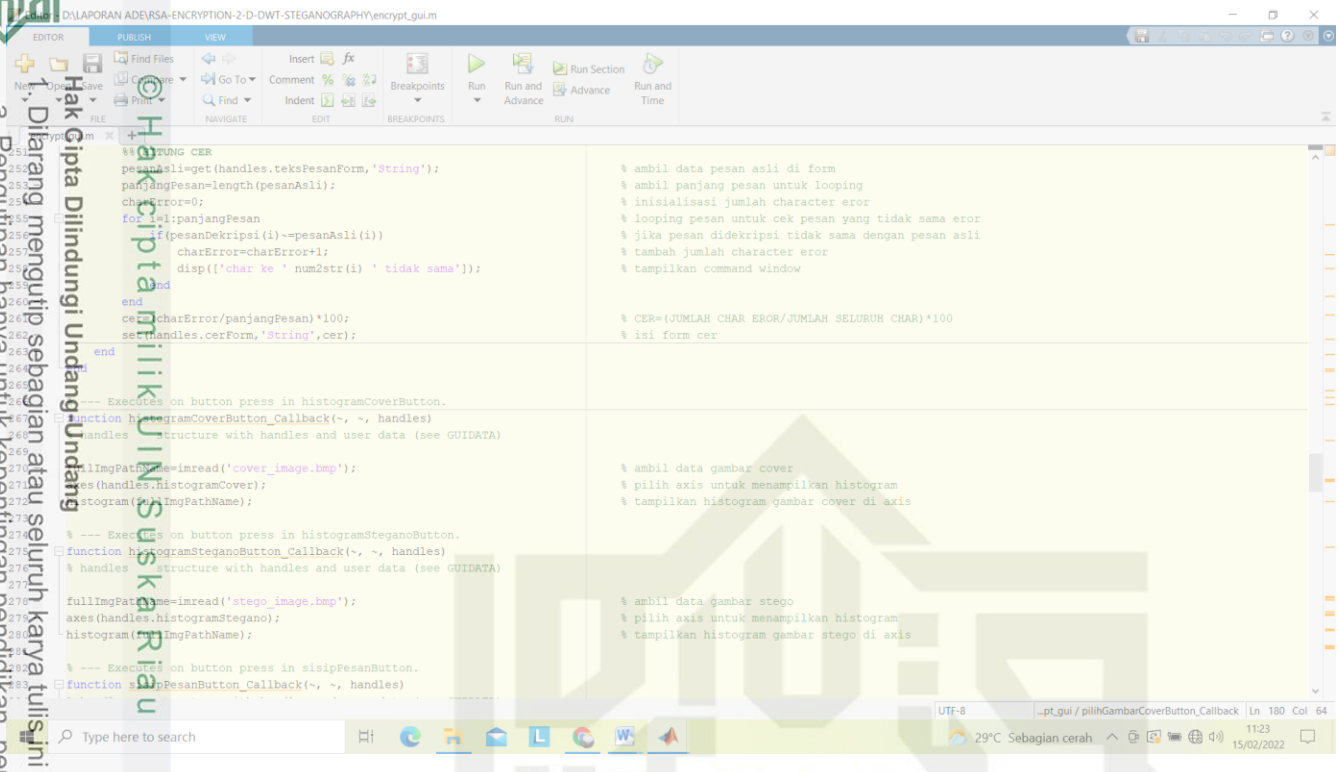
```
clc %clear command window
pesan=get(handles.teksPesanForm,'String');                  % ambil data form isi teks pesan
panjangPesan=length(pesan);
if isempty(pesan)                                           % warning jika file teks pesan blm dipilih
    warndlg('Silahkan pilih file teks terlebih dahulu','Error');
else
    p=get(handles.pEnkripsiForm,'String');                  % ambil data p di form
    q=get(handles.qEnkripsiForm,'String');                  % ambil data q di form

    if isempty(p) || isempty(q)                             % warning jika p dan q belum diisi
        warndlg('Silahkan isi p dan q terlebih dahulu','Error');
    else
        if str2double(p)>100 || str2double(q)>100           % warning jika p dan q lebih dari 100
            warndlg('Nilai p dan q sebaiknya tidak lebih dari 100','Error');
        else
            if isprime(str2double(p))~=true || isprime(str2double(q))~=true     % warning jika p dan q bukan bilangan prima
                warndlg('Nilai p dan q harus bilangan prima','Error');
            else
                kodeAscii=get(handles.kodeAsciiForm,'String');  % ambil data kode ascii di form
                if isempty(kodeAscii)                           % warning jika p dan q belum diisi
                    warndlg('Silahkan konversi teks ke kode ascii terlebih dahulu','Error');
                else
                    [n,Phi,e,d] = inisialisasi(str2double(p),str2double(q));    % cari n, phi, e, d dari p dan q
                    set(handles.nEnkripsiForm,'String',n);      % isi form n
                    set(handles.phiEnkripsiForm,'String',Phi);  % isi form Phi
                    set(handles.eEnkripsiForm,'String',e);      % isi form e (kunci publik/public  key)
                    set(handles.dEnkripsiForm,'String',d);      % isi form d (kunci privat/private key)
                    asciiPesan=double(char(pesan));             % konvert pesan jadi kode ascii untuk enkripsi
                    for j=1:panjangPesan                        % looping proses enkripsi menggunakan n dan e
                        cipher(j)=crypt(asciiPesan(j),n,e);
                    end
                    set(handles.kodeAsciiPesanTerenkripsiForm,'String',num2str(cipher)); % isi form kode ascii pesan terenkripsi
                    set(handles.pesanTerenkripsiForm,'String',char(cipher));    % isi form pesan terenkripsi
```

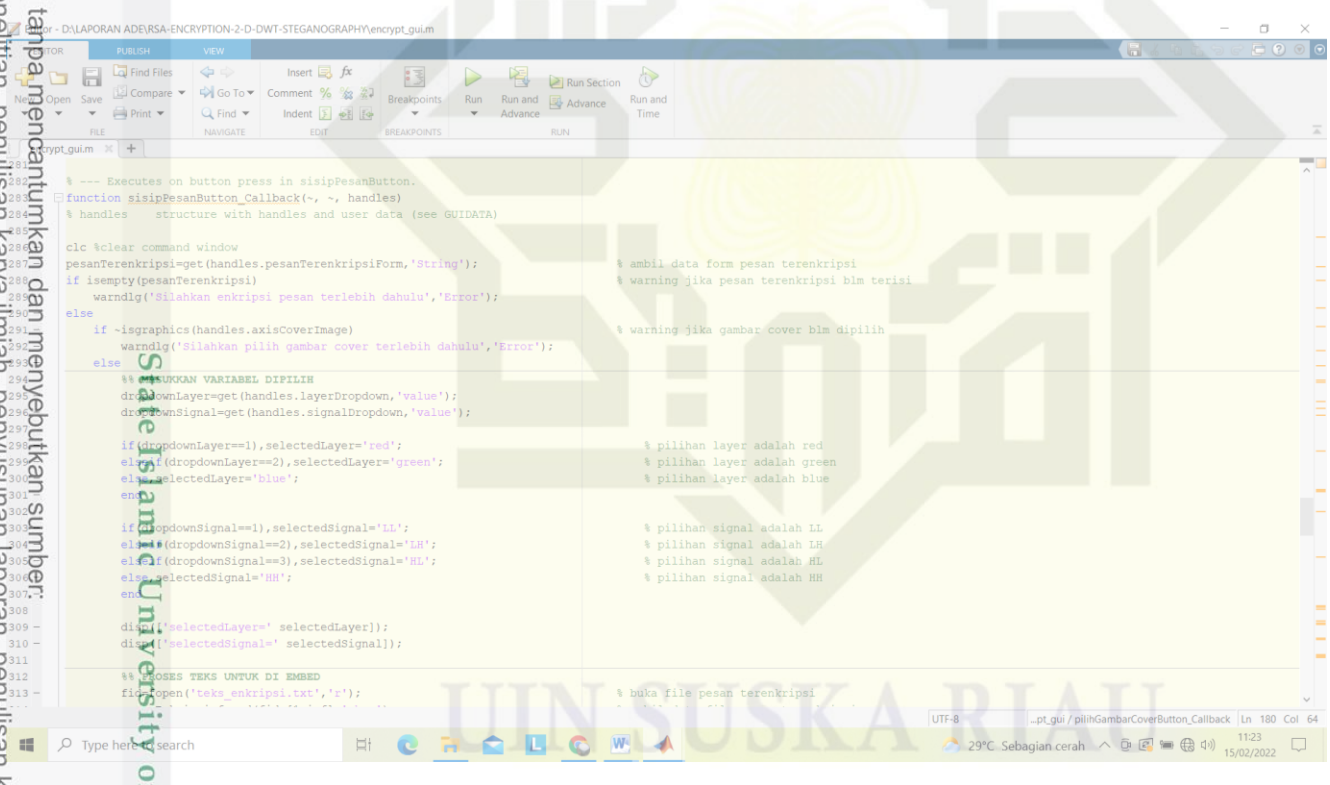


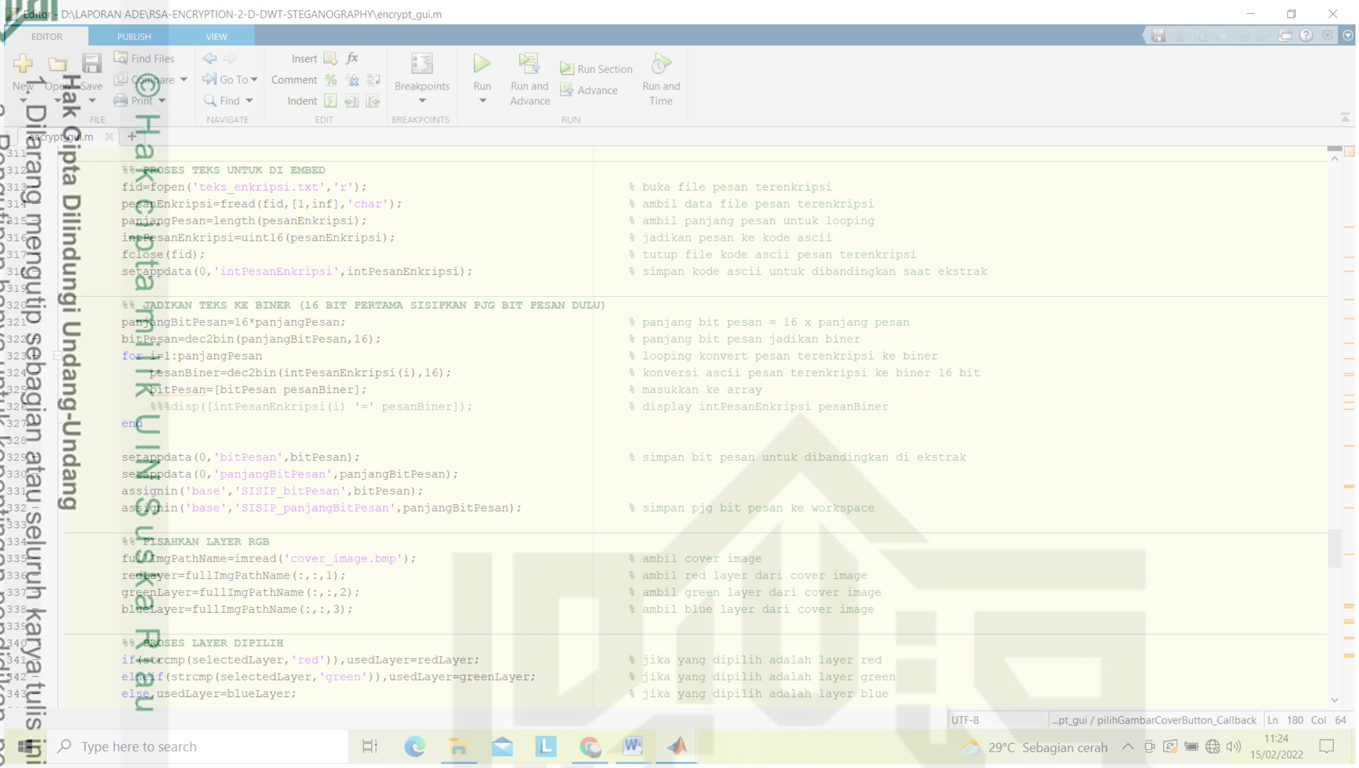
```
                    set(handles.pesanTerenkripsiForm,'String',char(cipher));    % isi form pesan terenkripsi
                    fid=fopen('teks_enkripsi.txt','w');         % buat file teks_enkripsi.txt
                    fprintf(fid,'%s',char(cipher));             % simpan kode ascii pesan terenkripsi ke file

                    disp(['Nilai (n) adalah: ' num2str(n)]);    % tampilkan command window
                    disp(['Nilai (Phi) adalah: ' num2str(Phi)]);% tampilkan command window
                    disp(['Nilai kunci publik (e) adalah: ' num2str(e)]);   % tampilkan command window
                    disp(['Nilai kunci privat (d) adalah: ' num2str(d)]);   % tampilkan command window
                    disp(['Pesan adalah: ' num2str(pesan)]);    % tampilkan command window
                    disp(['Ascii pesan adalah: ' num2str(asciiPesan)]);     % tampilkan command window
                    disp(['Ascii pesan setelah enkripsi adalah: ' num2str(cipher)]); % tampilkan command window
                    disp(['Pesan setelah enkripsi adalah: ' num2str(char(cipher))]); % tampilkan command window
                end
            end
        end
    end
end

% --- Executes on button press in pilihGambarCoverButton.
function pilihGambarCoverButton_Callback(~, ~, handles)
% handles    structure with handles and user data (see GUIDATA)

clc                                                         % clear command window
[fileName,pathName]=uigetfile('*.bmp;*.jpg;*.tiff','Select Cover Image File');  % tampilkan dialog pilih gambar cover
if fileName==0                                              % cek jika gambar cover tidak dipilih
    warndlg('File gambar cover tidak dipilih','Error');
else
    fullImgPathName=strcat(pathName,fileName);              % lokasi file
    gambarCover=imread(fullImgPathName);                    % buka file gambar cover
    imwrite(gambarCover, 'cover_image.bmp');                % simpan gambar cover menjadi bmp
    axes(handles.axisCoverImage);                           % pilih container menampilkan gambar cover
    imshow(gambarCover);                                    % menampilkan gambar cover dipilih
```



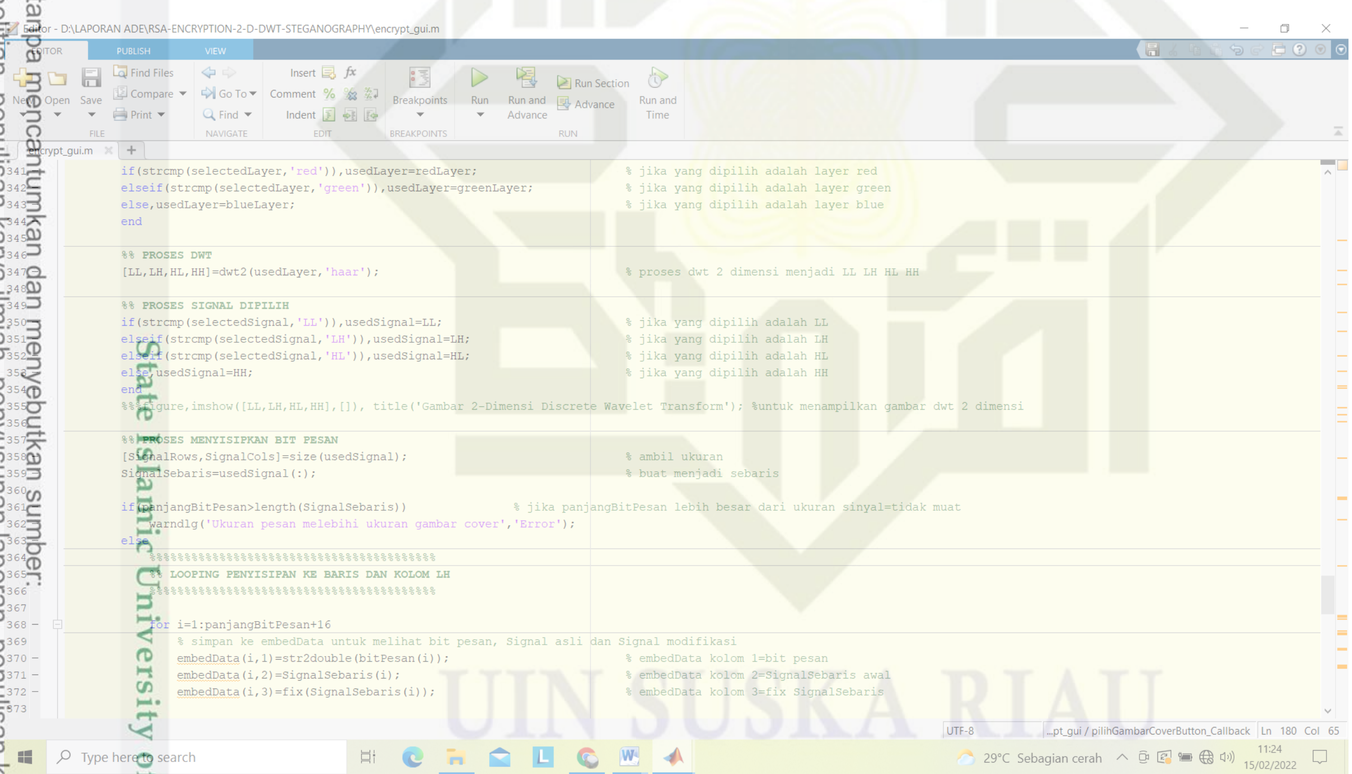




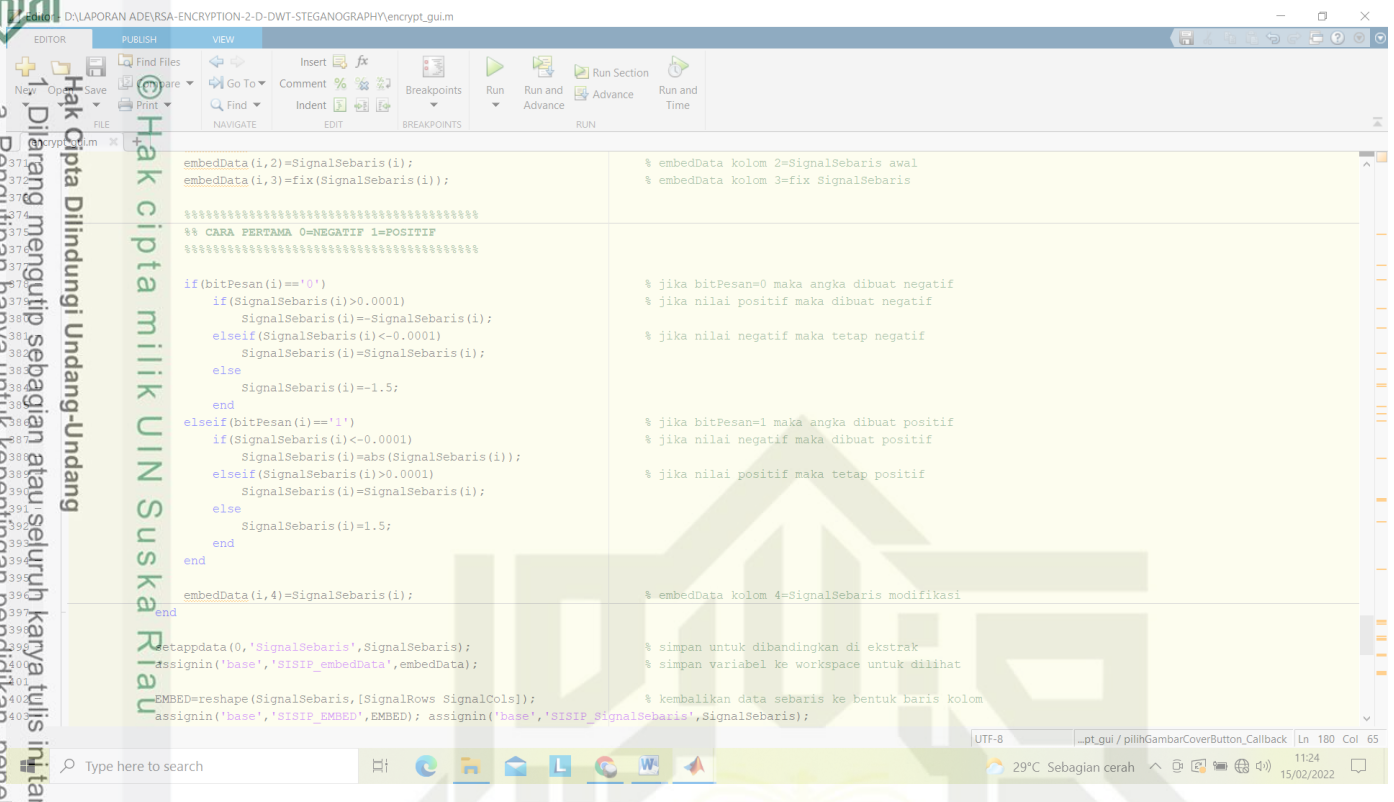


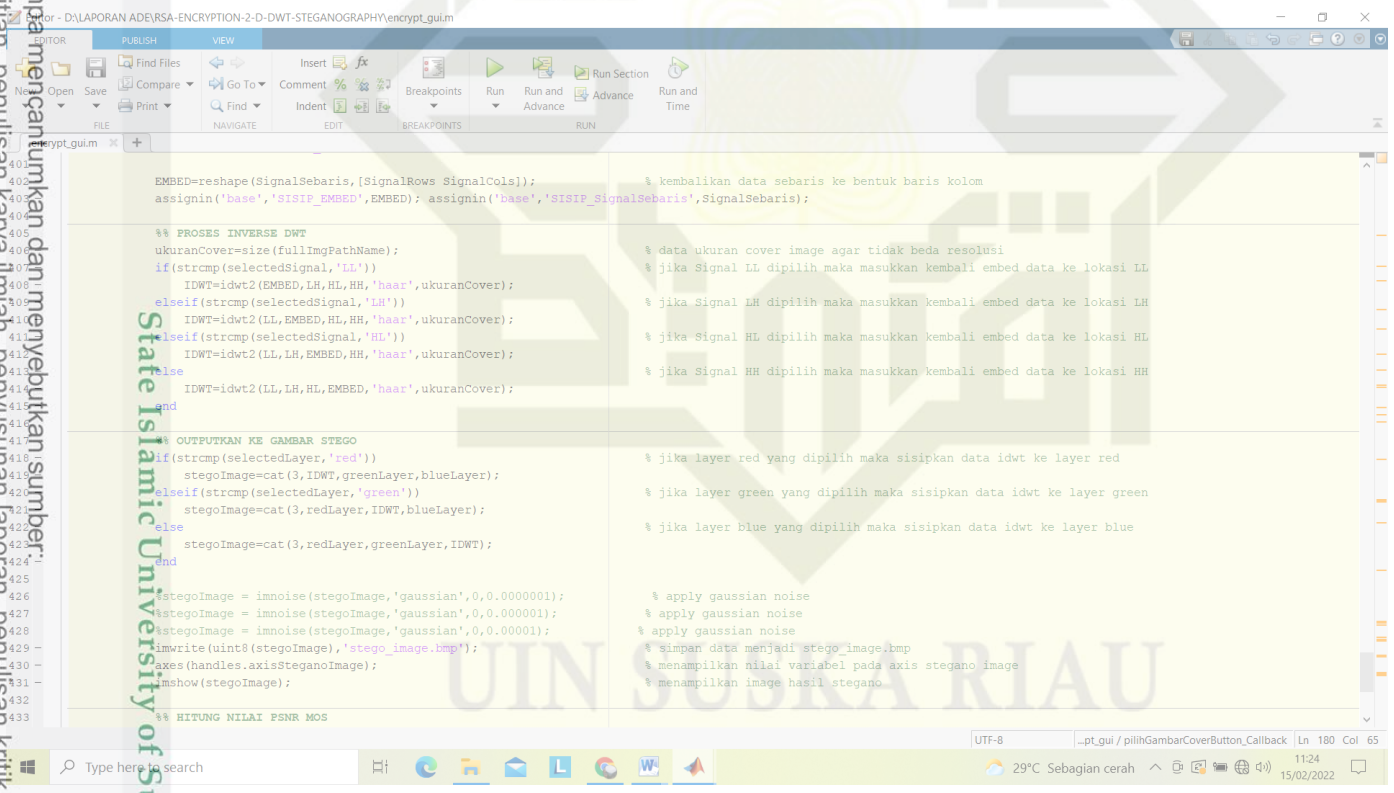


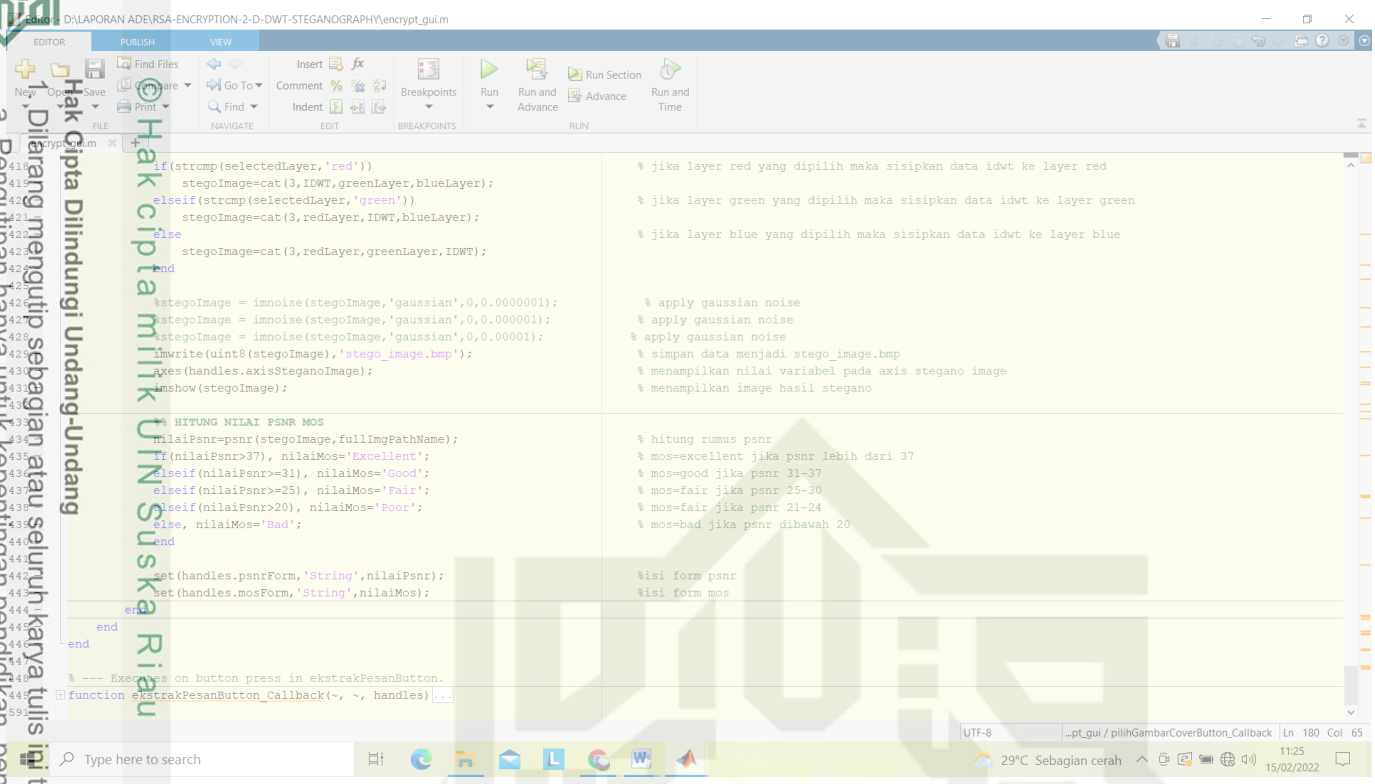
```
%% PROSES TEKS UNTUK DI EMBED
fid=fopen('teks_enkripsi.txt','r');                        % buka file pesan terenkripsi
pesanEnkripsi=fread(fid,[1,inf],'char');                    % ambil data file pesan terenkripsi
panjangPesan=length(pesanEnkripsi);                         % ambil panjang pesan untuk looping
intPesanEnkripsi=uint16(pesanEnkripsi);                     % jadikan pesan ke kode ascii
fclose(fid);                                                % tutup file kode ascii pesan terenkripsi
setappdata(0,'intPesanEnkripsi',intPesanEnkripsi);          % simpan kode ascii untuk dibandingkan saat ekstrak

%% JADIKAN TEKS KE BINER (16 BIT PERTAMA SISIPKAN PJG BIT PESAN DULU)
panjangBitPesan=16*panjangPesan;                            % panjang bit pesan = 16 x panjang pesan
bitPesan=dec2bin(panjangBitPesan,16);                       % panjang bit pesan jadikan biner
for i=1:panjangPesan                                        % looping konvert pesan terenkripsi ke biner
   pesanBiner=dec2bin(intPesanEnkripsi(i),16);              % konversi ascii pesan terenkripsi ke biner 16 bit
   bitPesan=[bitPesan pesanBiner];                          % masukkan ke array
   %%%disp([intPesanEnkripsi(i) '=' pesanBiner]);           % display intPesanEnkripsi pesanBiner
end

setappdata(0,'bitPesan',bitPesan);                          % simpan bit pesan untuk dibandingkan di ekstrak
setappdata(0,'panjangBitPesan',panjangBitPesan);
assignin('base','SISIP_bitPesan',bitPesan);
assignin('base','SISIP_panjangBitPesan',panjangBitPesan);   % simpan pjg bit pesan ke workspace

%% PISAHKAN LAYER RGB
fullImgPathName=imread('cover_image.bmp');                  % ambil cover image
redLayer=fullImgPathName(:,:,1);                            % ambil red layer dari cover image
greenLayer=fullImgPathName(:,:,2);                          % ambil green layer dari cover image
blueLayer=fullImgPathName(:,:,3);                           % ambil blue layer dari cover image

%% PROSES LAYER DIPILIH
if(strcmp(selectedLayer,'red')),usedLayer=redLayer;         % jika yang dipilih adalah layer red
elseif(strcmp(selectedLayer,'green')),usedLayer=greenLayer; % jika yang dipilih adalah layer green
else,usedLayer=blueLayer;                                   % jika yang dipilih adalah layer blue
```



```
if(strcmp(selectedLayer,'red')),usedLayer=redLayer;         % jika yang dipilih adalah layer red
elseif(strcmp(selectedLayer,'green')),usedLayer=greenLayer; % jika yang dipilih adalah layer green
else,usedLayer=blueLayer;                                   % jika yang dipilih adalah layer blue
end

%% PROSES DWT
[LL,LH,HL,HH]=dwt2(usedLayer,'haar');                       % proses dwt 2 dimensi menjadi LL LH HL HH

%% PROSES SIGNAL DIPILIH
if(strcmp(selectedSignal,'LL')),usedSignal=LL;              % jika yang dipilih adalah LL
elseif(strcmp(selectedSignal,'LH')),usedSignal=LH;          % jika yang dipilih adalah LH
elseif(strcmp(selectedSignal,'HL')),usedSignal=HL;          % jika yang dipilih adalah HL
else,usedSignal=HH;                                         % jika yang dipilih adalah HH
end
%%%figure,imshow([LL,LH,HL,HH],[]), title('Gambar 2-Dimensi Discrete Wavelet Transform'); %untuk menampilkan gambar dwt 2 dimensi

%% PROSES MENYISIPKAN BIT PESAN
[SignalRows,SignalCols]=size(usedSignal);                   % ambil ukuran
SignalSebaris=usedSignal(:);                                % buat menjadi sebaris

if(panjangBitPesan>length(SignalSebaris))          % jika panjangBitPesan lebih besar dari ukuran sinyal=tidak muat
   warndlg('Ukuran pesan melebihi ukuran gambar cover','Error');
else
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   %% LOOPING PENYISIPAN KE BARIS DAN KOLOM LH
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

   for i=1:panjangBitPesan+16
      % simpan ke embedData untuk melihat bit pesan, Signal asli dan Signal modifikasi
      embedData(i,1)=str2double(bitPesan(i));               % embedData kolom 1=bit pesan
      embedData(i,2)=SignalSebaris(i);                      % embedData kolom 2=SignalSebaris awal
      embedData(i,3)=fix(SignalSebaris(i));                 % embedData kolom 3=fix SignalSebaris
```

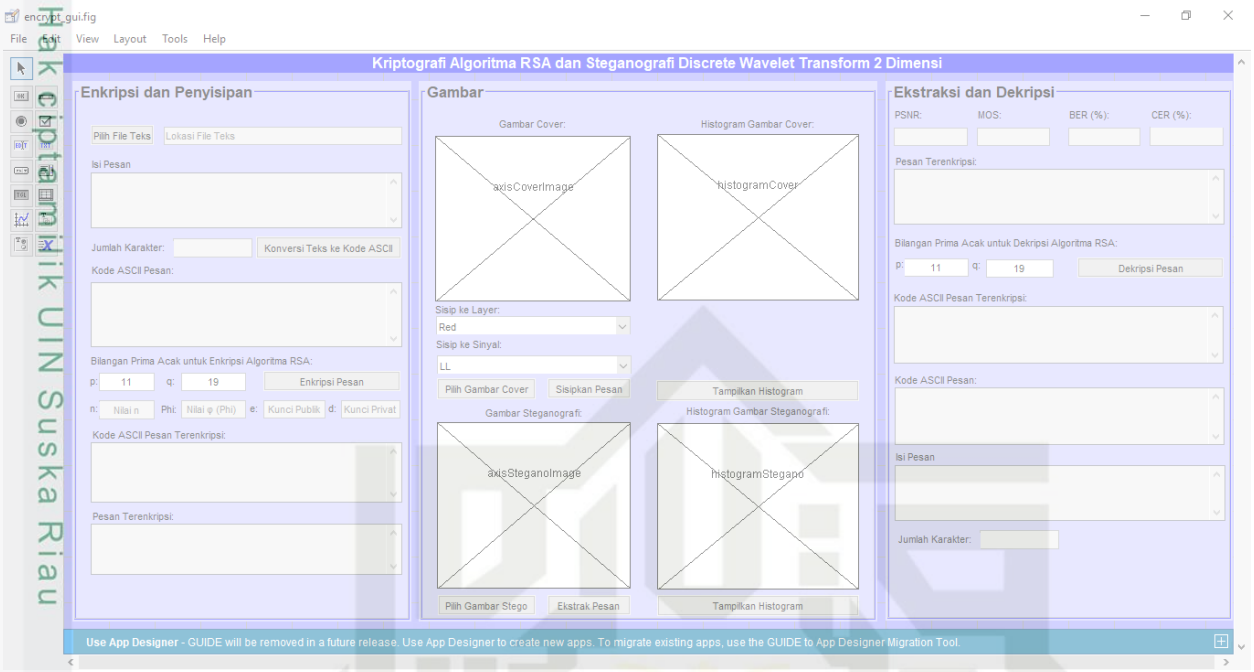




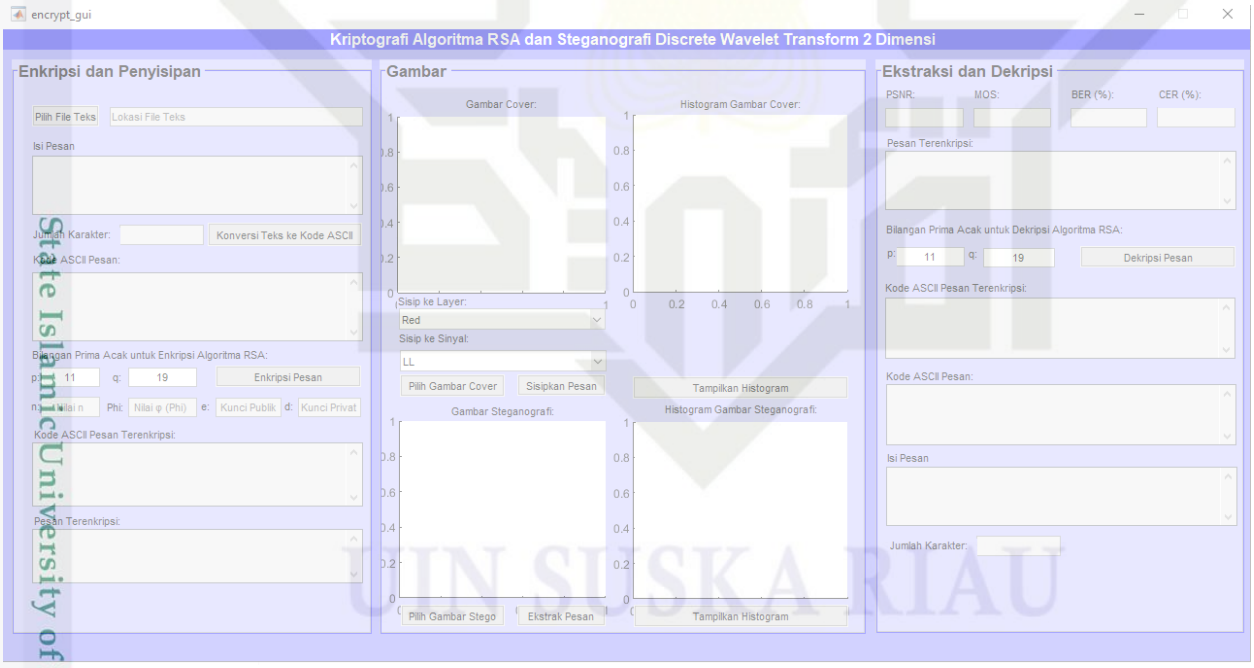


LAMPIRAN B

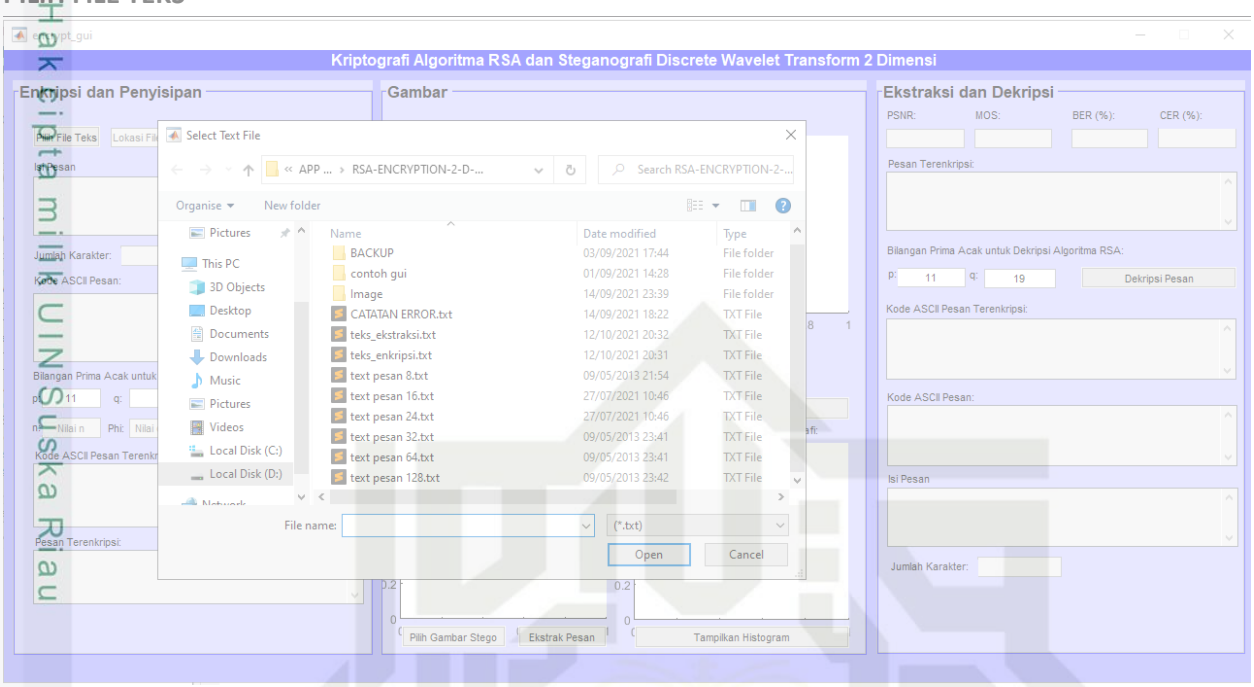
1. Gui SEBELUM DI *RUN*



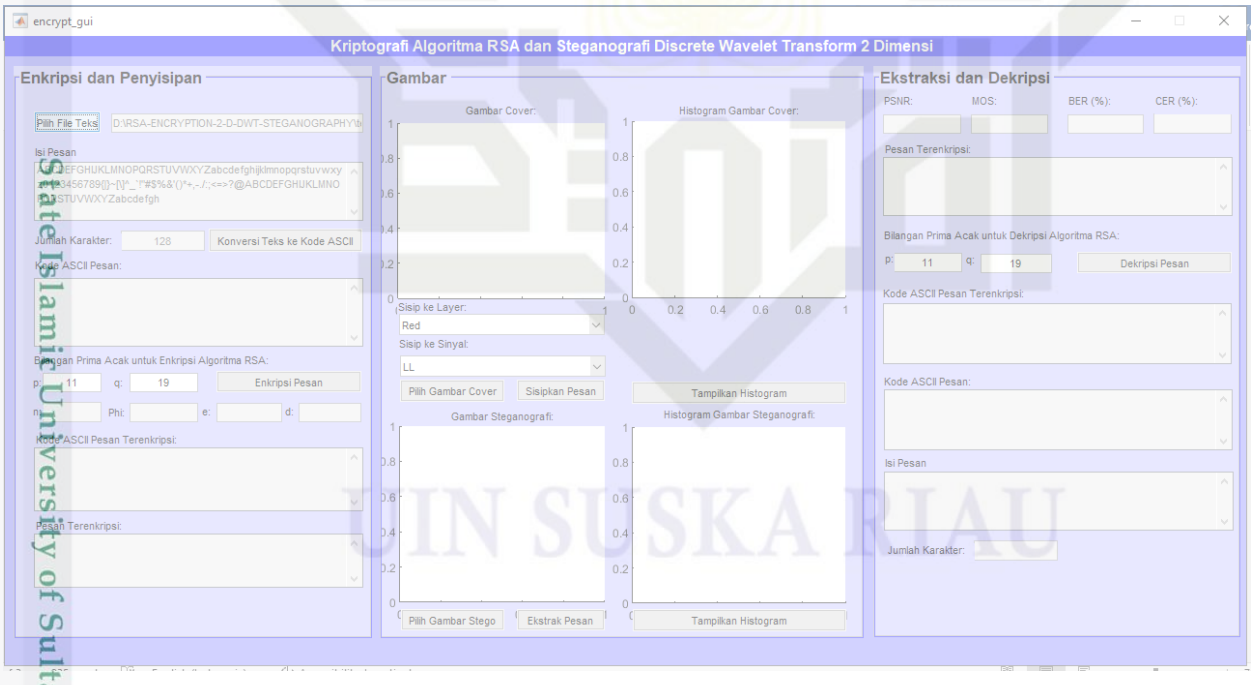
2. GUI SETELAH DI *RUN*



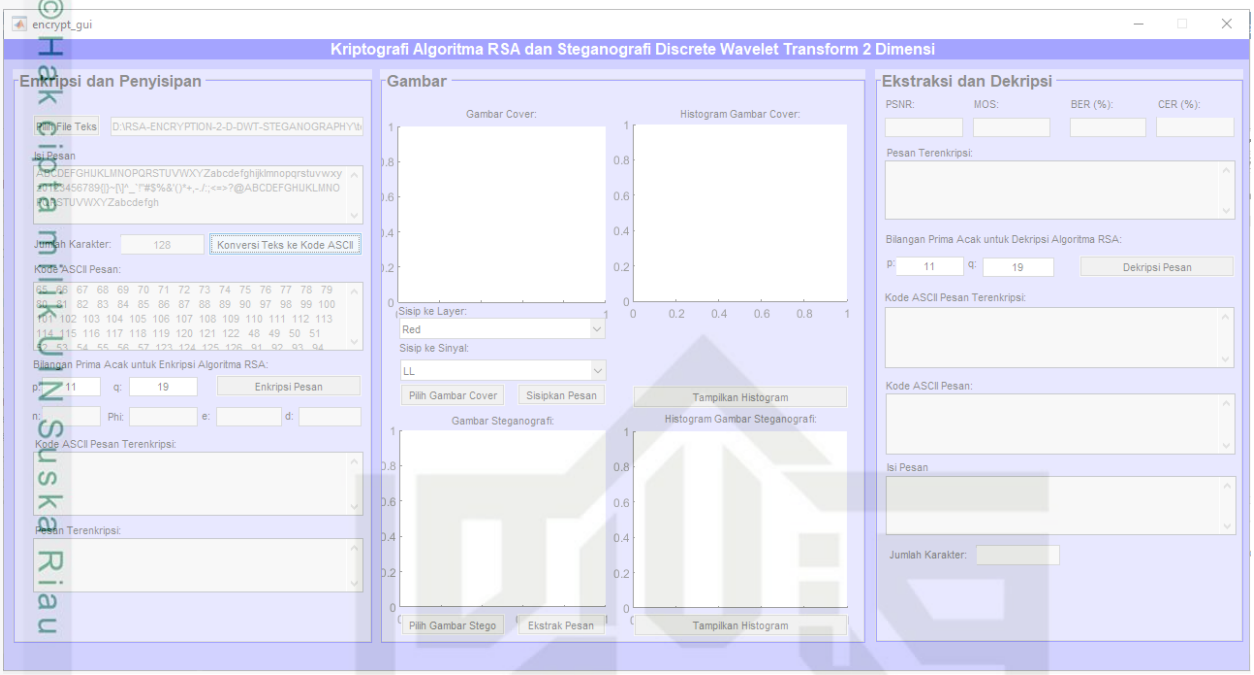
3. PILIH FILE TEKS



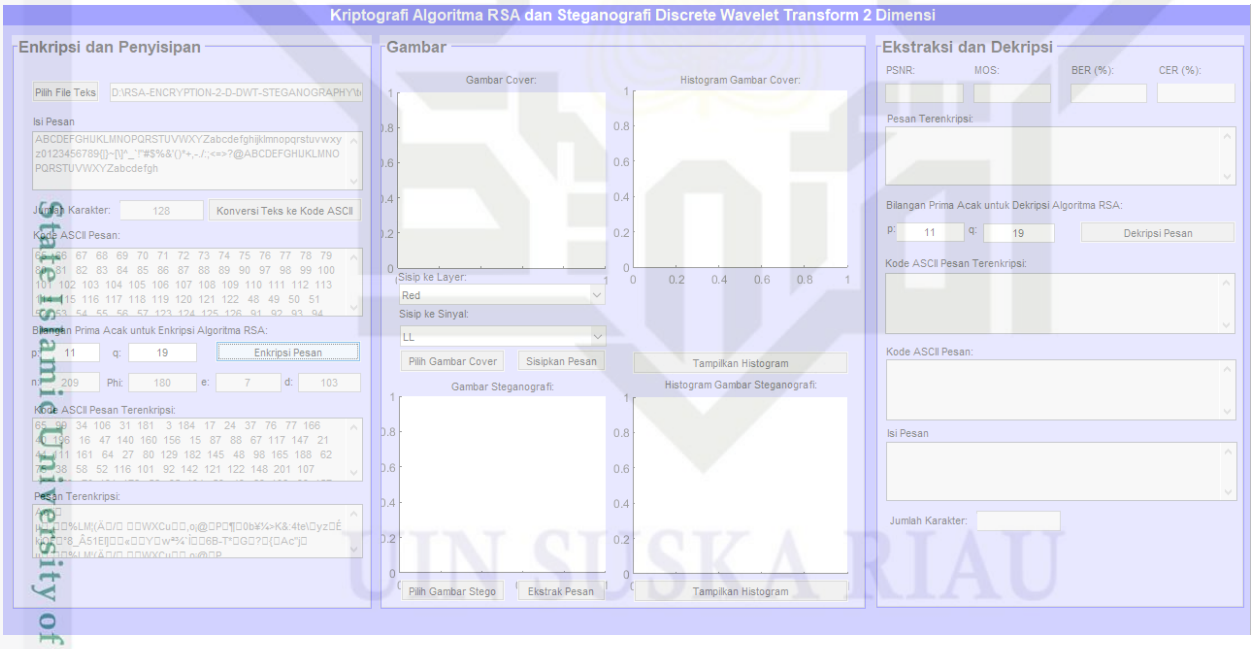
4. TAMPIL ISI PESAN



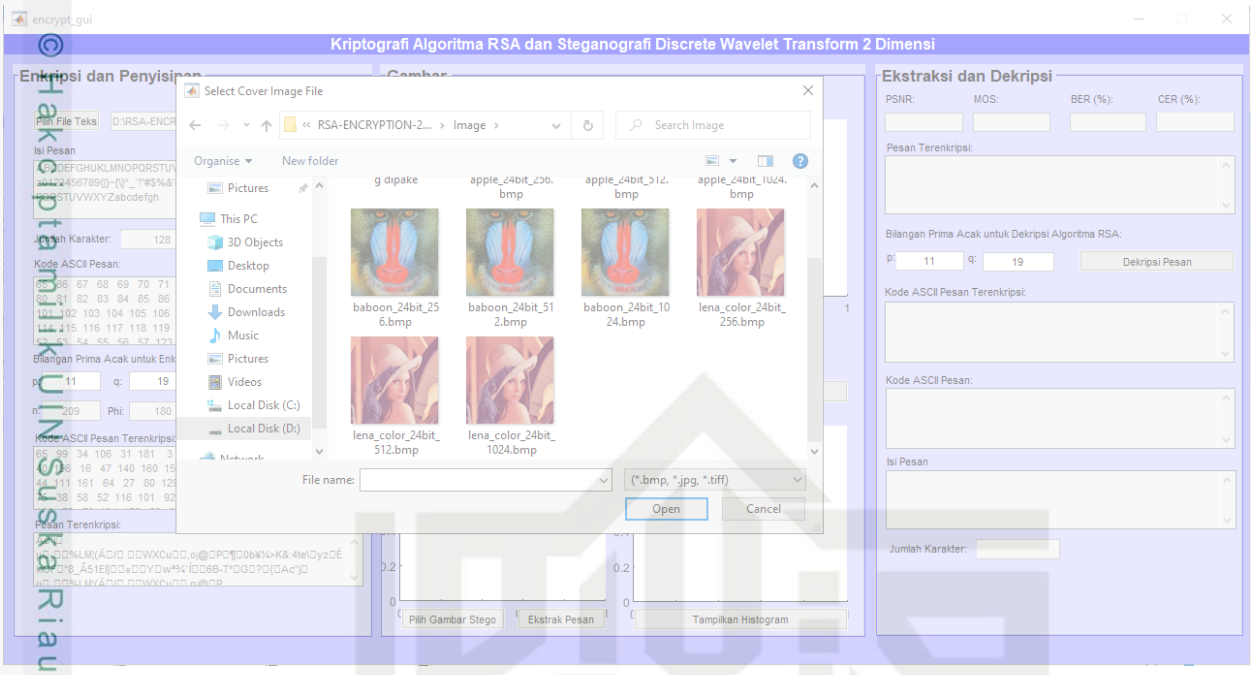
5. KONVERSI ASCII



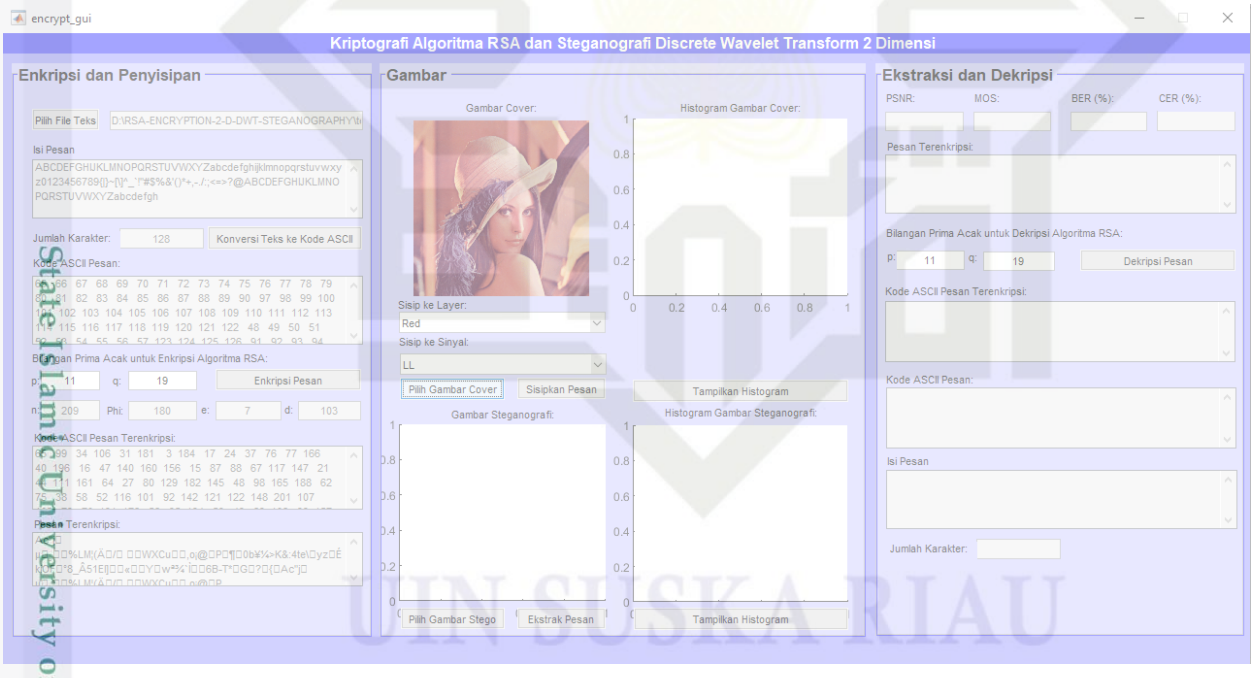
6. ENKRIPSI PESAN



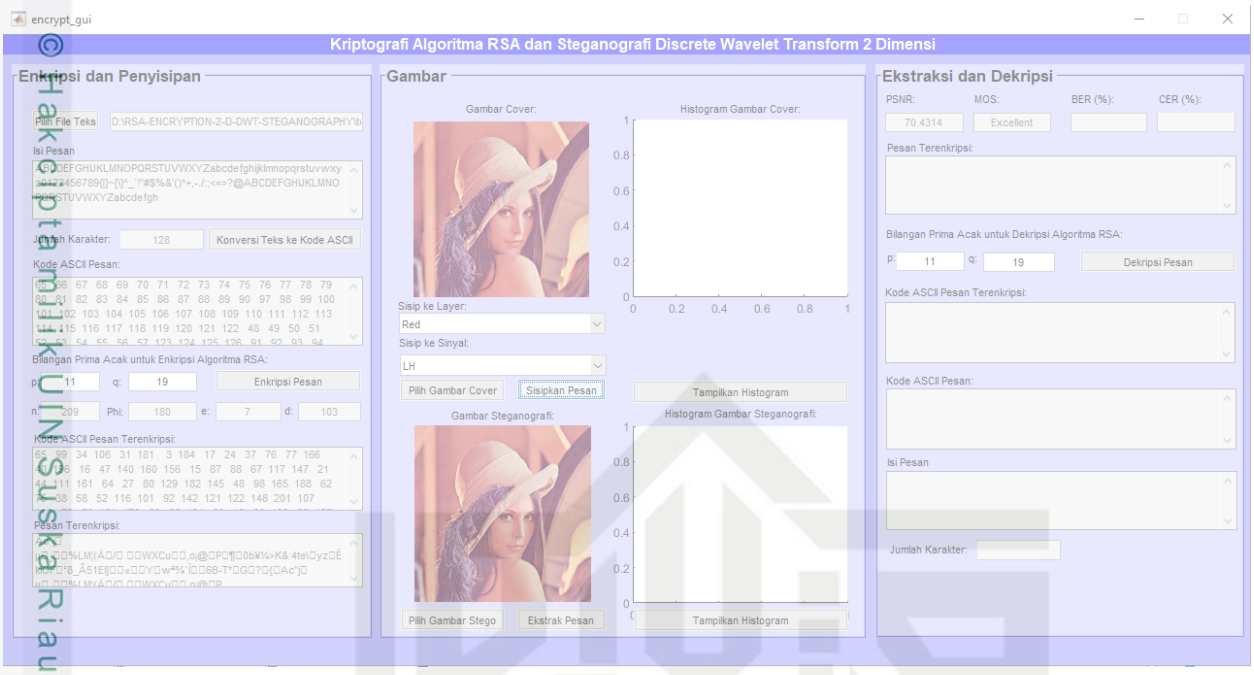
7. PILIH GAMBAR COVER



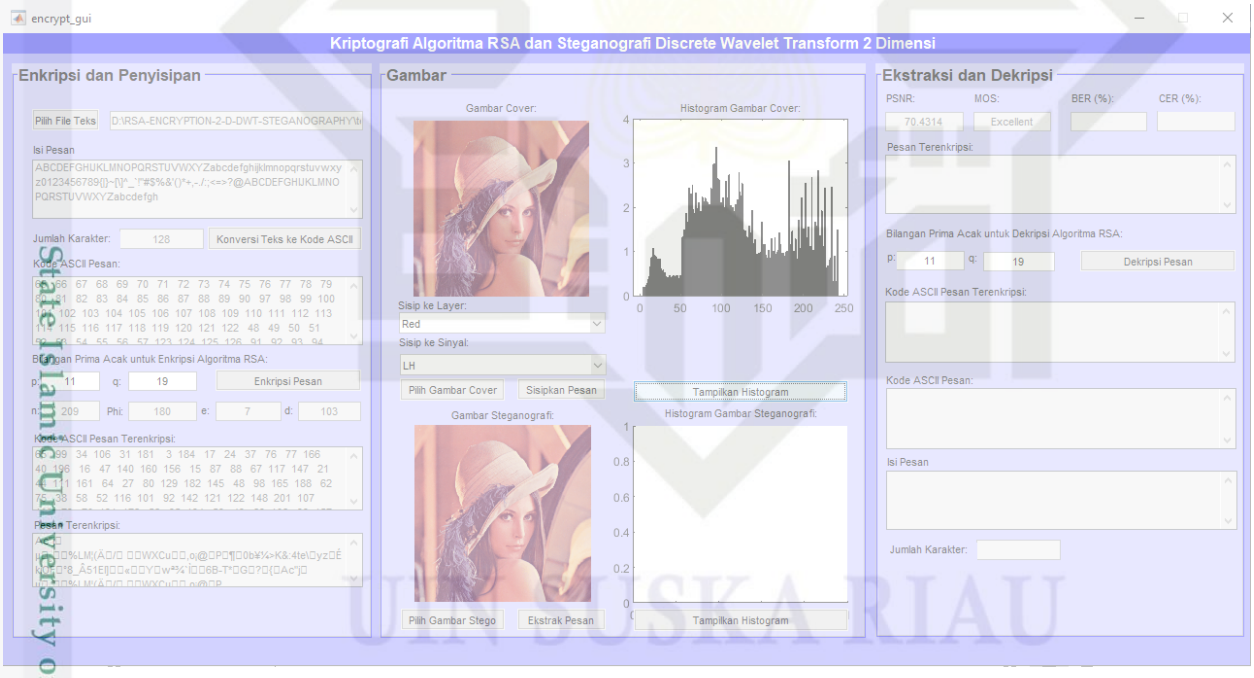
8. TAMPIL GAMBAR COVER



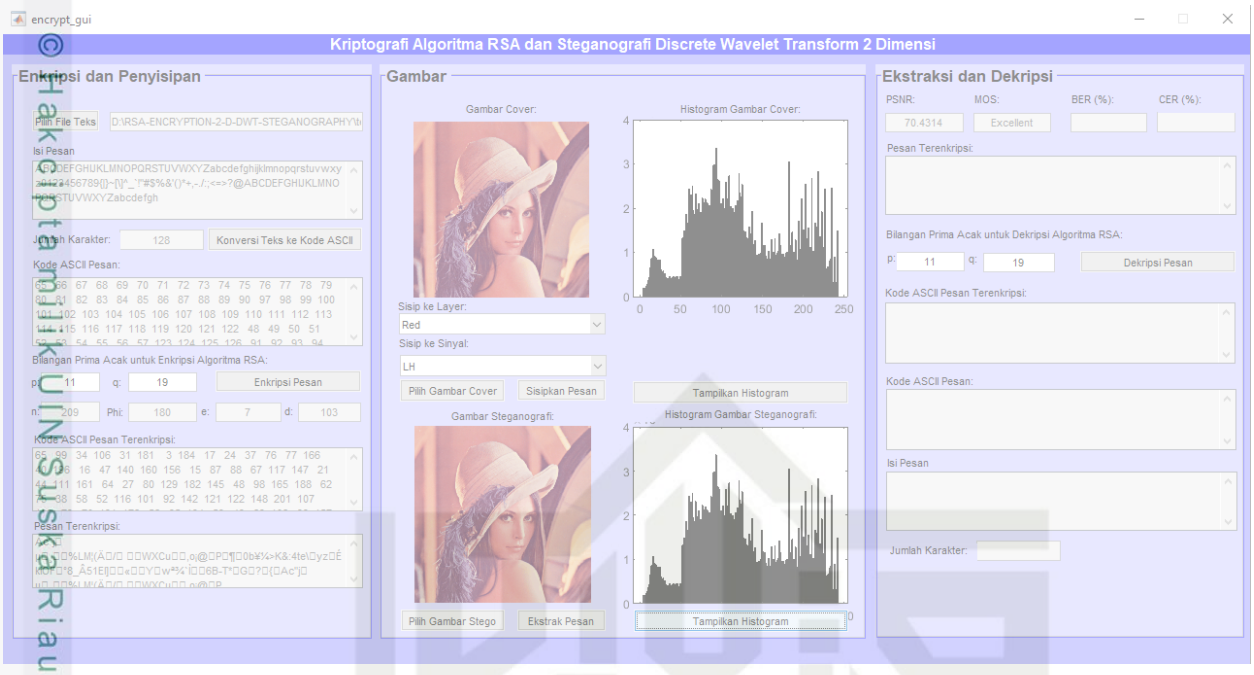
9. SISIP PESAN



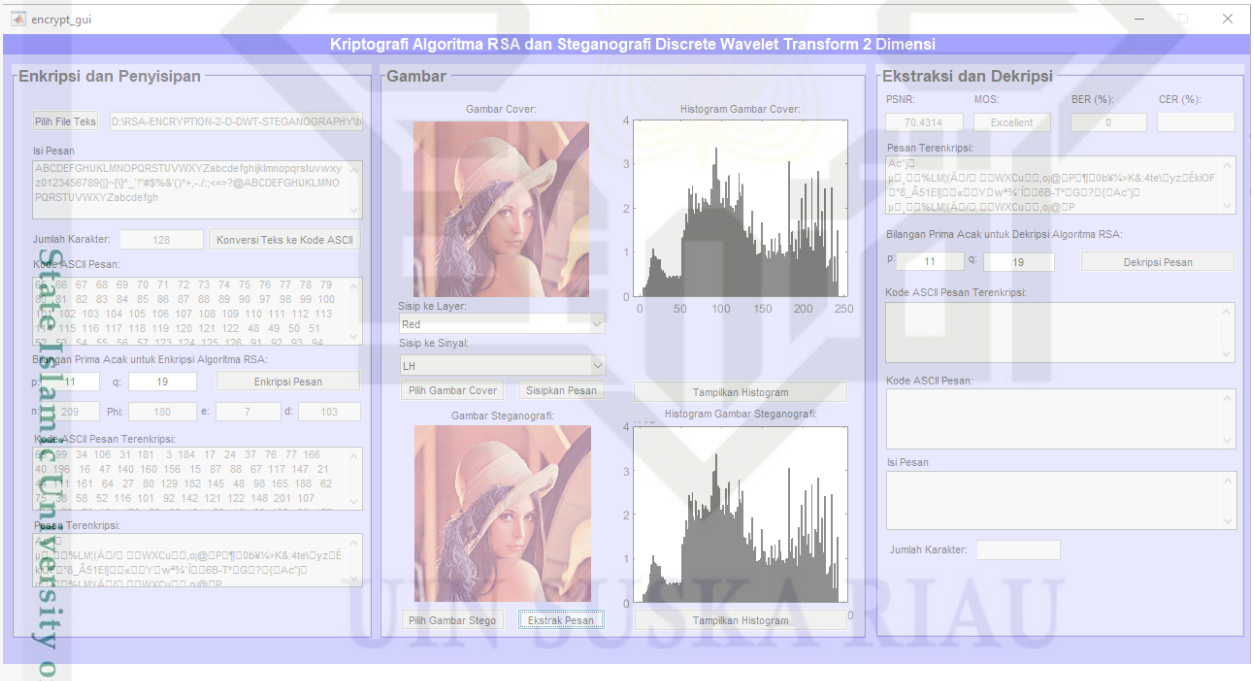
10. HISTOGRAM COVER



11. HISTOGRAM STEGO

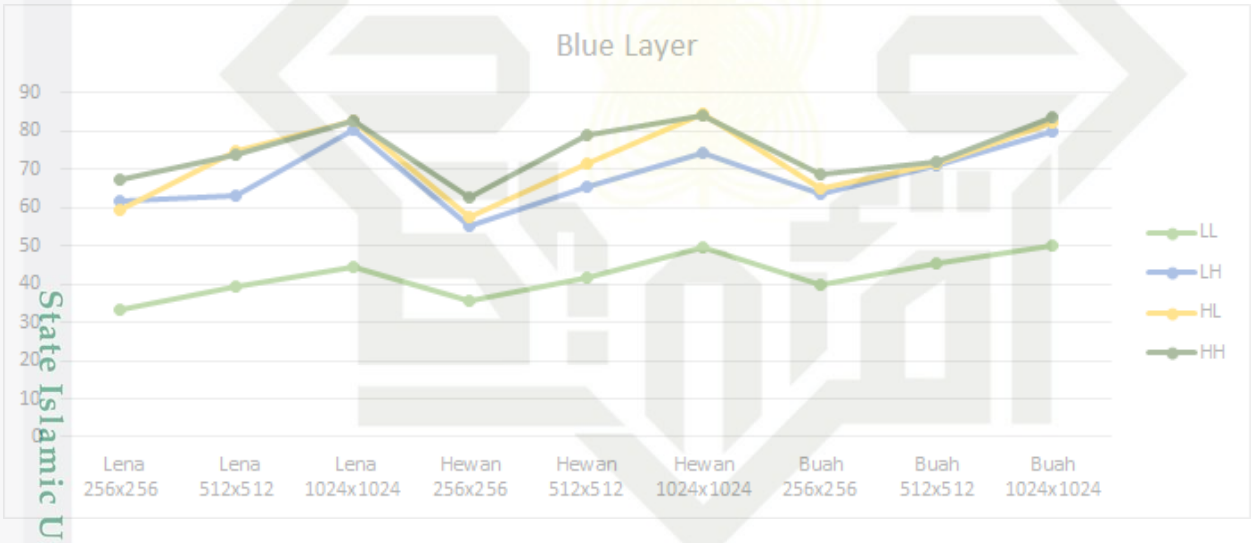


12. EKSTRAK PESAN

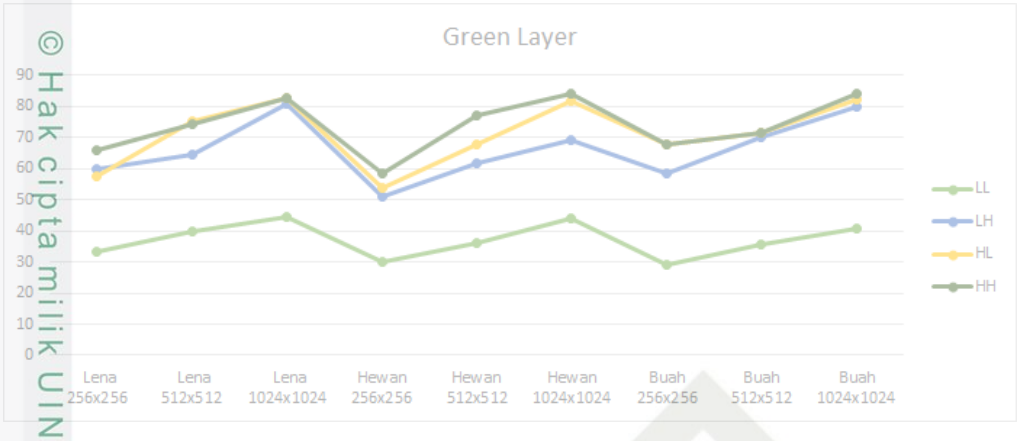


13. DESKRIPSI PESAN

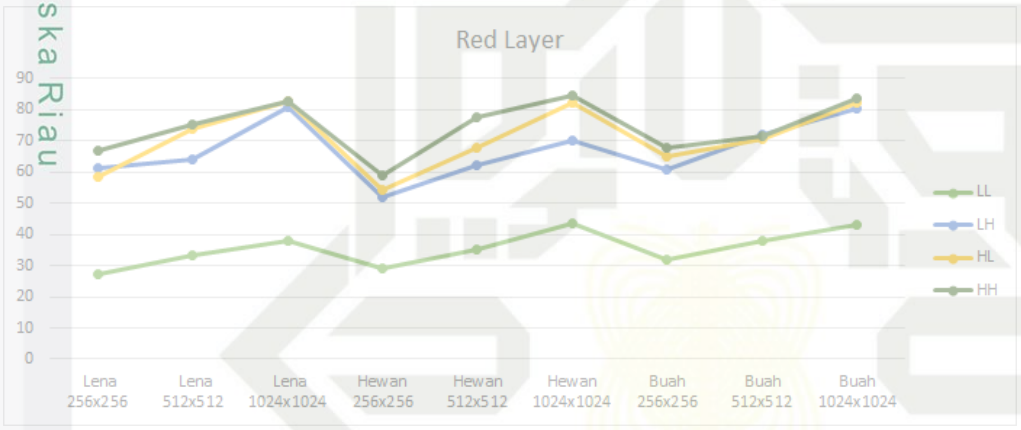
14. PENGUJIAN BLUE LAYER



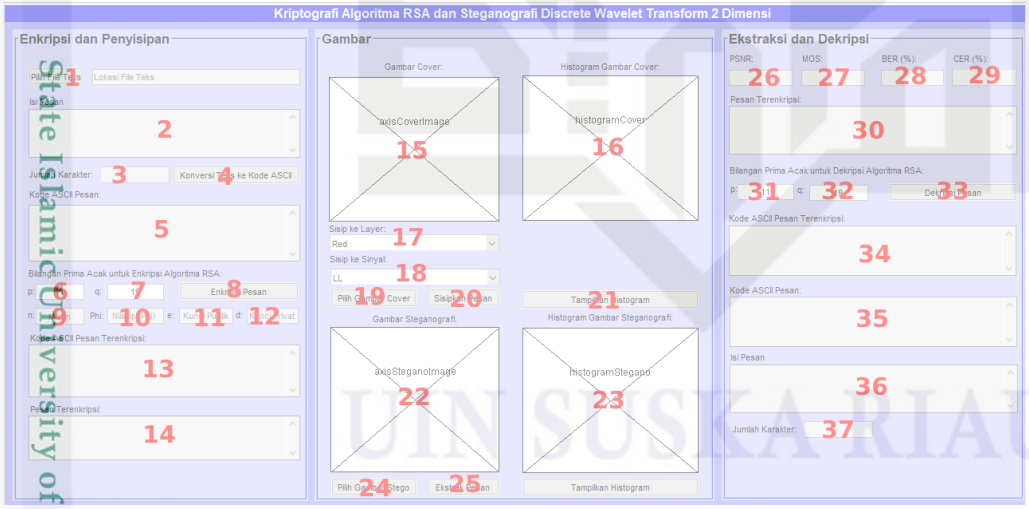
15. PENGUJIAN GREEN LAYER



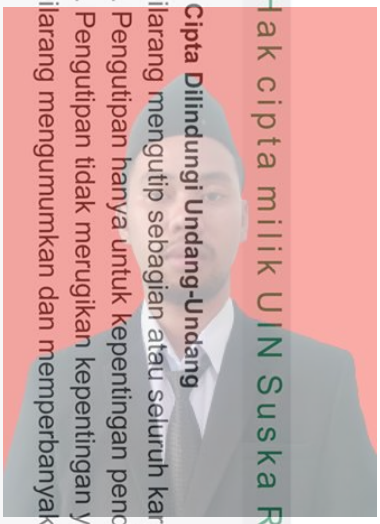
16. PENGUJIAN RED LAYER



17. RANCANGAN GUI



DAFTAR RIWAYAT HIDUP



Ade ibrahim lahir di pekanbaru pada tanggal 14 September 1997. Penulis merupakan anak kedua dari pasangan suami istri Bpk. Drs. Takim Napitupulu dan Asmurniati yang beralamat di Jalan Bandeng GG buntu No. 48 RT. 01 RW. 07, Kelurahan Tangkerang Tengah, Kecamatan Marpoyan Damai, Kota Pekanbaru, Provinsi Riau. Penulis menyelesaikan pendidikan di SDI As – Shofa Kota Pekanbaru tahun 2003 s/d 2008, dan melanjutkan pendidikan di SMP Negeri 13 Kota Pekanbaru pada tahun 2008 s/d 2011, kemudian melanjutkan pendidikan di SMK Negeri 2 Pekanbaru pada tahun 2011 s/d 2014. Setelah menyelesaikan pendidikan di SMK, penulis melanjutkan pendidikan di Program Studi Teknik Elektro Konsentrasi Teknik Komputer Fakultas Sains dan Teknologi UIN Suska Riau pada tahun 2014.

Akhir kata penulis mengucapkan rasa syukur yang sebesar-besarnya kepada **Allah Subhanahu wa ta'ala** atas terselesaikannya tugas akhir yang berjudul "**Menggabungkan Teknik Steganografi *Discrete Wavelet Transform* Dua dimensi (2-D) Dan Algoritma Kriptografi Rsa Pada Perancangan dan Analisi Keamanan Pesan**".

Nomor Handphone : 0822 8697 0066
E-mail : ade.ibrahim@students.uin-suska.ac.id