EMBEDDING FILE AUDIO TERENKRIPSI *INTERNATIONAL DATA ENCRYPTION ALGORITHM* (IDEA) PADA CITRA DENGAN METODE *DISCRETE WAVELET TRANSFORM* (DWT)

SKRIPSI

Oleh

Mohammad Iqbal Maulana
NIM 141810101046

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2018

EMBEDDING FILE AUDIO TERENKRIPSI *INTERNATIONAL DATA ENCRYPTION ALGORITHM* (IDEA) PADA CITRA DENGAN METODE *DISCRETE WAVELET TRANSFORM* (DWT)

SKRIPSI

diajukan guna memenuhi tugas akhir dan memenuhi salah satu syarat untuk
meyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh

Mohammad Iqbal Maulana
NIM 141810101046

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2018

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Musfiroh dan Ayahanda Asmai tercinta, yang telah membesarkan, mendidik, mendoakan, memotivasi dengan penuh kasih sayang dan perhatian yang tak pernah putus untuk putranya;
2. Adek Oktavia Khoirunnisa tersayang, yang telah mendoakan dan memberikan semangat dalam suka dan duka;
3. Guru-guru sejak taman kanak-kanak sampai dengan perguruan tinggi;
4. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember

MOTTO

"A person who reminds you to "Fear Allah" is your true companion worth more than anything and everything this world can possibly offer"

(Abu Maryam)

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

nama : Mohammad Iqbal Maulana

NIM : 141810101046

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul "*Embedding* File Audio Terenkripsi *International Data Encryption Algorithm* (IDEA) pada Citra dengan *Metode Discrete Wavelet Transform* (DWT)" adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya dan belum pernah diajukan pada institusi manapun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, 31 Januari 2018
Yang menyatakan,

Mohammad Iqbal Maulana
141810101046

SKRIPSI

EMBEDDING FILE AUDIO TERENKRIPSI *INTERNATIONAL DATA ENCRYPTION ALGORITHM* (IDEA) PADA CITRA DENGAN METODE *DISCRETE WAVELET TRANSFORM* (DWT)

Oleh

Mohammad Iqbal Maulana
NIM 141810101046

Pembimbing

Dosen Pembimbing Utama : Abduh Riski, S.Si., M.Si.

Dosen Pembimbing Anggota : Ahmad Kamsyakawuni, S.Si., M.Kom.

PENGESAHAN

Skripsi berjudul "*Embedding* File Audio Terenkripsi *International Data Encryption Algorithm* (IDEA) pada Citra dengan *Metode Discrete Wavelet Transform* (DWT)" telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Jember

Tim Penguji:

Ketua,

Abduh Riski, S.Si., M.Si.
NIP. 199004062015041001

Anggota I,

Ahmad Kamsyakawuni, S.S.i., M.Kom.
NIP. 197211291998021001

Anggota II,

Kusbudiono, S.Si., M.Si.
NIP. 197704302005011001

Anggota III,

Kosala Dwidja Purnomo, S.Si., M.Si.
NIP. 196908281998021001

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP. 196102041987111001

RINGKASAN

***Embedding* File Audio Terenkripsi *International Data Encryption Algorithm* (IDEA) pada Citra dengan Metode *Discrete Wavelet Transform* (DWT)**; Mohammad Iqbal Maulana, 141810101046; 2018: 61 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Perkembangan teknologi khususnya dalam bidang komunikasi memberikan dampak yang signifikan pada kebiasaan manusia. Perubahan terjadi dapat dilihat dari semakin mudahnya seseorang dalam mengiriman pesan atau informasi. Pengiriman pesan pada era sekarang, dapat dilakukan melalui media digital. Pengiriman pesan melalui media digital dipilih karena kemudahan pengiriman hingga efisiensi waktu pengiriman yang sangat cepat, salah satu jenis pesan data yang sering digunakan untuk menyimpan dan mengirim pesan berupa file audio. Namun perkembangan teknologi juga berdampak negatif, salah satunya yaitu penyalahgunaan pesan yang bersifat rahasia oleh pihak yang tidak berwenang.

Pengamanan pesan atau informasi dapat dilakukan untuk mengurangi penyalahgunaan pesan oleh pihak yang tidak berwenang. Pengamanan pesan dapat dilakukan dengan beberapa teknik seperti kriptografi dan steganografi. Kriptografi adalah ilmu untuk mengamankan suatu informasi dengan cara mengenkripsi data sehingga *attacker* (pihak tidak berwenang) tidak dapat memahami informasi yang ada di dalamnya. Pesan rahasia berupa file audio akan dienkripsi menggunakan IDEA (*International Data Encryption Algorithm*). Penggunaan IDEA pada proses kriptografi karena IDEA memiliki keamanan yang kuat untuk mengamankan data. Hasil dari proses enkripsi (*cipher audio*) masih memiliki kekurangan, yaitu data yang dihasilkan sangat tidak jelas, yang mana dapat menimbulkan kecurigaan bagi *attacker* yang mendapatkannya. Oleh karena itu audio terenkripsi dapat ditingkatkan keamanannya menggunakan metode steganografi, metode ini dilakukan dengan menyisipkan audio terenkripsi ke dalam sebuah citra sehingga tidak diketahui keberadaanya. Oleh karena itu, steganografi dapat digunakan sebagai pengamanan lanjutan dari proses enkripsi. Metode yang digunakan dalam steganografi yaitu DWT (*Discrete Wavelet Transform*), dimana DWT menyediakan

keamanan yang maksimal bagi pesan rahasia. Sehingga hasil dari pengamanan data berupa citra yang berisi data audio terenkripsi. Data yang digunakan dalam penelitian ini adalah audio berformat $*$.*wav* dengan teks sebagai kuncinya. Serta citra sebagai media penyisipannya. Citra yang digunakan dalam penilitian ini berformat $*$.png, $*$.tif, $*$.bmp dan $*$.jpg.

Analisis keamanan yang digunakan untuk mengetahui tingkat keamanan dari algoritma yang diajukan yaitu *Signal Noise to Ratio* (SNR), *Peak Signal Noise to Ratio* (PSNR) dan analisis sensitivitas kunci. Pengamanan data audio menggunakan algortima yang diajukan mendapatkan hasil yang baik, dimana proses enkripsi dan penyisipan ke dalam citra dapat berjalan dengan baik. Nilai PSNR dari proses penyisipan atau *embedding* dalam suatu citra menunjukkan nilai lebih dari 40dB sehingga citra tidak mengalami distorsi yang signifikan. Nilai SNR_{dB} dari data audio hasil proses pengungkapan data menunjukkan nilai lebih dari 25dB sehingga audio dari proses pengungkapan data mengalami distorsi (penambahan *noise*) namun tidak menyebabkan hilangnya data rahasia di dalamnya. Selain itu tingkat sensitivitas kunci algoritma yang diajukan tinggi, sehingga dari ketiga metode analisis keamanan yang diajukan, menunjukkan bahwa tingkat keamanan dari serangan *attacker* atau orang yang berusaha memecahkan pesan rahasia tersebut sangat baik.

PRAKATA

Puji syukur penulis panjatkan kehadirat Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul "*Embedding* File Audio Terenkripsi *International Data Encryption Algorithm* (IDEA) pada Citra dengan *Metode Discrete Wavelet Transform* (DWT)". Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak terlepas dari perhatian, bimbingan, motivasi, dan petunjuk dari beberapa pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Bapak Abduh Riski, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Bapak Ahmad Kamsyakawuni, S.Si., M.Kom. selaku Dosen Pembimbing Anggota yang dengan penuh kesabaran membimbing, mengarahkan, memberikan saran dan petunjuk dalam penyusunan skripsi ini;
2. Bapak Kusbudiono, S.Si., M.Si. dan Bapak Kosala Dwidja Purnomo, S.Si., M.Si. selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun dalam penyusunan skripsi;
3. Seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan skripsi ini;
4. Bapak Ahmad Kamsyakawuni, S.Si., M.Kom. selaku Dosen Pembimbing Akademik;
5. Ibu, Bapak, adikku serta seluruh keluarga dirumah yang telah memberikan doa dan motivasi;
6. Teman-teman kontrakan yang telah tinggal satu rumah selama menjadi mahasiswa.
7. Teman-teman Matematika 2014 yang telah menemani selama menjadi mahasiswa dan berbagi canda tawa;
8. Serta semua pihak yang tidak dapat disebutkan satu-persatu

Akhir kata, penulis berharap semoga skripsi ini bermanfaat dan bisa dikembangkan lagi agar lebih sempurna.

Jember, Januari 2018 Penulis

DAFTAR ISI

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

DAFTAR LAMPIRAN

Halaman

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Komunikasi merupakan hal yang selalu dilakukan setiap saat. Seiring dengan berkembangnya teknologi dalam dunia komunikasi, membuat semakin banyak pesan ataupun informasi yang disampaikan maupun dikirimkan melalui media digital. Penyampaian pesan ataupun informasi melalui media digital dipilih karena efisiensi waktu pengiriman yang sangat cepat dan penggunaanya yang semakin mudah. Permasalahan muncul ketika seseorang ingin mengirimkan pesan ataupun informasi yang bersifat rahasia, namun keamanan pada media digital masih minim. Permasalahan tersebut dapat diatasi dengan cara meningkatkan pengamanan pesan maupun informasinya.

Pengamanan pesan data atau informasi dapat menggunakan teknik kriptografi dan teknik steganografi. Teknik kriptografi dilakukan dengan cara mengenkripsi pesan data yang dapat berupa audio, teks, citra ataupun video. Enkripsi dilakukan pada pesan, sebelum pesan tersebut dikirim, sehingga *attacker* (orang yang tidak berhak atas pesan) tidak dapat memahami pesan yang dikirimkan tersebut meskipun pesan berhasil diakses/diperoleh. Salah satu jenis pesan data yang sering digunakan oleh pengguna sebagai media penyimpan yaitu pesan yang berjenis audio.

Pengiriman pesan (audio) digital juga tidak luput dari serangan *attacker* yang akan mengambil atau merusak pesan. Oleh karena itu pengamanan dalam pengiriman pesan dapat dilakukan dengan cara mengenkripsi data atau pesan menggunakan (*International Data Encryption Algorithm*) IDEA. Menurut Hanan (2013), *cipher text* yang dihasilkan menggunakan enkripsi IDEA, akan berbeda untuk setiap *plain text* yang sama dengan kunci yang sedikit berbeda, sehingga dapat dikatakan tingkat sensitivitas kuncinya tinggi. Ini sangat efektif jika digunakan untuk pengamanan suatu pesan maupun data.

Pada penelitian ini, data audio yang digunakan menggunakan format $*$.wav yang nantinya akan dienkripsi menggunakan algoritma IDEA. Algoritma IDEA menggunakan beberapa operasi dasar, seperti operasi logika XOR (*Exclusive-OR*),

operasi perkalian mod $2^{16}+1$ (*multiplication modulo* $2^{16}+1$) dan operasi penambahan mod 2^{16} (*addition modulo* 2^{16}). Metode ini terdiri dari 8 putaran (*round*) dengan algoritma yang sama dan 1 putaran *Transformasi Output* (TO) dengan menggunakan 64 bit *plain audio* dengan panjang kunci 128 bit.

Hasil dari proses enkripsi (*cipher audio*) akan menghasilkan audio baru yang tidak jelas suaranya. Hal ini akan menimbulkan kecurigaan bagi pihak lain yang akan merusak atau menggunakan pesan rahasia tanpa ijin. Oleh karena itu *cipher audio* akan disembunyikan pada citra menggunakan teknik steganografi. Steganografi pada pesan terenkripsi dilakukan agar pesan yang bersifat rahasia akan aman sehingga saat *attacker* mengetahui apabila ada suatu pesan rahasia dalam suatu citra, maka *attacker* akan mendapatkan pesan rahasia yang telah terenkripsi. Oleh karena itu *attacker* harus memecahkan lagi data terenkripsi agar data rahasia dapat dibaca maupun dipecahkan seutuhnya.

Pemanfaatan teknik steganografi diharapkan agar proses berkomunikasi dan berkirim pesan dapat dilakukan dengan lebih aman kapanpun dan dimanapun, karena citra hasil steganografi tidak akan terlihat bahwa di dalamnya ada pesan tersembunyi. Metode yang digunakan dalam teknik steganografi adalah metode DWT (*Discrete Wavelet Transform*) dengan menyisipkan data audio terenkripsi ke dalam media digital (citra). Goel (2013), menyimpulkan bahwa DWT adalah metode yang paling tinggi tingkat ketahananya dibandingkan dengan LSB dan (*Discrete Cosine Transform*) DCT, dimana citra tidak akan rusak saat proses ekstraksi pesan rahasia di dalamnya dan DWT juga menyediakan keamanan yang maksimal.

Pada penelitian ini penulis akan mengenkripsi pesan berjenis audio berformat ∗.wav dengan menggunakan algoritma IDEA dan hasil dari proses enkripsi akan dilanjutkan dengan teknik steganografi pada sebuah citra. Citra sebagai tempat penyisipan akan diproses terlebih dahulu menggunakan DWT dan selanjutnya pesan audio hasil enkripsi akan disisipkan pada citra hasil DWT menggunakan metode LSB.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah, rumusan masalah dalam penulisan proposal skripsi ini adalah:

a. Bagaimana proses enkripsi dan dekripsi pesan (data audio) berformat ∗.wav menggunakan algoritma IDEA?
b. Bagaimana proses (penyisipan) *embedding* dan ekstraksi pesan (data audio) terenkripsi menggunakan metode DWT?
c. Bagaimana analisis keamanan dari metode yang diajukan?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

a. Enkripsi dan dekripsi pesan audio menggunakan algoritma IDEA.
b. *Embedding* dan ekstraksi pesan audio yang telah terenkripsi menggunakan metode DWT pada citra digital.
c. Menganalisis keamanan dari metode yang diajukan.

1.4 Manfaat Penelitian

Adapun manfaat yang diharapkan pada penelitian ini adalah:

a. Mengetahui proses enkripsi dan dekripsi pesan audio menggunakan algoritma IDEA.
b. Mengetahui proses *embedding* dan ekstraksi pesan audio terenkripsi menggunakan DWT.
c. Mampu menganalisis keamanan dari metode yang diajukan.

BAB 2. TINJAUAN PUSTAKA

2.1 Metode Pengamanan Data

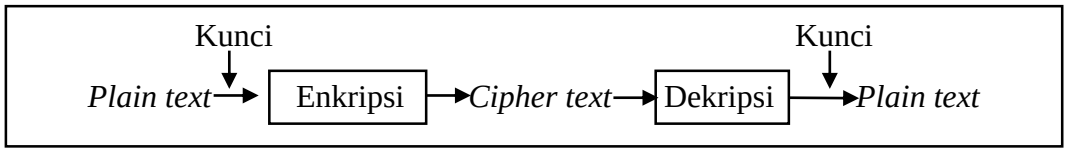
Metode dalam pengamanan data yang digunakan yaitu kriptografi dan juga steganografi. Tujuan pengamanan pesan atau data yaitu agar suatu pesan atau data yang ingin dikirim ataupun disimpan akan terjaga kerahasiaanya, serta akan aman terhadap *attacker* yang bertujuan merusak ataupun mendapatkan pesan (data).

2.1.1 Kriptografi

Kriptografi berasal dari bahasa Yunani yakni kriptos yang artinya tersembunyi dan graphia yang artinya sesuatu yang tertulis, sehingga kriptografi dapat disebut sebagai sesuatu yang tertulis secara rahasia. Menurut (Munir, 2006) kriptografi merupakan suatu bidang ilmu yang mempelajari tentang bagaimana merahasiakan suatu informasi penting ke dalam suatu bentuk yang tidak dapat dibaca oleh siapapun serta mengembalikannya kembali menjadi informasi semula dengan menggunakan berbagai macam teknik yang telah ada, sehingga informasi tersebut tidak dapat diketahui oleh pihak manapun yang bukan pemilik atau tidak berkepentingan. Sisi lain dari kriptografi ialah c*ryptanalysis* yang merupakan *study* tentang bagaimana memecahkan mekanisme kriptografi.

Beberapa istilah penting dalam kriptografi adalah *plain text*, *cipher text*, enkripsi, dekripsi, kunci (*key*), dan algoritma. *Plain text* merupakan informasi awal yang bisa dibaca. *Cipher text* merupakan informasi hasil pesan *plain text* yang sudah disandikan. Enkripsi adalah teknik untuk menjadikan data *plain text* agar tidak dapat dibaca. Dekripsi adalah teknik untuk mengembalikan *cipher text* menjadi *plain text* kembali. Kunci (*key*) berfungsi untuk mengatur dan menjalankan suatu algoritma. Sedangkan, algoritma adalah suatu metode untuk melalukan proses enkripsi dan dekripsi tersebut

Istilah-istilah tersebut dijalankan secara sistematis dengan diagram alir seperti berikut:



Gambar 2.1 Aliran proses enkripsi dan dekripsi

2.1.2 Steganografi

Selain kriptografi, metode dalam pengamanan pesan atau data dapat menggunakan metode steganografi. Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Kata steganografi berasal dari bahasa yunani yang berarti "tulisan tersembunyi" (*covered writing*). Steganografi dapat dipandang sebagai kelanjutan kriptografi. Jika pada kriptografi, data yang telah diproses enkripsi (*cipher text*) tetap tersedia atau dengan kata lain dapat terlihat, maka dengan steganografi, *cipher text* dapat disembunyikan sehingga pihak ketiga tidak mengetahui keberadaannya.

Secara garis besar metode steganografi terdiri dari 2 bagian utama, yaitu proses penyembunyian data (*hidden message*) atau biasa disebut penyisipan data *(embedding message)* dan proses pengembalian data kebentuk semula (*reveal message*) atau juga dapat disebut *extraction* (Munir, 2004).

Steganografi dapat diterapkan pada media digital seperti teks, citra, audio dan video. Terdapat beberapa metode steganografi untuk citra digital yang sudah ada, seperti metode steganografi yang bekerja pada domain spasial misalnya metode LSB (*Least Significant Bit*) dan metode yang mengalami transformasi terlebih dahulu, misalnya ke domain frekuensi seperti DCT (*Discrete Cosine Transform), Wavelet Transform, Spread Spectrum,* dan sebagainya (Pranoto, 2011).

2.2 Algortima IDEA

IDEA merupakan salah satu algoritma kriptografi yang beroperasi pada blok *plain text* 64 bit dengan panjang kuncinya 128 bit dan menggunakan operasi XOR, penambahan modulo 2^{16} dan juga perkalian modulo $2^{16}+1$.

Algoritma IDEA *(International Data Encryption Algorithm*) menggunakan perkalian modulo $2^{16}+1$. Perkalian modulo n tidak memiliki invers, jika angka yang

dikalikan tidak relatif prima terhadap n. Angka 65537 (2^{16}+1) adalah sebuah bilangan prima. Oleh karena itu, operasi perkalian modulo (2^{16}+1) pada algoritma IDEA pasti memiliki invers. Dalam algoritma IDEA untuk operasi perkalian, bilangan 16 bit yang terdiri dari nol semua dianggap sebagai bilangan 65536, sedangkan bilangan lainnya tetap sesuai dengan bilangan tak bertanda yang diwakilinya. Algoritma IDEA ini dapat dibagi menjadi 3 bagian besar, yaitu pembangkit kunci, enkripsi dan dekripsi.

2.2.1 Landasan Matematika Kriptografi

a. Sistem Basis Bilangan

Sistem basis pada bilangan merupakan suatu bilangan yang mewakili besaran dari suatu item fisik yang memiliki basis. Sistem basis bilangan yang sering digunakan adalah bilangan desimal dan juga biner. Bilangan desimal merupakan suatu sistem bilangan yang memiliki 10 basis, yakni 0, 1, 2, 3, 4, 5, 6, 7, 8, dan 9. Bilangan biner merupakan sistem bilangan yang memiliki 2 basis, yakni 0 dan 1. Suatu sistem basis bilangan dapat dikonversikan kedalam sistem basis bilangan yang lainya. Beberapa teknik untuk mengkonversi suatu sistem basis bilangan menjadi bentuk sistem basis bilangan yang lain adalah sebagai berikut:

1. Konversi bilangan desimal ke bilangan biner.

   Proses konversi bilangan desimal menjadi bentuk bilangan biner yaitu dengan membagi bilangan desimal dengan 2 secara terus menerus sampai tersisa 0, nilai dari pembagian terakhir dan sisa dari setiap pembagian tersebut merupakan nilai dari bilangan biner yang diinginkan.

   Contoh:

   $21_{(10)} = 10101_{(2)}$

   21 / 2 = 10 sisa 1

   10 / 2 = 5 sisa 0

   5 / 2 = 2 sisa 1

   2 / 2 = 1 sisa 0

   Maka hasil dari konversi $21_{(10)}$ adalah $10101_{(2)}$

2. Konversi bilangan biner ke bilangan desimal

   Proses konversi bilangan biner menjadi bentuk bilangan desimal yaitu dengan memisah setiap digit kemudian mengalikan setiap digitnya dengan 2^n, dimana n merupakan posisi dari basis. Seluruh nilai dari proses tersebut kemudian dijumlahkan untuk mendapatkan nilai dari bilangan desimal yang diinginkan.

   Contoh:

   $10110_{(2)} = 22_{(10)}$

   $(1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

   $= 16 + 0 + 4 + 2 + 0$

   $= 22$

   Maka hasil dari konversi $10110_{(2)}$ adalah $22_{(10)}$.

b. Operasi Modulo

Operasi modulo merupakan operasi matematika yang banyak diimplementasikan pada metode kriptografi. Misalkan a adalah bilangan bulat dan m adalah bilangan bulat positif. Jika a dibagi dengan m maka terdapat dua buah bilangan bulat unik q hasil bagi *(quotient)* dan r sisa hasil *(remainder)*, sedemikian sehingga

$$a = mq + r \quad (2.1)$$

$$a \equiv r \pmod{m} \quad (2.2)$$

dengan $0 \leq r < m$. Sehingga Persamaan 2.2 dapat dibaca “a mod m”. Ada beberapa operasi modulo yang sering digunakan dalam kriptografi. Pada algortima IDEA operasi modulo yang digunakan yaitu operasi penjumlahan modulo, invers penjumlahan modulo, operasi perkalian modulo serta invers perkalian modulo.

1. Penjumlahan modulo

   Penjumlahan modulo merupakan proses operasi modulo pada 2 bilangan atau lebih yang sebelumnya telah melalui operasi penjumlahan. Setelah operasi penjumlahan, hasil dari operasi penjumlahan tersebut dilakukan

operasi modulo. Pada algortima IDEA menggunakan penjumlahan modulo 2^{16}. Sebagai contoh:

$$65530 + 10 = 65540$$
$$\equiv 4 \pmod{2^{16}}$$

2. Perkalian modulo

Seperti halnya penjumlahan modulo, perkalian modulo merupakan proses operasi modulo pada 2 bilangan atau lebih yang sebelumnya melalui operasi perkalian. Hasil dari operasi perkalian dilakukan operasi modulo. Pada algortima IDEA menggunakan perkalian modulo $2^{16} + 1$. Sebagai contoh:

$$32675 \times 4 = 131060$$
$$\equiv 65523 \bmod (2^{16} + 1)$$

3. Invers penjumlahan modulo

Invers penjumlahan modulo n dimana invers dari $a \in Z_n$ yaitu $-a \in Z_n$ sehingga $a + (-a) = 0 \,|0 \in Z_n$. Pada algortima IDEA menggunakan penjumlahan modulo 2^{16} . Sebagai contoh:

Misalkan $a = 32654$, invers penjumlahan dari a didapatkan dari perhitungan:

Invers penjumlahan $= 65536 - a$
$= 65536 - 32654$
$= 32882$

Maka invers penjumlahan modulo 2^{16} dari a adalah $(-a)$ dengan nilai 32882

4. Invers perkalian modulo

Bilangan $a \in Z_n$ mempunyai invers perkalian modulo n jika dan hanya jika *FPB(a,n)* = 1 atau biasa disebut a dan n relatif prima. Pada algoritma IDEA nilai n = 2^{16} +1 dimana $2^{16} + 1$ merupakan bilangan prima, sehingga untuk $\forall\, a \in Z_n$ selalu memiliki invers perkalian, maka

didapatkan $(a \times a^{-1})$ mod $(2^{16}+1) = 1$, dimana a^{-1} merupakan invers perkalian dari a. Misalkan nilai dari a adalah 16666 maka invers perkalian modulonya didapatkan yaitu:

$$16666 \times a^{-1} \equiv 1 \bmod (2^{16}+1)$$
$$a^{-1} = 26750$$

c. Operasi XOR

XOR (*Exclusive-OR)* yang dilambangkan dengan tanda "⊕" merupakan operasi yang *output*-nya akan bernilai 1 apabila *input*-nya berbeda, dan *output*-nya akan bernilai 0 apabila *input*-nya sama. Apabila nilai *A* dilakukan operasi XOR terhadap *B* sebanyak dua kali maka akan didapatkan nilai *A* kembali. Karena sifat istimewa yang dimiliki operasi XOR tersebut sehingga operasi XOR cenderung dipakai dalam proses enkripsi dan dekripsi yang memiliki algoritma yang sama. Adapun aturan XOR dapat dilihat pada tabel dibawah:

Tabel 2.1 *XOR* (*Exclusive-OR*)

A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

d. Permutasi (*Permutation*)

Permutasi merupakan suatu proses korespondensi dari satu ke banyak. Permutasi dalam kriptografi sering digunakan untuk memindahkan posisi sejumlah bit ke posisi yang telah ditentukan dalam tabel permutasi. Ada beberapa metode dalam kriptografi yang menggunakan permutasi pada awal maupun akhir dari proses enkripsi maupun dekripsi dan ada juga metode yang menggunakan permutasi untuk menghasilkan beberapa *subkey* yang diperlukan dalam proses enkripsi dan dekripsi.

Diberikan 1 baris bit sebagai berikut: 1110 0110 1101 1100, terhadap barisan bit tersebut akan dilakukan permutasi menggunakan tabel permutasi sebagai berikut:

Tabel 2.2 Baris bit sebelum permutasi

Bit ke-	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Bit	1	1	1	0	0	1	1	0	1	1	0	1	1	1	0	0

Tabel 2.3 Baris bit setelah permutasi

Bit Ke-	4	5	14	3	0	9	8	15	12	13	1	2	6	7	10	11
Bit	0	1	0	0	1	1	1	0	1	1	1	1	1	0	0	1

e. Pergeseran Bit (*Shift*)

Pergeseran bit (*Shift*) adalah operasi pergeseran terhadap suatu barisan bit sebanyak yang diinginkan. Bit kosong yang telah tergeser akan diberikan nilai bit "0" (nol). Operasi pergeseran terbagi menjadi dua macam yaitu:

1. Operasi geser kiri (*Shift Left*) yaitu operasi yang menggeser (*shift*) sejumlah bit ke kiri (*left*) dengan nilai bit "0" (nol). Operasi *shift left* dilambangkan dengan "<<".

   Contoh operasi *shift left*: 11000110 << 1: 10001100
   11000110 << 2: 00011000

2. Operasi geser kanan (*Shift Right*) yaitu operasi yang menggeser (*shift*) sejumlah bit ke kanan (*right*) dengan nilai bit "0" (nol). Operasi *shift right* dilambangkan dengan ">>".

   Contoh operasi *shift right*: 11000110 >> 1: 01100011
   11000110 >> 2: 00110001

f. Rotasi Bit (*Rotate*)

Rotasi bit (*Rotate*) adalah operasi perputaran terhadap suatu barisan bit sebanyak yang diinginkan. Bit yang tergeser akan dipindahkan ke sisi barisan

bit yang berlawanan dengan arah putaran bit. Operasi rotasi terbagi atas dua macam yaitu:

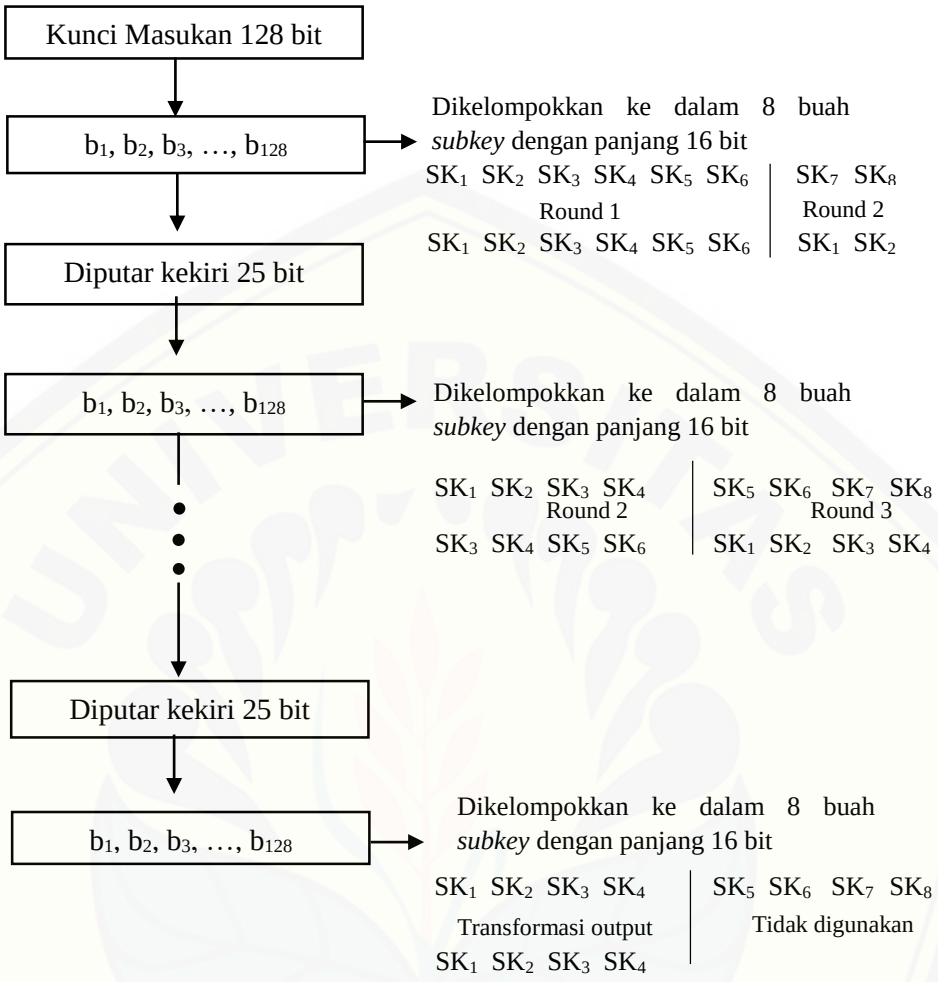
1. Operasi rotasi kiri (*Rotate Left*) yaitu operasi memutar barisan bit ke kiri sebanyak nilai yang diberikan secara per bit, bit kosong yang telah tergeser di sebelah kanan akan digantikan dengan bit yang telah tergeser di sebelah kirinya. Operasi *rotate left* dilambangkan dengan "<<<".
   Contoh operasi *rotate left*: 11000110 <<< 1: 10001101
   11000110 <<< 2: 00011011
2. Operasi rotasi kanan (*Rotate Right*) yaitu operasi memutar barisan bit ke kanan sebanyak nilai yang diberikan secara per bit, bit kosong yang telah tergeser di sebelah kiri akan digantikan dengan bit yang telah tergeser di sebelah kanannya. Operasi *rotate right* dilambangkan dengan ">>>".
   Contoh operasi *rotate right*: 11000110 >>> 1: 01100011
   11000110 >>> 2: 10110001

2.2.2 Pembangkit Kunci

Kunci pada algoritma IDEA menggunakan 16 karakter, dimana setiap karakter diubah dengan merujuk tabel ASCII (*American Standart Code for Information Interchange*) pada Lampiran A, sehingga didapatkan 8 bit perkarakter, maka kunci akan terdiri dari 128 bit. Kode ASCII sendiri merupakan sekumpulan karakter, baik huruf maupun symbol seperti *Hex* dan *Unicode* sesuai standar yang ditentukan. Kode ASCII dibagi beberapa bagian seperti ASCII *Control Characters*, ASCII *Printable Character*, dan *The Extended* ASCII *Codes*.

Proses selanjutnya yaitu membagi 128 *bit key* menjadi 8 buah 16 bit *subkey*. Ini merupakan delapan *subkey* pertama untuk algoritma dengan perincian enam *subkey* pertama untuk putaran (*round*) 1 dan dua *subkey* terakhir untuk putaran selanjutnya. Bit k*ey* dirotasikan 25 bit ke kiri dan dibagi menjadi 8 *subkey* lagi. Ini merupakan delapan *subkey* kedua untuk algoritma dengan perincian empat *subkey* pertama untuk putaran 2 dan empat *subkey* terakhir untuk putaran 3. Algoritma IDEA hanya menggunakan 52 buah *subkey* dengan perincian 6 buah *subkey* untuk 8 putaran ditambah 4 buah *subkey* untuk Transformasi *Output* (TO).

Proses pembentukan kunci dapat dilihat pada Gambar 2.2.



Gambar 2.2 Bagan pembangkit kunci

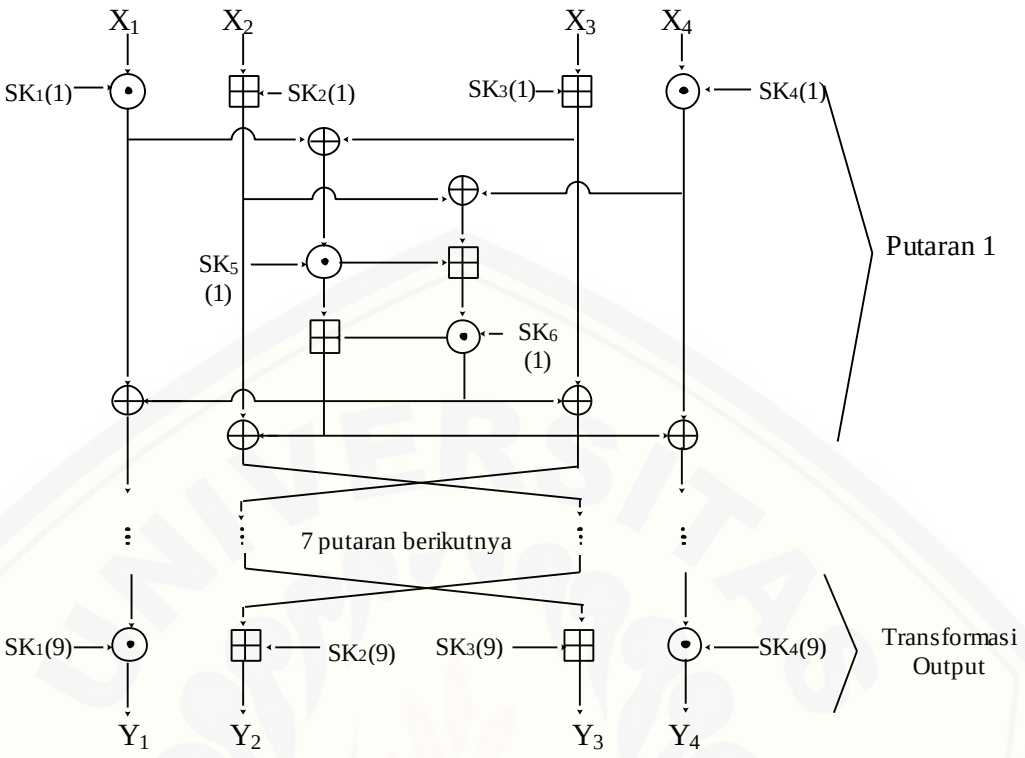
2.2.3 Enkripsi

Proses enkripsi menggunakan IDEA adalah sebagai berikut, pertama *plain text* 64 bit dibagi menjadi 4 buah subblok dengan panjang 16 bit, yaitu X_1, X_2, X_3, X_4. Empat subblok ini menjadi masukan bagi iterasi tahap pertama. Algoritma IDEA terdapat 8 iterasi. Pada setiap iterasi, 4 subblok di-xor-kan, ditambahkan, dikalikan dengan yang lain dan dengan 6 buah *subkey* 16 bit. *Subkey* untuk proses enkripsi dapat dilihat pada Tabel 2.4, dimana antara iterasi subblok kedua dan ketiga saling ditukarkan. Akhirnya 4 buah subblok dikombinasikan

dengan 4 *subkey* dalam transformasi *output*. Pada setiap tahapan, urutan berikut ini dikerjakan:

1. Kalikan X_1 dengan SK_1 mod $(2^{16} + 1)$.
2. Tambahkan X_2 dengan SK_2 mod 2^{16}.
3. Tambahkan X_3 dengan SK_3 mod 2^{16}.
4. Kalikan X_4 dengan SK_4 mod $(2^{16} + 1)$.
5. XOR hasil dari step 1 dan 3.
6. XOR hasil dari step 2 dan 4.
7. Kalikan hasil dari step 5 dengan SK_5 mod $(2^{16} + 1)$.
8. Tambahkan hasil dari step 6 dan 7 mod 2^{16}.
9. Kalikan hasil dari step 8 dengan SK_6 mod $(2^{16} + 1)$.
10. Tambahkan hasil dari step 7 dan 9 mod 2^{16}.
11. XOR hasil dari step 1 dan 9.
12. XOR hasil dari step 3 dan 9.
13. XOR hasil dari step 2 dan 10.
14. XOR hasil dari step 4 dan 10.

Adapun Algoritma IDEA ditampilkan pada Gambar 2.3



Keterangan :

⊞ : Operasi Penjumlahan Modulo (2^{16})

⊙ : Operasi Perkalian Modulo (2^{16} +1)

⊕ : Operasi XOR

Gambar 2.3 Bagan IDEA

Tabel 2.4 *Subkey* untuk enkripsi

Putaran	Enkripsi					
1	$SK_1(1)$	$SK_2(1)$	$SK_3(1)$	$SK_4(1)$	$SK_5(1)$	$SK_6(1)$
2	$SK_1(2)$	$SK_3(2)$	$SK_2(2)$	$SK_4(2)$	$SK_5(2)$	$SK_6(2)$
3	$SK_1(3)$	$SK_3(3)$	$SK_2(3)$	$SK_4(3)$	$SK_5(3)$	$SK_6(3)$
4	$SK_1(4)$	$SK_3(4)$	$SK_2(4)$	$SK_4(4)$	$SK_5(4)$	$SK_6(4)$
5	$SK_1(5)$	$SK_3(5)$	$SK_2(5)$	$SK_4(5)$	$SK_5(5)$	$SK_6(5)$
6	$SK_1(6)$	$SK_3(6)$	$SK_2(6)$	$SK_4(6)$	$SK_5(6)$	$SK_6(6)$
7	$SK_1(7)$	$SK_3(7)$	$SK_2(7)$	$SK_4(7)$	$SK_5(7)$	$SK_6(7)$
8	$SK_1(8)$	$SK_3(8)$	$SK_2(8)$	$SK_4(8)$	$SK_5(8)$	$SK_6(8)$
TO	$SK_1(9)$	$SK_2(9)$	$SK_3(9)$	$SK_4(9)$		

Output dari setiap putaran adalah empat subblok yang dihasilkan pada langkah 11, 12, 13 dan 14 dan menjadi *input*-an putaran selanjutnya. Pada putaran terakhir (putaran 8) subblok 12 dan 13 ditukarkan sehingga *input*-an dari transformasi *output* adalah hasil kombinasi dari langkah 11, 13, 12 dan 14. Setelah 8 putaran akan dilakukan tranformasi *output* dengan algoritma sebagai berikut,

1. Kalikan X_1 dengan *subkey* SK_1 mod $(2^{16} + 1)$.
2. Tambahkan X_2 dengan *subkey* SK_2 mod 2^{16}.
3. Tambahkan X_3 dengan *subkey* SK_3 mod 2^{16}.
4. Kalikan X_4 dengan *subkey* SK_4 mod $(2^{16} + 1)$.

Didapatkan dari transformasi *output* (TO) langkah 1, 2, 3 dan 4 yaitu Y_1, Y_2, Y_3 dan Y_4 sebagai *cipher text*.

2.2.4 Dekripsi

Pada proses dekripsi, IDEA menggunakan algoritma yang sama dengan proses enkripsi namun perbedaanya terletak pada *subkey*-nya. Urutan *subkey* terbalik dengan proses enkripsi dan *subkey*-nya diinverskan. *Subkey* pada putaran 8 diinverskan dan digunakan sebagai *subkey* pada putaran 1 dan 2 pada proses dekripsi. Demikian seterusnya, agar lebih jelas kesuluruhan *subkey* dapat dilihat pada Tabel 2.5 dengan a^{-1} merupakan invers perkalian modulo dan $-a$ merupakan invers penjumlahan modulo.

Tabel 2.5 *Subkey* untuk dekripsi

Putaran	*Subkey*					
1	$SK_1(9)^{-1}$	$-SK_2(9)$	$-SK_3(9)$	$SK_4(9)^{-1}$	$SK_5(8)$	$SK_6(8)$
2	$SK_1(8)^{-1}$	$-SK_3(8)$	$-SK_2(8)$	$SK_4(8)^{-1}$	$SK_5(7)$	$SK_6(7)$
3	$SK_1(7)^{-1}$	$-SK_3(7)$	$-SK_2(7)$	$SK_4(7)^{-1}$	$SK_5(6)$	$SK_6(6)$
4	$SK_1(6)^{-1}$	$-SK_3(6)$	$-SK_2(6)$	$SK_4(6)^{-1}$	$SK_5(5)$	$SK_6(5)$
5	$SK_1(5)^{-1}$	$-SK_3(5)$	$-SK_2(5)$	$SK_4(5)^{-1}$	$SK_5(4)$	$SK_6(4)$
6	$SK_1(4)^{-1}$	$-SK_3(4)$	$-SK_2(4)$	$SK_4(4)^{-1}$	$SK_5(3)$	$SK_6(3)$
7	$SK_1(3)^{-1}$	$-SK_3(3)$	$-SK_2(3)$	$SK_4(3)^{-1}$	$SK_5(2)$	$SK_6(2)$
8	$SK_1(2)^{-1}$	$-SK_3(2)$	$-SK_2(2)$	$SK_4(2)^{-1}$	$SK_5(1)$	$SK_6(1)$
TO	$SK_1(1)^{-1}$	$-SK_2(1)$	$-SK_3(1)$	$SK_4(1)^{-1}$		

2.3 Transformasi Wavelet

Transformasi wavelet merupakan suatu proses pengubahan data dalam bentuk lain agar lebih mudah dianalisis. Menurut Sydney (1998), wavelet merupakan gelombang mini (*small wave*) yang mempunyai kemampuan mengelompokkan energi citra dan terkonsentrasi pada sekelompok kecil koefisien, sedangkan kelompok koefisien lainnya hanya mengandung sedikit energi yang dapat dihilangkan tanpa mengurangi nilai informasinya.

Transformasi wavelet mempunyai penerapan yang luas pada aplikasi pengolahan isyarat dan pengolahan citra. Ada berbagai jenis transformasi wavelet, akan tetapi pada bagian ini lebih menitik beratkan pada transformasi wavelet yaitu *Discrete Wavelet Transform* (DWT).

Transformasi wavelet 1-D membagi sinyal menjadi dua bagian, frekuensi tinggi dan frekuensi rendah berturut-turut dengan *low-pass filter* dan *high-pass filter*. Frekuensi rendah dibagi kembali menjadi frekuensi tinggi dan rendah. Proses diulang sampai sinyal tidak dapat didekomposisi lagi atau sampai pada level yang memungkinkan. Pembagian sinyal menjadi frekuensi rendah dalam proses filterisasi yaitu *highpass filter* dan *lowpass filter* disebut dekomposisi. Sinyal asli dapat dipulihkan kembali melalui rekonstruksi dari sinyal yang telah didekomposisi dengan menerapkan *Inverse Discrete Wavelet Transform* (IDWT).

Pada penilitian kali ini akan menggunakan dekomposisi Haar Wavelet. Haar wavelet mengubah citra dengan domain spasial ke domain frekuensi dengan persamaan berikut ini:

$$H_0: f(n) = \frac{X_n + X_{n+1}}{2} \tag{2.2}$$

$$H_1: f(n) = \frac{X_n - X_{n+1}}{2} \tag{2.3}$$

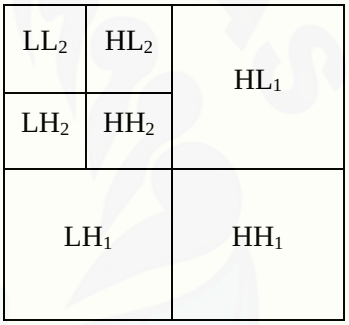
Dimana H_0 merupakan *low pass filter*, H_1 merupakan *high pass filter* dan $X = \{X_n\} | n = 1,2, \ldots, N$ dimana X_n merupakan *pixel-pixel* dari matrik citra. Pada *Inverse Discrete Wavelet Transform* (IDWT) mentransformasi kembali ke domain spasial menggunakan persamaan berikut:

$$Y_{2n-1} = X_n + X_{n+m} \quad (2.4)$$

$$Y_{2n} = X_n - X_{n+m} \quad (2.5)$$

Transformasi wavelet 2-dimensi (2-D) proses dekomposisi sama dengan pada 1 dimensi. Hanya saja proses dekomposisi dilakukan dalam 2 tahap, yaitu dilakukan pada seluruh baris dan tahap kedua pada citra hasil tahap pertama dilakukan dekomposisi dalam arah kolom. Hingga menghasilkan subband LL, LH, HL dan HH. Pada DWT tingkat 2 proses dekomposisi dilakukan kembali pada subband LL (Zakaria, 2015). Setelah didapatkan semua subband maka penyisipan secara modifikasi LSB akan dilakukan pada subband HL, LH dan HH.

(a)



(b)

(c)

(d)

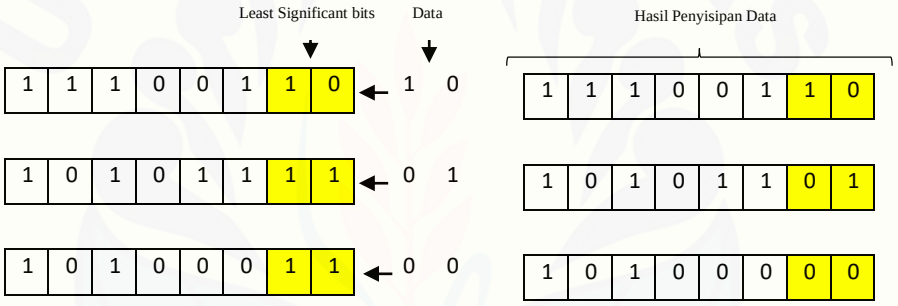
(a) Citra barbara.png asli, (b) Struktur DWT dua tingkat, (c) Dekomposisi satu tingkat, (d) Dekomposisi dua tingkat.

Gambar 2.4 Citra tempat penyisipan

2.4 LSB (*Least Significant Bits*)

LSB *(Least Significant Bist)* merupakan salah satu metode dalam steganografi. LSB dilakukan dengan mengambil bit–bit terakhir warna pada citra dan menggantinya dengan bit – bit data. Tujuan utama dari LSB adalah memanipulasi nilai suatu titik warna (*pixel*) sehingga data dapat disembunyikan ke dalam titik warna tersebut namun perubahan yang terjadi berusaha diminimalisasi sehingga seakan – akan perubahannya tidak dapat dideteksi oleh mata manusia.

File citra pesan dapat disembunyikan dengan menggunakan cara menyisipkannya pada bit rendah atau bit yang paling kanan yang biasa disebut lsb, pada data *pixel* yang menyusun file tersebut. Misalkan kita memiliki data 100100 akan disisipkan pada 3 *pixel* gambar 11100110, 10101111, 10100011.



Gambar 2.5 *Least Significant Bits* proses

Pada Modifikasi LSB dilakukan pada *pixel-pixel* yang memiliki nilai riil. Pada bilangan yang bernilai negatif akan dimutlakan dahulu setelah itu apabila nilainya berbentuk pecahan maka akan dibagi 2 kelompok yaitu kelompok bilangan bulat dan bilangan setelah koma. Bilangan bulat yang akan diproses proses penyisipan dilakukan seperti biasanya yang nantinya akan dikembalikan lagi nilai setiap *pixel* serta penambahan nilai bilangan pecahanya (bilangan bulat setelah koma). Misalkan nilai sebuah *pixel* gambar -89,5 $_{(10)}$ akan disipkan biner 10 $_{(2)}$, nilai $|-89{,}5| = 89{,}5$ dan biner dari 89 (1011001) maka penyisipan biner 10 akan dilakukan pada biner 1011001 sehingga didapatkan biner baru 1011010 sehingga didapatkan bilangan desimal 90 karena menyimpan nilai 0,5 sebelumnya maka 90 akan ditambahkan dengan nilai 0,5 sehingga nilainya menjadi 90,5 setelah itu akan

dikembalikan pada nilai nya. Karena nilai awal bernilai negatif maka nilai *pixel* akan menjadi -90,5.

2.5 Audio WAV

Berkas audio WAV atau WAVE merupakan standar format berkas audio yan digunakan oleh IBM dan Microsoft dalam menyimpan aliran data audio pada PC. Berkas audio WAV menerapkan teknik Linier Pulse Code Modulation (LPCM) dalam mempresentasikan data. LPCM merupakan salah satu jenis PCM yang menggunakan metode lossless dan tanpa kompresi, yaitu metode dengan menyimpan seluruh sampel audio, sehingga berkas WAV merupakan berkas mentah (sesuai dengan aslinya) (Rasyid, 2009).

Kualitas dari file wave ditentukan oleh *bitrate, samplerate* dan jumlah *channel*. *Bitrate* merupakan ukuran bit tiap *sample*-nya, dapat disimpan per 8 bits, 16 bits dan 32 bits. Semakin besar *bitrate* dalam satu *sample* suara makin baik kualitas suara file tersebut, karena data yang disimpan semakin akurat. *Samplerate* menyatakan banyaknya jumlah *sample* yang dimainkan setiap detiknya. *Samplerate* yang umum dipakai adalah 8000 Hz, 11025 Hz (untuk perkaman suara manusia), 22050 Hz (untuk perekaman suara musik) dan 44100 Hz (sering dipakai dalam audio cd karena cocok untuk semua jenis suara) (Soleh, 2010).

Jumlah *channel* akan menentukan suara yang dihasilkan termasuk *mono* atau *stereo*. *Mono* menggunakan 1 *channel* suara, sedangkan *stereo* menggunakan lebih dari 1 *chanel* suara (umumnya 2). Suara manusia dapat direkam secara *mono*, sedangkan file-file untuk kualitas cd direkam secara *stereo* (Soleh,2010).

Data audio yang memiliki rentang -1 hingga 1 akan diubah ke dalam rentang 0-255 dengan menggunakan fungsi:

$$f(x) = \left\lfloor \frac{255x+255}{2} \right\rfloor \tag{2.6}$$

dimana x merupakan data audio dari setiap frekuensinya. Sedangkan untuk mengubah data dengan rentang 0-255 kedalam rentang -1 hingga 1 menggunakan fungsi:

$$f(y) = \frac{2y}{255} - 1 \quad (2.7)$$

dimana y merupakan data dengan rentang 0 hingga 255.

2.6 Citra

Citra merupakan gambar pada bidang dua dimensi (dwimatra) dan merupakan suatu fungsi kontinu dari intensitas cahaya pada bidang dua dimensi tersebut. Citra digolongkan menjadi 2 bentuk yaitu citra diam dan citra bergerak. Citra diam merupakan citra tunggal yang tidak bergerak, sedangkan citra bergerak merupakan kumpulan citra diam yang ditampilkan secara beruntun sehingga tampak bergerak oleh pandangan visual (Munir, 2004).

Secara matematis, fungsi intensitas cahaya pada suatu citra di bidang dwimatra disimbolkan dengan *f*(*x*, *y*), dimana:

(*x*, *y*) : koordinat pada bidang dwimatra

F(*x*, *y*): intensitas cahaya (*brightness*) pada titik (*x*, *y*)

Citra digital berukuran $\mathrm{N} \times \mathrm{M}$ juga dapat direpresentasikan dalam bentuk matriks dimana N mempresentasikan baris dan M mempresentasikan kolom, sehingga didapat sebagai berikut:

$$f(x,y) = \begin{bmatrix} f(1,1) & \dots & f(1,M) \\ \vdots & \ddots & \vdots \\ f(N,1) & \dots & \mathrm{f(N,M)} \end{bmatrix}$$

Banyaknya *element* matriks tersebut berjumlah $\mathrm{N} \times \mathrm{M}$ buah, dimana banyaknya intensitas pada satu citra tersebut diistilahkan sebagai *image element*, *picture element*, *pixel*, atau *pel.* Indek baris (*i*) dan indeks kolom (*j*) menyatakan suatu koordinat titik pada suatu citra digital dan *f*(*i, j)* merupakan intensitas (derajat keabuan) pada titik (*i*, *j*) (Dulimarta, 1997).

2.6.1 Citra *Grayscale* (Skala Keabuan)

Pada citra *grayscale* jumlah bit yang disediakan oleh memori penampung kebutuhan warna mempengaruhi banyak gradasi warna yang terbentuk. Pada citra *grayscale*, warna hitam menunjukkan intensitas terlemah dan warna putih

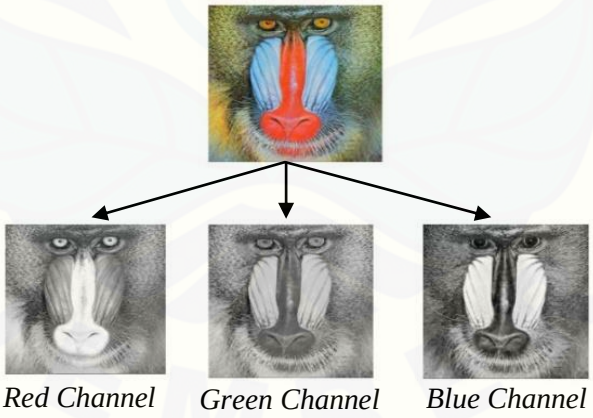
menunjukkan intensitas terkuat. Variasi warna diantara hitam dan putih yang banyak menyebabkan warna terlihat abu-abu (*grayscale*). Misalkan 2 bit (2^2) mewakili 4 warna, 3 bit (2^3) mewakili 8 warna dan maksimal hingga 8 bit mewakili 256 warna (Hakim, 2012).

Gambar 2.6 Gradasi citra *grayscale* 3 bit

2.6.2 Citra RGB

Tipe ini digunakan untuk melakukan manipulasi sebuah *pixel* pada citra. RBG merupakan *record* yang mempunyai tiga kanal yaitu *rgbred*, *rgbgreen*, *rgbblue* yang secara berurutan merepresentasikan nilai RGB suatu *pixel*. Intensitas suatu titik pada citra merupakan kombinasi dari tiga intensitas yaitu:

a. derajat keabuan merah ($f_r(x, y)$)
b. derajat keabuan hijau ($f_g(x, y)$)
c. derajat keabuan biru ($f_b(x, y)$)



Gambar 2.7 Citra RGB b*aboon.png*

2.7 Analisis Keamanan

Pada proses perlindungan data gambar dan audio terdapat beberapa metode yang dapat digunakan untuk menganalisis keamanan dari suatu algoritma yang digunakan. Berikut ini merupakan beberapa analisis keamanan yang ada, dengan kegunaan dan fungsinya tersendiri.

2.7.1 SNR (*Signal to Noise Ratio*)

SNR (*Signal to Noise Ratio*) didefinisikan sebagai ratio antara daya sinyal yang diinginkan dengan daya derau (*noise*). Derau pada sinyal merupakan gangguan pada sinyal yang menyebabkan rusaknya sinyal informasi tersebut. Perhitungan SNR dapat dilakukan dengan metode korelasi. Metode korelasi dilakukan dengan membandingkan dua runtun data (sinyal) yang masing-masing nilai sampelnya diambil secara serempak (Haq, 2012). Perhitungan SNR menggunakan persamaan berikut:

$$\rho = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2}.\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{2.8}$$

$$\text{SNR} = \frac{\text{sinyal}}{\text{derau}} = \left(\frac{\rho}{1-\rho}\right) \tag{2.9}$$

dimana ρ merupakan koefisien korelasi antara sinyal informasi dan sinyal berderau. Taguchi merekomendasikan untuk menggunakan logaritma SNR dikalikan 10 sebagai bentuk ratio dalam desibel (dB). Desibel mengekspresikan suatu perbandingan, perbandingan tersebut dapat berupa daya, tekanan suara, tegangan atau hal-hal lainnya (Listyoanik, 2012).

$$\text{dB} = 10 \times \log_{10}\left(\frac{P_1}{P_2}\right) \tag{2.10}$$

dimana P_1 dan P_2 merupakan 2 daya yang akan dibandingkan. SNR dalam desibel (dB) dimaksudkan untuk mempermudah perhitungan jika nilai dalam persamaan 2.9 relatif besar. Penggunaan SNR dalam desibel (dB) tidak mempengaruhi pada analisis hasil yang dihasilkan (Listyoanik, 2012). Sehingga bentuk persamaan SNR menjadi:

$$\text{SNR}_{dB} = 10 \times \log_{10}\left(\frac{\rho}{1-\rho}\right) \tag{2.11}$$

Semakin besar nilai SNR_{dB} semakin baik kualitas sinyal yang dihasilkan atau dengan kata lain semakin kecil deraunya. Sinyal dikatakan baik apabila nilai SNR_{dB} lebih besar dari atau sama dengan 25dB sedangkan untuk nilai kurang dari atau sama dengan 13dB dimana terdapat derau atau *noise* yang besar dalam sinyal (Fitri, 2014).

2.7.2 PSNR (*Peak Signal to Noise Ratio*)

PSNR adalah sebuah istilah dalam bidang teknik yang menyatakan perbandingan antara kekuatan sinyal maksimum yang mungkin dari suatu sinyal digital dengan kekuatan derau yang mempengaruhi kebenaran sinyal tersebut. Sama halnya dengan SNR hanya saja penggunaan PSNR digunakan pada suatu citra. Nilai PSNR dikatakan memiliki kemiripan yang tinggi jika nilai PSNR lebih besar atau sama dengan 40dB (Hakim, 2012). Perhitungan PSNR dilakukan dengan menghitung nilai MSE pertama kali. *Mean Square Eror* (*MSE*) dihitung untuk seluruh *pixel* dalam citra.

$$\text{MSE} = \frac{\sum_{i=1,j=1}^{i=n,j=m}(f(i,j)-F(i,j))^{2}}{M \times N} \tag{2.12}$$

M×N merupakan perkalian panjang dan lebar suatu citra dalam *pixel*. $F(i,j)$ merupakan citra hasil rekontruksi, sedangkan $f(i,j)$ adalah citra asal.

$$\text{PSNR} = 10 \times \log_{10}\left(\frac{255^{2}}{MSE}\right) \tag{2.13}$$

2.7.3 Analisis Sensitivitas Kunci

Sebuah pesan akan menghasilkan *cipher* yang berbeda saat menggunakan kunci yang berbeda pada proses enkripsi (Prasetyo, 2012). Namun apabila dua kunci yang digunakan itu mirip, *cipher* yang dihasilkan haruslah berbeda pula. Oleh karena itu dikenal dengan istilah sensitivitas kunci. Sensitivitas kunci juga dikenal sebagai *avalanche effect* (Mishra, 2012), dimana *avalanche effect* menyatakan saat perubahan sedikit (1 bit) pada kunci asli harus menyebabkan *output* (*cipher text*) akan berubah signifikan dari *cipher text* yang menggunakan kunci asli (Dawson, 1992).

Selain itu kunci dikatakan sensitif jika *plain text* tidak dapat diperoleh seperti semula jika ada perbedaaan kunci antara proses enkripsi dan dekripsi (Song, 2015) atau menurut Munir (2012) saat mengubah sedikit saja (1 bit) kuncinya pada saat proses dekripsi akan menyebabkan gagal mengembalikan *cipher text* menjadi *plain text* seperti semula. Pengukuran perbedaan data, diukur menggunakan persamaan (2.11). *Avalanche effect* sendiri dikemukakan pertama kali oleh Fiestel (1973).

BAB 3. METODE PENELITIAN

3.1 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data audio berformat ∗.wav sebagai *plain audio*, teks sebagai kuncinya serta gambar yang berformat ∗.jpg, ∗.png, ∗.bmp maupun ∗.tif yang merupakan tempat penyisipan pesan.

3.2 Langkah-langkah Penelitian

Langkah-langkah penelitian yang dilakukan dalam penelitian ini adalah sebagai berikut:

a. Studi Literatur

Tahap ini dilakukan dengan mempelajari mengenai teori-teori yang dipakai sebagai acuan penelitian. Teori yang dipakai dalam penelitian ini adalah teknik kriptografi IDEA (*International Data Encryption Algorithm*) untuk mengenkripsi dan mendekripsi pesan berupa audio, serta teknik DWT (*Discrete Wavelet Transform)* untuk memecah citra menjadi beberapa subband dan teknik Steganografi LSB (*Least Significant Bit*) untuk proses penyembunyian (*embedding*) dan pengungkapan (*extraction*) data.

b. Analisis Data

1. Proses Pembangkitan Kunci

   Setiap karakter dari kunci awal akan dikonversi menjadi bentuk biner sehingga terbentuk barisan biner 128bit yang digunakan sebagai kunci biner awal dan dilakukan pembangkitan kunci seperti pada Gambar 2.2.

2. Proses Enkripsi

   a) *Plain audio* dikonversi ke dalam betuk desimal dengan rentang 0 hingga 255 dengan persamaan 2.6.

   b) Ubah tiap *plain audio* (frekuensi) yang berbentuk desimal ke dalam bentuk biner 8 bit.

c) Susun ke dalam subblok *plain audio*, dengan 64 bit setiap subbloknya yang akan dijadikan *plain audio* setiap prosesnya.

d) Lakukan setiap subblok ke dalam proses enkripsi menggunakan IDEA dengan kunci yang sudah didapatkan pada proses pembangkit kunci.

e) Hasil proses enkripsi (*cipher audio*) yang berbentuk biner dengan setiap blok 16 bit dan bagi ke dalam 8 bit setiap subblok sehingga didapatkan *cipher audio* 8 bit persampelnya.

3. Proses *Embedding*

a) Ubah citra *cover* menjadi beberapa subband menggunakan transformasi *wavelet* DWT

b) Sisipkan data audio terenkripsi menggunakan metode LSB (2 bit terakhir tiap *pixel).*

c) Penyisipan dilakukan dengan meyisipkan semua data pada *pixel* dalam subband HL kemudian apabila data terenkripsi masih tersisa, maka proses penyisipan dilakukan pada subband LH kemudian HH.

d) Penyisipan pada setiap subband dilakukan dengan mengubah semua LSB (2 bit terakhir) pada setiap *pixel* baris, setelah terisi semua lanjutkan pada baris selanjutnya hingga semua *pixel* LSB tergantikan dengan data terenkripsi.

e) Citra yang telah tersisipi dilakukan IDWT untuk didapatkan citra seperti semula (citra asli).

4. Proses Ekstraksi

a) Citra telah tersisipi dilakukan DWT untuk mendapatkan subband HL, LH dan HH.

b) Membaca semua 2 bit terakhir dari semua *pixel* pada sub band HL, LH dan HH.

c) Mengelompokkan semua data biner menjadi data biner.

d) Citra diproses IDWT kembali.

5. Proses Dekripsi

Pada hasil ekstraksi didapatkan data berbentuk biner dan kemudian dibagi subblok dimana terdiri dari 64 bit setiap subbloknya. Proses subblok

dengan IDEA setelah didapatkan kunci dekripsi. Setelah didapatkan hasil dari proses dekripsi maka didapatkan 4 data dengan rentang 0 hingga 255 yang kemudian dikonversi ke dalam rentang -1 higga 1 menggunakan persamaan (2.7) dan dijadian file audio (∗.wav).

c. Perancangan Program

Perancangan program dilakukan dengan menggunakan bahasa pemrograman Matlab. Desain program menggunakan aplikasi GUI (*Graphic User Interface*) yang ada dalam Matlab untuk merancang *form layout* dan desain interaksi program agar lebih menarik.

d. Pembuatan Program

Pembuatan program didasarkan pada konsep enkripsi dan dekripsi algoritma IDEA kriptografi serta konsep *embedding* dan *extraction* untuk steganografi.

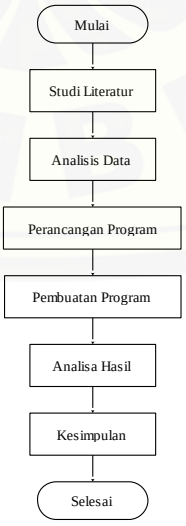
e. Analisis Hasil

Tahap pengujian serta menganalisis program dan menganalisis keamanan dari algoritma yang diajukan agar sesuai dengan konsep teori yang digunakan.

f. Kesimpulan

Mengambil kesimpulan dari penelitian yang telah dilakukan, yaitu dengan menganalisis proses dan hasil pengamanan dan pengungkapan suatu pesan audio menggunakan metode yang di ajukan.

Diagram alir langkah –langkah penelitian adalah sebagai berikut,



Gambar 3.2 Diagram alir penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut:

a. Proses pengamanan pada data audio berformat $*$.wav dapat dilakukan dengan melakukan enkripsi menggunakan algoritma IDEA yang selanjutnya melakukan *embedding* data audio terenkripsi ke dalam sebuah citra menggunakan DWT dan LSB. Proses enkripsi dapat memberikan keamanan bagi data audio karena hasil dari enkripsi (*cipher audio*) sangatlah berbeda dengan *plain audio*-nya. Serta proses *embedding* pada citra dapat meningkatkan keamanan bagi data audio terenkripsi karena data audio terenkripsi yang berada di dalam sebuah citra tidak akan menimbulkan kecurigaan sebab citra hasil proses *embedding* tidak memiliki perbedaan dengan citra sebelum proses *embedding* karena PSNR dari beberapa data uji menunjukkan nilai lebih dari 40dB.

b. Proses pengungkapan data dilakukan dengan melakukan proses ekstraksi pada citra sehingga didapatkan data audio terenkripsi, kemudian dilakukan proses dekripsi dengan algortima IDEA. Proses pengungkapan data mampu mengembalikan sebuah citra berisi data rahasia menjadi *plain audio* kembali, dengan nilai dari data audio asli dengan data audio setelah proses pengamanan data menunjukkan nilai SNR_{dB} lebih dari 25dB, sehingga data audio dapat dikatakan bagus (derau tidak merusak audio asli).

c. Berdasarkan analisis keamanan, algoritma yang diajukan dalam proses enkripsi aman untuk proses perlindungan data audio karena memiliki kunci yang sensitif, serta memiliki keamanan ganda yang baik setelah melewati proses *embedding* data audio kedalam sebuah citra karena memiliki nilai PNSR lebih besar dari 40dB maka dapat dikatakan sangat bagus (tidak memiliki perbedaan yang signifikan dengan citra asli).

5.2 Saran

Saran yang diberikan untuk penelitian selanjutnya yaitu dapat menerapkan pengamanan audio dengan format lain seperti ∗.mp3, ∗.MPEG-4 dan lainya. Dapat menerapkan keamanan data audio menggunakan algoritma enkripsi modern yang lain dan terbaru selain itu pada penelitian selanjutnya dapat memperbaiki proses konversi data audio menjadi data desimal sebelum masuk kedalam proses enkripsi, sehingga hasil dari pengungkapan tidak mengalami distorsi saat file audio yang telah didekripsi kembali.

DAFTAR PUSTAKA

Dawson, E., H. Gustafon., A. N. Pettitt. 1992. Strict Key Avalanche Criterion. *Australasian Journal of Combinatorics*. 6: 147-153

Dulimarta, H.S. 1997.*Diktat Kuliah Pengolahan Citra.*Bandung: Jurusan Teknik Informatika Institut Teknologi Bandung.

Feistel, H. 1973. Cryptography and Computer Privacy. *Scientific American*. 228(5): 15-23

Fitri, N. A., Srihendayana, H., Dasril. 2014. *Analisis Kualitas Jaringan Useetv Cable menggunakan kabel tembaga pada PT Telkom Pontianak*. Pontianak: Jurusan Teknik Elektro Fakultas Teknik Universitas Tanjungpura.

Goel, S., A. Rana., M. Kaur. 2013. A Review of Comparison Techniques of Image Steganography. *Global Journals of Computer Science and Technology*. 17(4-F)

Hakim, A. R. 2012. *Analisa Perbandingan Watermarking Image Menggunakan Discrete Wavelet Transform. Skripsi.* Depok: Fakultas Teknik Universitas Indonesia.

Hanan, A. 2013. *Metode Enkripsi Dan Deskripsi Data menggunakan Kriptografi Idea. Skripsi.* Aceh: Teknik Informatika Sekolah Tinggi Manajemen Informatika Dan Komputer Stmik U'budiyah Indonesia.

Haq, A. D., Santoso, I., Macrina, A. A. 2012. *Estimasi Signal to Noise Ratio (SNR) Menggunakan Metode Korelasi*. Semarang: Universitas Diponegoro.

Listyoanik, P. S. 2012. *Peningkatan Kualitas Batu Bata Dengan Metode Taguchi. Skripsi*. Surabaya: Program Studi Matematika Departemen Matematika Fakultas Sains Dan Teknologi Universitas Airlangga.

Mishra, M., & Mankar, V.H. 2012. Hybrid Message-Embedded Cipher using Logistic Map. *IJSPTM*. 1(3/4): 81-91.

Munir, R. 2004. *Pengolahan Citra digital dengan Pendekatan Algoritmik*, Bandung:Informatika Bandung.

Munir, R. 2006. *Diktat Kuliah IF504 Kriptografi*. Jakarta: Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika.

Munir,R. 2012. Analisis Keamanan Algoritma Enkripsi Citra Digital Menggunakan Kombinasi Dua Chaos Map dan Penerapan Teknik Selektif. *Juti*. 10(2): 45-51.

Pranoto, B. 2011. *Steganografi Pada Citra Digital Menggunakan Metode Spread Spectrum Dan Metode Least Significant Bit (Lsb) Modification*. *Skripsi*. Pekanbaru: Universitas Islam Negeri Sultan Syarif Kasim Riau.

Prasetyo, E. 2012. *Diktat Kuliah Kriptografi (Simetrik Key)*. Gresik: Program Studi Teknik Informatika, Universitas Muhammadiyah Gresik.

Rasyid, M. F. 2009. *Kriptografi Audio dengan Teknik Interferensi Data Non Biner*. *Skripsi*. Bandung: Fakultas Informatika Institut Teknologi Bandung.

Soleh, M. 2010, *Analisis Dan Implementasi Watermarking Dengan Algoritma Aes Untuk Pemberian Data Hak Cipta Pada File Audio*. *Skripsi*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah.

Song, C., & Y. Qiao. 2015. A Novel Image Encryption Algortithm Based on DNA encoding and Spatiotempral Chaos. *Entropy*. 17: 6954-6968.

Sydney, B. C., A. G. Remesg, G. Haito. 1998. *Introduction to Wavelets and Wavelet Transform.* New Jersey: Prentice-Hall International, Inc.

Zakaria, A., & R. Munir. 2015. *Steganografi citra digital menggunakan teknik discrete wavelet transform pada ruang CIELab*. *Konferensi Nasional Sistem Informasi 2015 Fakultas Ilmu Komputer Universitas Klabat*.

LAMPIRAN A. Tabel ASCII (*American Standart Code for Information Interchange*)

Dec	*Hex*	*Binary*	*character*	*Dec*	*Hex*	*Binary*	*character*
1	1	00000001	SOH	43	2B	00101011	+
2	2	00000010	STX	44	2C	00101100	,
3	3	00000011	ETX	45	2D	00101101	-
4	4	00000100	EOT	46	2E	00101110	.
5	5	00000101	ENQ	47	2F	00101111	/
6	6	00000110	ACK	48	30	00110000	0
7	7	00000111	BEL	9	31	00110001	1
8	8	00001000	BS	50	32	00110010	2
9	9	00001001	HT	51	33	00110011	3
10	0A	00001010	LF	52	34	00110100	4
12	0C	00001100	FF	53	35	00110101	5
13	0D	00001101	CR	54	36	00110110	6
14	0E	00001110	SO	55	37	00110111	7
15	0F	00001111	SI	56	38	00111000	8
16	10	00010000	DLE	57	39	00111001	9
17	11	00010001	DC1	58	3A	00111010	:
18	12	00010010	DC2	59	3B	00111011	;
19	13	00010011	DC3	60	3C	00111100	<
20	14	00010100	DC4	61	3D	00111101	=
21	15	00010101	NAK	62	3E	00111110	>
22	16	00010110	SYN	63	3F	00111111	?
23	17	00010111	ETB	64	40	01000000	@
24	18	00011000	CAN	65	41	01000001	A
25	19	00011001	EM	66	42	01000010	B
26	1A	00011010	SUB	67	43	01000011	C
27	1B	00011011	ESC	68	44	01000100	D
28	1C	00011100	FS	69	45	01000101	E
29	1D	00011101	GS	70	46	01000110	F
30	1E	00011110	RS	71	47	01000111	G
31	1F	00011111	US	72	48	01001000	H
32	20	00100000	Space	73	49	01001001	I
33	21	00100001	!	74	4A	01001010	J
34	22	00100010	"	75	4B	01001011	K
35	23	00100011	#	76	4C	01001100	L
36	24	00100100	$	77	4D	01001101	M
37	25	00100101	%	78	4E	01001110	N
38	26	00100110	&	9	4F	01001111	O
39	27	00100111	'	80	50	01010000	P
40	28	00101000	(	81	51	01010001	Q
41	29	00101001	)	82	52	01010010	R
42	2A	00101010	*	83	53	01010011	S

Dec	*Hex*	*Binary*	*character*	*Dec*	*Hex*	*Binary*	*character*
84	54	01010100	**T**	130	82	10000010	‚
85	55	01010101	**U**	131	83	10000011	Ƒ
86	56	01010110	**V**	132	84	10000100	„
87	57	01010111	**W**	133	85	10000101	…
88	58	01011000	**X**	134	86	10000110	†
89	59	01011001	**Y**	135	87	10000111	‡
90	5A	01011010	**Z**	136	88	10001000	ˆ
91	5B	01011011	**[**	137	89	10001001	‰
92	5C	01011100	\	138	8A	10001010	Š
93	5D	01011101	**]**	139	8B	10001011	‹
94	5E	01011110	^	140	8C	10001100	Œ
95	5F	01011111	_	141	8D	10001101	•
96	60	01100000	`	142	8E	10001110	Ž
97	61	01100001	**a**	143	8F	10001111	•
98	62	01100010	**b**	144	90	10010000	•
99	63	01100011	**c**	145	91	10010001	‘
100	64	01100100	**d**	146	92	10010010	’
101	65	01100101	**e**	147	93	10010011	“
102	66	01100110	**f**	148	94	10010100	”
103	67	01100111	**g**	149	95	10010101	•
104	68	01101000	**h**	150	96	10010110	–
105	69	01101001	**i**	151	97	10010111	—
106	6A	01101010	**j**	152	98	10011000	˜
107	6B	01101011	**k**	153	99	10011001	™
108	6C	01101100	**l**	154	9A	10011010	š
109	6D	01101101	**m**	155	9B	10011011	›
110	6E	01101110	**n**	156	9C	10011100	œ
111	6F	01101111	**o**	157	9D	10011101	•
112	70	01110000	**p**	158	9E	10011110	ž
113	71	01110001	**q**	159	9F	10011111	Ÿ
114	72	01110010	**r**	160	A0	10100000	
116	74	01110100	**t**	161	A1	10100001	¡
117	75	01110101	**u**	162	A2	10100010	¢
118	76	01110110	**v**	163	A3	10100011	£
119	77	01110111	**w**	164	A4	10100100	¤
120	78	01111000	**x**	165	A5	10100101	¥
121	79	01111001	**y**	166	A6	10100110	¦
122	7A	01111010	**z**	167	A7	10100111	§
123	7B	01111011	**{**	168	A8	10101000	¨
124	7C	01111100	**|**	169	A9	10101001	©
125	7D	01111101	**}**	170	AA	10101010	ª
126	7E	01111110	~	171	AB	10101011	«
127	7F	01111111	DEL	172	AC	10101100	¬
128	80	10000000	€	173	AD	10101101	
129	81	10000001	•	174	AE	10101110	®

Dec	*Hex*	*Binary*	*character*	*Dec*	*Hex*	*Binary*	*character*
175	AF	10101111	¯	223	DF	11011111	ß
176	B0	10110000	°	225	E1	11100001	Á
177	B1	10110001	±	226	E2	11100010	Â
178	B2	10110010	²	227	E3	11100011	Ã
179	B3	10110011	³	228	E4	11100100	Ä
180	B4	10110100	´	229	E5	11100101	Å
181	B5	10110101	µ	230	E6	11100110	Æ
182	B6	10110110	¶	231	E7	11100111	Ç
183	B7	10110111	·	232	E8	11101000	È
184	B8	10111000	¸	233	E9	11101001	É
185	B9	10111001	¹	234	EA	11101010	Ê
186	BA	10111010	º	235	EB	11101011	Ë
187	BB	10111011	»	236	EC	11101100	Ì
188	BC	10111100	¼	237	ED	11101101	Í
189	BD	10111101	½	238	EE	11101110	Î
190	BE	10111110	¾	239	EF	11101111	Ï
191	BF	10111111	¿	240	F0	11110000	ð
192	C0	11000000	À	227	E3	11100011	Ã
193	C1	11000001	Á	228	E4	11100100	Ä
194	C2	11000010	Â	229	E5	11100101	Å
195	C3	11000011	Ã	230	E6	11100110	Æ
196	C4	11000100	Ä	231	E7	11100111	Ç
197	C5	11000101	Å	232	E8	11101000	È
198	C6	11000110	Æ	233	E9	11101001	É
199	C7	11000111	Ç	234	EA	11101010	Ê
200	C8	11001000	È	235	EB	11101011	Ë
201	C9	11001001	É	236	EC	11101100	Ì
202	CA	11001010	Ê	237	ED	11101101	Í
203	CB	11001011	Ë	238	EE	11101110	Î
204	CC	11001100	Ì	239	EF	11101111	Ï
205	CD	11001101	Í	240	F0	11110000	ð
206	CE	11001110	Î	241	F1	11110001	ñ
207	CF	11001111	Ï	242	F2	11110010	ò
208	D0	11010000	Ð	243	F3	11110011	ó
209	D1	11010001	Ñ	244	F4	11110100	ô
210	D2	11010010	Ò	245	F5	11110101	õ
211	D3	11010011	Ó	246	F6	11110110	ö
212	D4	11010100	Ô	247	F7	11110111	÷
213	D5	11010101	Õ	248	F8	11111000	ø
214	D6	11010110	Ö	249	F9	11111001	ù
215	D7	11010111	×	250	FA	11111010	ú
216	D8	11011000	Ø	251	FB	11111011	û
217	D9	11011001	Ù	252	FC	11111100	ü
218	DA	11011010	Ú	253	FD	11111101	ý
219	DB	11011011	Û	254	FE	11111110	þ
220	DC	11011100	Ü	255	FF	11111111	ÿ
221	DD	11011101	Ý				
222	DE	11011110	Þ				

LAMPIRAN B. Matrik data *plainaudio* *"not-a-dream-whats-happening-to-place.wav"*

Kolom	*Plainaudio*	*Data 8 bit*	Kolom	*Plainaudio*	*Data 8 bit*	Kolom	*Plainaudio*	*Data 8 bit*
1	-0,0010	127	25	-0,0045	127	⋮	⋮	⋮
2	-0,0012	127	26	-0,0047	127	44712	0,0000	128
3	-0,0008	127	27	-0,0049	127	44711	0,0000	128
4	0,0002	128	28	-0,0052	127	44710	0,0000	128
5	0,0008	128	29	-0,0056	127	44709	0,0000	128
6	0,0007	128	30	-0,0060	127	44708	0,0000	128
7	0,0002	128	31	-0,0064	127	44707	0,0000	128
8	-0,0009	127	32	-0,0067	127	44706	0,0000	128
9	0,0001	128	33	-0,0071	127	44705	0,0000	128
10	0,0008	128	34	-0,0075	127	44704	0,0000	128
11	0,0013	128	35	-0,0081	126	44703	0,0000	128
12	0,0016	128	36	-0,0088	126	44702	0,0000	128
13	0,0017	128	37	-0,0095	126	44701	0,0000	128
14	0,0016	128	38	-0,0100	126	44700	0,0000	128
15	0,0014	128	39	-0,0103	126	44699	0,0000	128
16	0,0011	128	40	-0,0103	126	44698	0,0000	128
17	0,0006	128	41	-0,0102	126	44697	0,0000	128
18	0,0000	128	42	-0,0101	126	44696	0,0000	128
19	-0,0007	127	43	-0,0100	126	44695	0,0000	128
20	-0,0015	127	44	-0,0100	126	44694	0,0000	128
21	-0,0024	127	45	-0,0099	126	44693	0,0000	128
22	-0,0032	127	46	-0,0096	126	44692	0,0000	128
23	-0,0038	127	47	-0,0093	126	44691	0,0000	128
24	-0,0042	127	48	-0,0089	126	44690	0,0000	128

LAMPIRAN C. Data *Cipheraudio "not-a-dream-whats-happening-to-place.wav"* (1)

Kunci : KRIPTOGRAFI IDEA

Kolom	Data 8 bit	*Cipher*	Kolom	Data 8 bit	*Cipher*	Kolom	Data 8 bit	*Cipher*
1	167	0,3098	25	250	0,9608	⋮	⋮	⋮
2	9	-0,9294	26	155	0,2157	44638	66	-0,482
3	164	0,2863	27	67	-0,4745	44639	165	0,2941
4	228	0,7882	28	215	0,6863	44640	156	0,2235
5	228	0,7882	29	113	-0,1137	44641	231	0,8118
6	166	0,3020	30	135	0,0588	44642	60	-0,529
7	111	-0,1294	31	168	0,3176	44643	183	0,4353
8	222	0,7412	32	186	0,4588	44644	245	0,9216
9	231	0,8118	33	26	-0,7961	44645	192	0,5059
10	60	-0,5294	34	176	0,3804	44646	66	-0,482
11	183	0,4353	35	187	0,4667	44647	165	0,2941
12	245	0,9216	36	96	-0,2471	44648	156	0,2235
13	192	0,5059	37	132	0,0353	44649	231	0,8118
14	66	-0,4824	38	229	0,7961	44650	60	-0,529
15	165	0,2941	39	180	0,4118	44651	183	0,4353
16	156	0,2235	40	84	-0,3412	44652	245	0,9216
17	105	-0,1765	41	30	-0,7647	44653	192	0,5059
18	72	-0,4353	42	52	-0,5922	44654	66	-0,482
19	1	-0,9922	43	65	-0,4902	44655	165	0,2941
20	2	-0,9843	44	115	-0,0980	44656	156	0,2235
21	132	0,0353	45	102	-0,2000	44657	231	0,8118
22	99	-0,2235	46	202	0,5843	44658	60	-0,529
23	182	0,4275	47	69	-0,4588	44659	183	0,4353
24	173	0,3569	48	203	0,5922	44690	245	0,9216

LAMPIRAN D. Data *Cipheraudio "not-a-dream-whats-happening-to-place.wav"* (2)

Kunci : KRIPTOGRAFI IDEC

Kolom	Data 8 bit	*Cipher*	Kolom	Data 8 bit	*Cipher*	Kolom	Data 8 bit	*Cipher*
1	213	0,671	25	17	-0,867	⋮	⋮	⋮
2	12	-0,906	26	105	-0,176	44638	176	0,380
3	107	-0,161	27	12	-0,906	44639	114	-0,106
4	191	0,498	28	106	-0,169	44640	127	-0,004
5	220	0,725	29	60	-0,529	44641	47	-0,631
6	34	-0,733	30	192	0,506	44642	92	-0,278
7	146	0,145	31	90	-0,294	44643	114	-0,106
8	192	0,506	32	44	-0,655	44644	166	0,302
9	47	-0,631	33	93	-0,271	44645	39	-0,694
10	92	-0,278	34	33	-0,741	44646	176	0,380
11	114	-0,106	35	216	0,694	44647	114	-0,106
12	166	0,302	36	16	-0,875	44648	127	-0,004
13	39	-0,694	37	28	-0,780	44649	47	-0,631
14	176	0,380	38	205	0,608	44650	92	-0,278
15	114	-0,106	39	136	0,067	44651	114	-0,106
16	127	-0,004	40	61	-0,522	44652	166	0,302
17	113	-0,114	41	59	-0,537	44653	39	-0,694
18	141	0,106	42	128	0,004	44654	176	0,380
19	45	-0,647	43	153	0,200	44655	114	-0,106
20	59	-0,537	44	72	-0,435	44656	127	-0,004
21	211	0,655	45	7	-0,945	44657	47	-0,631
22	234	0,835	46	43	-0,663	44658	92	-0,278
23	69	-0,459	47	181	0,420	44659	114	-0,106
24	231	0,812	48	189	0,482	44690	166	0,302

LAMPIRAN E. Data *plainaudio "not-a-dream-whats-happening-to-place.wav"*(1)

Kunci (enkripsi) : KRIPTOGRAFI IDEA

Kunci (dekripsi) : KRIPTOGRAFI IDEA

Kolom	*Plainaudio*	Kolom	*Plainaudio*	Kolom	*Plainaudio*	Kolom	*Plainaudio*
1	-0,0039	25	-0,0039	49	-0,0118	⋮	⋮
2	-0,0039	26	-0,0039	50	-0,0118	44638	0,0039
3	-0,0039	27	-0,0039	51	-0,0118	44639	0,0039
4	0,0039	28	-0,0039	52	-0,0118	44640	0,0039
5	0,0039	29	-0,0039	53	-0,0118	44641	0,0039
6	0,0039	30	-0,0039	54	-0,0118	44642	0,0039
7	0,0039	31	-0,0039	55	-0,0118	44643	0,0039
8	-0,0039	32	-0,0039	56	-0,0118	44644	0,0039
9	0,0039	33	-0,0039	57	-0,0118	44645	0,0039
10	0,0039	34	-0,0039	58	-0,0118	44646	0,0039
11	0,0039	35	-0,0118	59	-0,0118	44647	0,0039
12	0,0039	36	-0,0118	60	-0,0118	44648	0,0039
13	0,0039	37	-0,0118	61	-0,0118	44649	0,0039
14	0,0039	38	-0,0118	62	-0,0118	44650	0,0039
15	0,0039	39	-0,0118	63	-0,0118	44651	0,0039
16	0,0039	40	-0,0118	64	-0,0118	44652	0,0039
17	0,0039	41	-0,0118	65	-0,0118	44653	0,0039
18	0,0039	42	-0,0118	66	-0,0118	44654	0,0039
19	-0,0039	43	-0,0118	67	-0,0039	44655	0,0039
20	-0,0039	44	-0,0118	68	-0,0039	44656	0,0039
21	-0,0039	45	-0,0118	69	-0,0039	44657	0,0039
22	-0,0039	46	-0,0118	70	0,0039	44658	0,0039
23	-0,0039	47	-0,0118	71	0,0039	44659	0,0039
24	-0,0039	48	-0,0118	72	0,0117	44690	0,0039

LAMPIRAN F. Data *plainaudio "not-a-dream-whats-happening-to-place.wav"*(2)

Kunci (enkripsi): KRIPTOGRAFI IDEA

Kunci (dekripsi): KRIPTOGRAFI IDEC

Kolom	*Plainaudio*	Kolom	*Plainaudio*	Kolom	*Plainaudio*	Kolom	*Plainaudio*
1	-0,765	25	-0,984	49	-0,247	⋮	⋮
2	0,882	26	0,373	50	0,490	44638	0,663
3	-0,624	27	0,255	51	0,663	44639	-0,380
4	0,043	28	0,914	52	0,451	44640	0,239
5	-0,875	29	0,216	53	-0,537	44641	-0,757
6	-0,122	30	0,812	54	0,522	44642	0,169
7	0,576	31	0,263	55	0,349	44643	0,184
8	-0,020	32	-0,710	56	0,380	44644	-0,788
9	-0,757	33	0,192	57	-0,247	44645	-0,043
10	0,169	34	-0,153	58	0,490	44646	0,663
11	0,184	35	0,090	59	0,663	44647	-0,380
12	-0,788	36	-0,333	60	0,451	44648	0,239
13	-0,043	37	-0,647	61	-0,537	44649	-0,757
14	0,663	38	-0,004	62	0,522	44650	0,169
15	-0,380	39	0,827	63	0,349	44651	0,184
16	0,239	40	-0,937	64	0,380	44652	-0,788
17	0,349	41	-0,247	65	-0,310	44653	-0,043
18	-0,537	42	0,490	66	-0,333	44654	0,663
19	0,875	43	0,663	67	-0,475	44655	-0,380
20	0,765	44	0,451	68	-0,278	44656	0,239
21	-0,192	45	-0,537	69	-0,153	44657	-0,757
22	0,945	46	0,522	70	-0,592	44658	0,169
23	0,639	47	0,349	71	0,239	44659	0,184
24	0,663	48	0,380	72	0,380	44690	-0,788

LAMPIRAN G. Matrik derajat keabuan barbara.png

Ukuran matrik 1024×1024

	1	2	3	4	5	6	7	8	9	10	11	12	…	1024
1	181	185	196	204	203	201	197	193	190	189	192	195	…	92
2	179	183	195	204	203	201	197	193	190	189	192	195	…	93
3	173	178	193	203	203	200	196	191	191	190	192	194	…	95
4	171	177	191	201	200	197	192	188	189	190	191	194	…	97
5	173	178	190	198	196	192	187	185	186	188	190	193	…	97
6	177	182	192	197	194	188	183	181	184	188	191	194	…	100
7	181	185	196	199	195	188	182	180	185	190	192	195	…	103
8	188	191	200	203	197	189	183	181	185	190	193	196	…	106
9	194	197	204	205	198	191	185	183	185	189	192	196	…	106
10	199	201	205	205	197	191	187	186	187	190	193	197	…	107
11	201	202	204	202	195	190	187	188	190	193	195	198	…	108
12	198	199	202	198	191	186	186	190	193	196	196	199	…	107
13	193	195	199	195	187	182	185	190	193	196	196	198	…	104
14	192	192	195	191	183	180	184	190	195	198	197	200	…	103
15	194	194	192	187	182	181	185	191	196	200	200	203	…	103
16	198	196	191	186	183	184	188	193	198	202	204	207	…	109
17	202	199	193	188	186	187	191	196	200	204	206	209	…	116
18	203	200	195	190	190	191	193	198	202	205	207	210	…	128
19	201	199	196	192	192	192	193	197	202	205	206	209	…	137
20	199	197	193	191	192	194	196	199	202	205	206	210	…	147
21	197	195	189	188	190	195	199	202	202	205	208	212	…	151
22	192	190	185	185	189	195	201	204	204	207	210	214	…	150
23	187	186	183	184	188	195	201	205	205	208	212	215	…	144
24	184	183	181	184	190	197	201	205	206	209	214	216	…	136
25	186	185	181	185	191	199	202	205	205	209	215	216	…	128
26	187	186	183	188	195	201	204	205	205	210	215	216	…	120
27	188	187	187	191	199	204	204	205	207	211	216	214	..	117
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱	⋮
1024	96	96	95	96	97	99	99	99	100	101	102	102	…	109

LAMPIRAN H. Matrik derajat keabuan barbara.png tersisipi

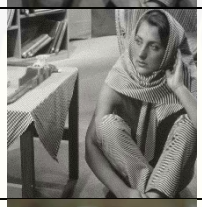
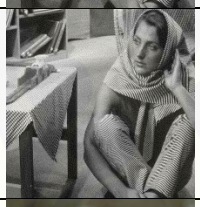
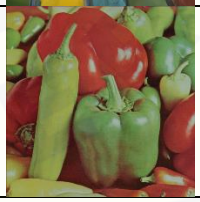
Plainaudio : *not-a-dream-whats-happening-to-place.wav*

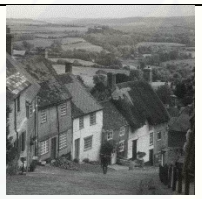
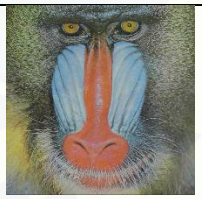
Kunci : KRIPTOGRAFI IDEA

Ukuran matrik 1024×1024

	1	2	3	4	5	6	7	8	9	10	11	12	…	1024
1	181	185	194	206	203	201	198	192	190	189	193	194	⋯	89
2	179	183	193	206	203	201	198	192	190	189	193	194	⋯	90
3	173	178	193	203	203	200	194	193	193	188	190	196	⋯	97
4	171	177	191	201	200	197	190	190	191	188	189	196	⋯	99
5	175	176	191	197	197	191	188	184	184	190	191	192	⋯	98
6	179	180	193	196	195	187	184	180	182	190	192	193	⋯	101
7	182	184	197	198	195	188	184	178	187	188	191	196	⋯	105
8	189	190	201	202	197	189	185	179	187	188	192	197	⋯	108
9	194	197	204	205	195	194	188	180	183	191	193	195	⋯	107
10	199	201	205	205	194	194	190	183	185	192	194	196	⋯	108
11	198	205	206	200	195	190	187	188	190	193	195	198	⋯	108
12	195	202	204	196	191	186	186	190	193	196	196	199	⋯	107
13	190	198	200	194	187	182	187	188	192	197	197	197	⋯	103
14	189	195	196	190	183	180	186	188	194	199	198	199	⋯	102
15	196	192	192	187	184	179	187	189	195	201	198	205	⋯	100
16	200	194	191	186	185	182	190	191	197	203	202	209	⋯	106
17	203	198	191	190	184	189	193	194	199	205	205	210	⋯	116
18	204	199	193	192	188	193	195	196	201	206	206	211	⋯	128
19	201	199	195	193	190	194	192	198	202	205	205	210	⋯	135
20	199	197	192	192	190	196	195	200	202	205	205	211	⋯	145
21	196	196	190	187	190	195	197	204	203	204	210	210	⋯	150
22	191	191	186	184	189	195	199	206	205	206	212	212	⋯	149
23	187	186	184	183	191	192	202	204	206	207	211	216	⋯	145
24	184	183	182	183	193	194	202	204	207	208	213	217	⋯	137
25	186	185	181	185	192	198	200	207	207	207	214	217	⋯	127
26	187	186	183	188	196	200	202	207	207	208	214	217	⋯	119
27	188	187	186	192	201	202	202	207	208	210	215	215	⋯	115
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱	⋮
1024	96	96	95	96	97	99	99	99	100	101	102	102	…	109

LAMPIRAN I. Tabel Uji dengan berbagai Data

No	Data	Citra	Citra Hasil	PSNR	SNR_{dB}
1	Audio: not-a-dream-whats-happening-to-place.wav Durasi:± 4 detik Kunci: KRIPTOGRAFI IDEA Citra: barbara.png (1024 x 1024) DWT: tingkat 1			45,4455	40,7
2	Audio: not-a-dream-whats-happening-to-place.wav Durasi:± 4 detik Kunci: KRIPTOGRAFI IDEA Citra: barbara.png (1024 x 1024) DWT: tingkat 2			42,55	40,7
3	Audio: Recording 1.wav Kunci: KRIPTOGRAFI IDEA Durasi:± 19 detik Citra: barbara.png (1024 x 1024) DWT: tingkat 1			40	35,24
4	Audio: Recording 1.wav Durasi:± 19 detik Kunci: kriptografi idea Citra: parrots.png (1024 x 1024) DWT: tingkat 1			40,98	35,24
5	Audio: Recording 2.wav Durasi:± 26 detik Kunci: kriptografi idea Citra: parrots.png (1024 x 1024) DWT: tingkat 1			42,7178	37,499
6	Audio: Recording 2.wav Durasi:± 26 detik Kunci: kriptografi idea Citra: PeppersRGB.png (512x512) DWT: tingkat 1			41,0168	37,499

No	Data	Citra	Citra Hasil	PSNR	SNR_{dB}
7	Audio: Recording 2.wav Durasi:± 26 detik Kunci: kriptografi idea Citra: PeppersRGB.tif (512x512) DWT: tingkat 1			40,85	37,499
8.	Audio: human_voice.wav Durasi:± 2 detik Kunci: KRIPTOGRAFI IDEA Citra: parrots.png (1024 x 1024) DWT: tingkat 2			50,2193	Inf
9.	Audio: Epic male countdown.wav Durasi:± 13 detik Kunci: kriptografi idea Citra: barbara.png (1024 x 1024) DWT: tingkat 1			40,8364	32,0839
10	Audio: human_voice.wav Durasi:± 2 detik Kunci: KRIPTOGRAFI IDEA Citra: barbara.png (1024 x 1024) DWT: tingkat 3			43,1533	Inf
11.	Audio: not-a-dream-whats-happening-to-place.wav Durasi:± 4 detik Kunci: kriptografi idea Citra: Goldhill.bmp (512x512) DWT: tingkat 1			39,6149	40,7
12.	Audio: music.wav Durasi:± 54 detik Kunci: kriptografi idea Citra: baboonl.bmp (1024x1024) DWT: tingkat 1			43,4823	42,4633

LAMPIRAN J. *Peak Signal to Noise Ratio Source Code*

PSNR.m

```
function hasil=PSNR(datahasil,dataasli)
[m, n]=size(datahasil);
DataH=zeros(m,n);
DataA=zeros(m,n);
DataH(:,:)=datahasil(:,:);
DataA(:,:)=dataasli(:,:);
MSE=0;
for i=1:m
    for j=1:n
    MSE=MSE+(DataH(i,j)-DataA(i,j))^2;
    end
end
hasil=MSE/(m*n);
hasil=10*log10(255^2/hasil);
```

LAMPIRAN K. ***Signal to Noise Ratio Source Code***

```
function SNR=SNR(a,b)
ra=mean(a);
rb=mean(b);
N=length(a);
c=0;
c2=0;
c3=0;
for i=1:N
 c=c+(a(i)-ra)*(b(i)-rb);
 c2=c2+((a(i)-ra))^2;
 c3=c3+((b(i)-rb))^2;
end
hasil=(1/N*c)/(sqrt(1/N*c2)*sqrt(1/N*c3));
SNR=10*log10(hasil/(1-hasil));
```