

Kompresi Citra Dengan Menggabungkan Metode *Discrete Cosine Transform* (DCT) dan Algoritma *Huffman*

Raras Krasmala¹, Arif Budimansyah Purba², U. Tresna Lenggana³

^{1,2,3}Teknik Informatika, STMIK Kharisma Karawang

¹raraskrasmala15@gmail.com, ²arifbudimansyahpurba@gmail.com

³tresna.mobile.pro@gmail.com

Abstrak—Penelitian ini bertujuan untuk mengkompresi citra dengan menggabungkan metode DCT dan Algoritma Huffman untuk membuat kapasitas file gambar menjadi kecil sehingga dapat menghemat media penyimpanan dan tidak lambat jika pengiriman citra dari satu tempat ke tempat lain. Discrete Cosine Transform (DCT) adalah sebuah teknik yang mengubah sinyal ke dalam komponen frekwensi dasar dan Algoritma Huffman adalah algoritma yang digunakan untuk membuat kompresi jenis *lossy compression* yaitu penempatan data dimana tidak ada satu byte pun data yang hilang sehingga data tersebut utuh dan disimpan sesuai dengan aslinya. Dengan menggabungkan metode Discrete Cosine Transform (DCT) dan Algoritma Huffman dapat mengkompresi gambar dengan maksimal. Dengan teknik *lossy compression* pada DCT, kompresi citra yang dihasilkan sedikit mengurangi warna (*pixel*) namun tampak tidak terlihat perbedaannya dengancitra asli sebelum dikompresi. Hasil kompresi tergantung pada pemilihan kualitas kompresi yang diinginkan. Jika memilih kompresi dengan kualitas standar, maka citra hasil kompresi dengan citra yang asli tidak akan terlihat perbedaannya namun pengurangan ukuran bytes tidak terlalu drastis. Tetapi apabila kita memilih kualitas kompresi rendah, maka ukuran *bytes* pada citra akan berkurang namun kualitas gambar hasil kompresi akan terlihat perbedaannya dengan citra asli.

Kata kunci—DCT, Huffman, Lossy Compression

I. PENDAHULUAN

Sejak ditemukannya alat untuk menangkap suatu gambar pada bidang dua dimensi (citra) berupa kamera, dengan semakin berkembangnya teknologi pada saat ini sehingga hal tersebut tidak hanya berfokus pada alat-alat yang digunakan untuk menangkap citra tersebut [6]. Dengan menggunakan kamera digital, semua persyaratan

untuk penyimpanan, manipulasi dan transfer gambar digital dapat dilakukan, sehingga teknologi yang dapat mengelola suatu citra yang telah ditangkap juga merupakan hal yang sangat penting, karena citra yang telah ditangkap oleh kamera tersebut tidak dapat dipastikan akan menghasilkan citra yang baik dan sesuai dengan kebutuhan manusia [3]. Sebagian besar data citra terdiri dari data multimedia dan mereka menempati sebagian besar dari *bandwidth* komunikasi untuk mengembangkan komunikasi multimedia [3].

Masalah pada citra adalah besarnya ruang penyimpanan yang diperlukan karena file-file gambar yang didapat sangatlah besar dan dapat menempati banyak ruang dalam media penyimpanan. Selain itu, data citra berukuran besar jika dikirim melalui jaringan akan membuat pengiriman citra dari satu tempat ke tempat lain menjadi lambat [4]. Oleh karena itu dibutuhkan teknik yang efisien untuk dapat melakukan kompresi citra [3]. Kompresi citra digital merupakan upaya untuk melakukan transformasi terhadap data atau simbol penyusunan citra digital menjadi data atau simbol lain, tanpa menimbulkan perubahan yang terlihat signifikan atas citra digital tersebut bagi mata manusia yang mengamatinya. Tujuannya adalah untuk mengurangi redundansi dari data-data yang terdapat dalam citra sehingga dapat di simpulkan atau ditransmisikan secara efisien. Pada teknik kompresi sendiri dibedakan menjadi dua, yaitu *lossless compression* dan *lossy compression* [1]. “*Lossless compression yaitu teknik yang memproses data asli menjadi bentuk yang lebih ringkas tanpa hilangnya informasi, biasanya digunakan pada aplikasi biomedis. Lossy compression yaitu teknik mendapatkan data yang lebih ringkas dengan melalui suatu proses penghampiran (aproksimasi) dari data asli dengan tingkat error yang dapat diterima*” [1].

Dari permasalahan yang telah diuraikan tersebut diatas, untuk menindaklanjuti permasalahan tersebut maka pada penelitian ini penulis mengangkat judul “Kompresi Citra Dengan Menggabungkan Metode *Discrete Cosine*

Transform (DCT) Dan Algoritma Huffman” dimana DCT adalah sebuah skema *lossy compression* dimana NxN blok ditransformasikan dari domain spasial ke domain DCT [2] begitu pula dengan Algoritma *Huffman* yang sama-sama digunakan untuk membuat kompresi jenis *lossy compression*.

II. METODE PENELITIAN

A. Bahan Penelitian

Bahan penelitian pada penelitian ini diambil dari beberapa jurnal tentang kompresi citra dengan metode *Discrete Cosine Transform (DCT)* dan beberapa jurnal kompresi citra dengan Algoritma *Huffman* yang didapatkan dari berbagai sumber dan *e-book* Satzinger J, Jackson R, dan Stephen Burd tahun 2007 tentang *Analysis System And Design In A Changing World*.

B. Alat Penelitian

1. Kebutuhan Hardware

Untuk memulai tahap penelitian sampai dengan implementasi pada penelitian mengenai kompresi citra ini dibutuhkan alat berupa perangkat keras (*hardware*) yaitu sebuah laptop dengan spesifikasi *Processor 2 MB Cache 2.16 GHz, RAM 2 GB, HDD 500 GB*.

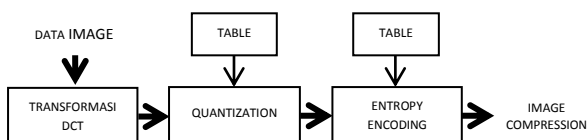
2. Kebutuhan Software

Untuk membangun aplikasi kompresi citra ini diperlukan beberapa perangkat lunak. Perangkat lunak yang digunakan dalam penelitian ini adalah:

- Sistem operasi Ubuntu 14.04 LTS 64 bit
- NetBeans IDE 8.1
- JDK 8.1 (*Java Deploment Kit*)
- StarUML
- Dia Diagram Editor

C. Tahapan Metode DCT

Tahapan-tahapan pada metode DCT ini adalah data *image* diproses mulai dari tahap *preparation process*, kemudian transformasi DCT, *quantization* dan *entropy encoding* kemudian keluar sebagai *image compression* (hasil kompresi). Tahapan proses tersebut digambarkan sebagai berikut :



Gambar 1. Tahapan Proses Metode DCT

1. Transformasi DCT

Semakin tinggi kemampuan mengompresi informasi dalam koefisien yang lebih sedikit maka

semakin baik transformasinya. Oleh karena itu digunakan metode *Discrete Cosine Transform (DCT)*. Pada tahap ini mengubah input data ke dalam format untuk mengurangi redundansi interpixel pada gambar masukan. Teknik perubahan pengkodean menggunakan reversibel, linier matematika transformasi untuk memetakan nilai piksel ke satu set koefisien, yang kemudian dikuantisasi dan dikodekan.

2. Quantization

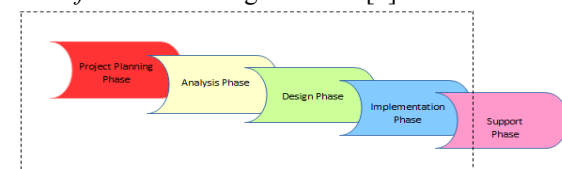
Tahap kuantisasi dilakukan untuk membersihkan koefisien DCT yang tidak penting untuk pembentukan image baru. Dimana frekuensi yang tinggi akan diseleksi untuk dihilangkan yang terikat pada pengaturan kualitas yang digunakan. Hal ini yang menyebabkan JPEG bersifat *lossy*.

3. Entropy Encoding

Entropy encoding yaitu proses penggunaan algoritma entropi, pada tahap ini menggunakan algoritma *huffman* untuk mengkodekan koefisien hasil proses DCT yang akan mengeliminasi nilai-nilai matriks yang bernilai nol dimana akan menghilangkan kelebihan dari keluaran kuantiser secara *zigzag* dan akhirnya akan diperoleh *image* yang telah direkonstruksi (*image* yang sudah dikompres).

D. Metode Pengembangan Sistem SDLC Model Waterfall

Tahapan-tahapan pada pengembangan sistem ini adalah melakukan perencanaan, analisis, desain dan implementasi. Setiap tahapan dikerjakan sampai selesai sebelum mengerjakan tahapan berikutnya. Tahapan pengerjaan dengan model *waterfall* adalah sebagai berikut [5] :



Gambar 2. SDLC Waterfall[5]

1. Project Planning Phase

Tahap perencanaan adalah proses dasar memahami mengapa aplikasi *m-learning* mengenai taksonomi vertebrata harus dibangun dan menentukan bagaimana membangun sistem tersebut. Pada tahapan ini dilakukan penelitian atau riset terlebih dahulu untuk mendapatkan data serta informasi yang terkait. Teknik pengumpulan data yang dilakukan adalah dengan melakukan wawancara dan studi literatur/pustaka baik melalui buku ataupun jurnal.

2. Analysis Phase

- 1) Analisis sistem berjalan dengan menggunakan grafis dan narasi
Hasil dari tahapan ini adalah analisa sistem yang sedang berjalan dalam bentuk grafis dan narasi. Dalam berbentuk grafis penulis menggunakan *flowchart*, sedangkan bentuk penulisan berupa narasi.
- 2) Analisis sistem ajuan dengan menggunakan *Object Oriented Analysis* (OOA)
Hasil dari tahapan ini adalah tujuan perbaikan sistem terhadap masalah serta manfaat yang akan diperoleh. Tahapan dari analisis tersebut yaitu:
 1. *System Activities (Actor Description and Use Case Description, Use Case Diagram, Scenario Use Case)*.
 2. *Class Diagram (Class Definition, Class Relation)*.
 3. *Object Interaction (Sequence Diagram)*.
 4. *Object Behavior (Activity Diagram)*.

3. Design Phase

Tahap desain akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada: struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail algoritma prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistemnya. Dalam tahapan ini desain yang dilakukan oleh peneliti adalah pendesainan berbasis *Object Oriented Design* (OOD) terdiri dari :

1. Desain Proses.
Rancangan logika pemrosesan data yang akan digunakan akan disajikan menggunakan *flow chart*.
2. Desain Antarmuka.
Rancangan tampilan masukan dan keluaran yang akan dioperasikan oleh *user*.

4. Implementation Phase

Tahapan implementasi merupakan tahap pembuatan program termasuk penulisan kode program, pengetesan program secara terus menerus untuk mencari kesalahan yang terjadi pada program yang di buat, kemudian meletakkan sistem untuk dioperasikan.

Penulisan kode program dilakukan dengan teknik *Object Oriented Programming* (OOP) secara modular sehingga akan mempermudah dalam memodifikasi kode yang sudah ada. Objek yang baru dapat dibuat tanpa mengubah kode yang sudah ada. Dapat menyederhanakan kompleksitas dengan memungkinkan *programmer* untuk mendefinisikan sebuah sistem besar dan kompleks menggunakan set yang lebih kecil dari objek yang saling terkait.

Pada Tahapan ini dilakukan beberapa tahapan, antara lain:

1. Instalasi Sistem
Menjelaskan tahapan tahapan dilakukannya proses instalasi aplikasi dalam komputer atau laptop.
2. Pengujian Terhadap Sistem
Pengujian sistem dilakukan dengan dua tahap yaitu pengujian *white box* dan *black box*. Pengujian *white box* dilakukan pada *syntax* dan logika. Sedangkan pengujian *black box* dilakukan untuk menguji apakah setiap fungsi di dalam program dapat berjalan dengan benar.

III. HASIL DAN PEMBAHASAN

A. Metode Discrete Cosine Transform (DCT)

Berikut adalah algoritma *discrete cosine transform*:

1. Langkah pertama adalah citra *original* dibagi menjadi beberapa blok dan masing-masing blok memiliki 8 *pixel* kali 8 *pixel* :



(1)



(2)

Gambar 3. (1) citra *original*; (2) citra *original* dikompakkan menjadi beberapa blok.

Pada gambar diatas dimaksudkan bahwa apabila sebuah gambar diperbesar, maka akan terlihat *pixel (picture element)*. *Pixel* tersebut kemudian dibagi menjadi beberapa blok dan masing-masing blok terdiri dari 8x8 *pixel*.

2. Selanjutnya data matriks *original* akan dikurangi dengan 128, itu karena algoritma DCT bekerja pada rentang -128 sampai 127 sesuai dengan ketentuan pengolahan citra

digital pada citra berwarna. Berikut matriks *original* dari gambar diatas :

$$\begin{bmatrix} 104 & 101 & 99 & 97 & 95 & 95 & 96 & 99 \\ 101 & 96 & 95 & 95 & 96 & 98 & 99 & 99 \\ 99 & 92 & 91 & 92 & 97 & 102 & 103 & 99 \\ 98 & 90 & 89 & 91 & 98 & 104 & 104 & 98 \\ 98 & 90 & 89 & 92 & 97 & 103 & 104 & 98 \\ 99 & 91 & 90 & 92 & 97 & 102 & 103 & 98 \\ 99 & 92 & 91 & 93 & 97 & 102 & 102 & 98 \\ 100 & 93 & 92 & 94 & 98 & 102 & 102 & 98 \end{bmatrix}$$

Gambar 4. Matriks *Original*

Dan berikut adalah matriks *original* yang sudah dikurangi dengan 128.

$$\begin{bmatrix} -24 & -27 & -29 & -31 & -33 & -33 & -32 & -29 \\ -27 & -32 & -33 & -33 & -32 & -30 & -29 & -29 \\ -29 & -36 & -37 & -36 & -31 & -36 & -25 & -29 \\ -30 & -38 & -39 & -37 & -30 & -24 & -24 & -30 \\ -30 & -38 & -39 & -36 & -31 & -25 & -24 & -30 \\ -29 & -37 & -38 & -36 & -31 & -26 & -25 & -30 \\ -29 & -36 & -37 & -35 & -31 & -26 & -26 & -30 \\ -28 & -35 & -36 & -34 & -30 & -26 & -26 & -30 \end{bmatrix}$$

Gambar 5. Matriks M

3. Menghitung nilai matriks DCT untuk matriks T dan matriks *transpose* untuk matriks T' dengan rumus sebagai berikut :

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

Dengan menggunakan rumus matriks diatas dapat dihitung nilai matriks T mulai dari T (0,0) sampai T (7,7) sebagai berikut :

$$\begin{aligned} T(0,0) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,1) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,2) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,3) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,4) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,5) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,6) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(0,7) &= \frac{1}{\sqrt{8}} = \frac{1}{\sqrt{8}} = 0,3536 \\ T(1,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)1.180^0}{2.8} = 0,4904 \\ T(1,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)1.180^0}{2.8} = 0,4157 \\ T(1,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)1.3.14}{2.8} = 0,2778 \\ T(1,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)1.180^0}{2.8} = 0,0975 \\ T(1,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)1.180^0}{2.8} = -0,0975 \\ T(1,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)1.180^0}{2.8} = -0,2778 \\ T(1,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)1.180^0}{2.8} = -0,4157 \\ T(1,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)1.180^0}{2.8} = -0,4904 \end{aligned}$$

$$\begin{aligned} T(2,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)2.180^0}{2.8} = 0,4619 \\ T(2,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)2.180^0}{2.8} = 0,1913 \\ T(2,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)2.180^0}{2.8} = -0,1913 \\ T(2,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)2.180^0}{2.8} = -0,4619 \\ T(2,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)2.180^0}{2.8} = -0,4619 \\ T(2,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)2.180^0}{2.8} = -0,1913 \\ T(2,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)2.180^0}{2.8} = 0,1913 \\ T(2,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)2.180^0}{2.8} = 0,4619 \\ T(3,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)3.180^0}{2.8} = 0,4157 \\ T(3,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)3.180^0}{2.8} = -0,0975 \\ T(3,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)3.180^0}{2.8} = -0,4904 \\ T(3,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)3.180^0}{2.8} = -0,2778 \\ T(3,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)3.180^0}{2.8} = 0,2778 \\ T(3,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)3.180^0}{2.8} = 0,4904 \\ T(3,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)3.180^0}{2.8} = 0,0975 \\ T(3,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)3.180^0}{2.8} = -0,4157 \\ T(4,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)4.180^0}{2.8} = 0,3536 \\ T(4,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)4.180^0}{2.8} = -0,3536 \\ T(4,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)4.180^0}{2.8} = -0,3536 \\ T(4,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)4.180^0}{2.8} = 0,3536 \\ T(4,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)4.180^0}{2.8} = 0,3536 \\ T(4,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)4.180^0}{2.8} = 0,3536 \\ T(4,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)4.180^0}{2.8} = -0,3536 \\ T(4,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)4.180^0}{2.8} = 0,3536 \\ T(5,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)5.180^0}{2.8} = 0,2778 \\ T(5,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)5.180^0}{2.8} = -0,4904 \\ T(5,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)5.180^0}{2.8} = 0,0975 \\ T(5,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)5.180^0}{2.8} = 0,4157 \\ T(5,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)5.180^0}{2.8} = -0,4157 \\ T(5,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)5.180^0}{2.8} = -0,0975 \\ T(5,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)5.180^0}{2.8} = 0,4904 \\ T(5,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)5.180^0}{2.8} = -0,4904 \\ T(6,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)6.180^0}{2.8} = 0,1913 \\ T(6,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)6.180^0}{2.8} = -0,4619 \\ T(6,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)6.180^0}{2.8} = 0,4619 \end{aligned}$$

$$\begin{aligned}
 T(6,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)6.180^0}{2.8} = -0,1913 \\
 T(6,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)6.180^0}{2.8} = -0,1913 \\
 T(6,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)6.180^0}{2.8} = 0,4619 \\
 T(6,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)6.180^0}{2.8} = -0,4619 \\
 T(6,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)6.180^0}{2.8} = 0,1913 \\
 T(7,0) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.0+1)7.180^0}{2.8} = 0,0975 \\
 T(7,1) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.1+1)7.180^0}{2.8} = -0,2778 \\
 T(7,2) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.2+1)7.180^0}{2.8} = 0,4157 \\
 T(7,3) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.3+1)7.180^0}{2.8} = -0,4904 \\
 T(7,4) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.4+1)7.180^0}{2.8} = 0,4904 \\
 T(7,5) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.5+1)7.180^0}{2.8} = -0,4157 \\
 T(7,6) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.6+1)7.180^0}{2.8} = 0,2778 \\
 T(7,7) &= \sqrt{\frac{2}{8}} \cos \frac{(2j+1)i\pi}{2N} = \sqrt{\frac{2}{8}} \cos \frac{(2.7+1)7.180^0}{2.8} = -0,0975
 \end{aligned}$$

Dari perhitungan diatas maka didapatkan nilai untuk matriks T sebagai berikut:

$$\begin{bmatrix}
 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 \\
 0,4904 & 0,4157 & 0,2778 & 0,0975 & -0,0975 & -0,2778 & -0,4157 & -0,4904 \\
 0,4619 & 0,1919 & -0,1913 & -0,4619 & -0,4619 & -0,1913 & 0,1913 & 0,4619 \\
 0,4157 & -0,0975 & -0,4904 & -0,2778 & 0,2778 & 0,4904 & 0,0975 & -0,4157 \\
 0,3536 & -0,3536 & -0,3536 & 0,3536 & 0,3536 & -0,3536 & -0,3536 & 0,3536 \\
 -0,2778 & -0,4904 & 0,0975 & 0,4157 & -0,4157 & -0,0975 & -0,4904 & -0,2778 \\
 0,1913 & -0,4619 & 0,4619 & -0,1913 & -0,1913 & 0,4619 & -0,4619 & 0,1913 \\
 0,0975 & -0,2778 & 0,4157 & -0,4904 & 0,4904 & -0,4157 & 0,2778 & -0,0975
 \end{bmatrix}$$

Gambar 3.4 Matriks T

$$\begin{bmatrix}
 0,3536 & 0,4904 & 0,4619 & 0,4157 & 0,3536 & -0,2778 & 0,1913 & 0,0975 \\
 0,3536 & 0,4157 & 0,1919 & -0,0975 & -0,3536 & -0,4904 & -0,4619 & -0,2778 \\
 0,3536 & 0,2778 & -0,1913 & -0,4904 & -0,3536 & 0,0975 & 0,4619 & 0,4157 \\
 0,3536 & 0,0975 & -0,4619 & -0,2778 & 0,3536 & 0,4157 & -0,1913 & -0,4904 \\
 0,3536 & -0,0975 & -0,4619 & 0,2778 & 0,3536 & -0,4157 & -0,1913 & 0,4904 \\
 0,3536 & -0,2778 & -0,1913 & 0,4904 & -0,3536 & -0,0975 & 0,4619 & -0,4157 \\
 0,3536 & -0,4157 & 0,1913 & 0,0975 & -0,3536 & -0,4904 & -0,4619 & 0,2778 \\
 0,3536 & -0,4904 & 0,4619 & -0,4157 & 0,3536 & -0,2778 & 0,1913 & -0,0975
 \end{bmatrix}$$

Gambar 3.5 Matriks T^t

- Berikutnya adalah tahap menghitung matriks D, dimana matriks D akan digunakan untuk kuantisasi lanjutan.
 $D = T.M.T^t$

$$\begin{bmatrix}
 -248 & -16,1592 & 11,3996 & 19,2935 & 0,5 & 6,1245 & 2,0431 & 1,5029 \\
 2,2777 & 8,2394 & 1,8936 & -6,5075 & 0,8284 & -1,8108 & 0,0814 & -0,2827 \\
 3,9989 & 11,1516 & 0,6036 & -7,3684 & 1,5772 & -1,8480 & 0,1036 & -0,8600 \\
 0,2381 & 4,3461 & 0,8372 & -2,9047 & 0,6091 & -0,2061 & -0,2008 & -0,4762 \\
 1 & 1,3390 & -0,3266 & -0,1420 & 0 & -0,1829 & -0,1353 & -0,1688 \\
 0,0710 & 0,1907 & 0,1665 & 0,3974 & -0,4070 & -0,5022 & 0,2587 & 0,0355 \\
 0,5084 & 0,2214 & 0,6036 & 0,1661 & -0,1121 & -0,0122 & -0,1036 & 0,3436 \\
 0,0179 & -0,1792 & 0,2387 & -0,3785 & -0,1648 & 0,2781 & -0,1218 & -0,3325
 \end{bmatrix}$$

Gambar 6. Matriks D

- Matriks D telah terisi dengan koefisien DCT, dimana data yang terletak pada kiri atas merupakan korelasi dari frekuensi-frekuensi rendah dari data *original*. Sedangkan yang terletak pada kanan bawah merupakan korelasi dari frekuensi-frekuensi tinggi dari data *original*. Selanjutnya adalah tahap kuantisasi dengan menggunakan aturan matriks kuantisasi kualitas 50 (kualitas) standar sebagai berikut :

$$Q = \begin{bmatrix}
 16 & 11 & 10 & 16 & 24 & 51 & 51 & 61 \\
 12 & 12 & 14 & 19 & 26 & 60 & 60 & 55 \\
 14 & 13 & 16 & 24 & 40 & 69 & 69 & 56 \\
 14 & 17 & 22 & 29 & 51 & 80 & 80 & 62 \\
 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\
 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\
 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\
 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99
 \end{bmatrix}$$

Gambar 7. Matriks Kuantisasi (Q)

Persamaan matriks kuantisasi adalah sebagai berikut :

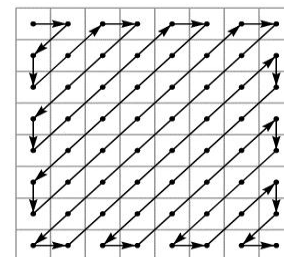
$$C_{i,j} = \text{round} \frac{D_{ij}}{Q_{ij}}$$

Dimana *round* berarti mendekati nilai hasil pembagian ke pembulatan bilangan integer terdekat.

$$C = \begin{bmatrix}
 -5 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}$$

Gambar 8. Matriks C

- Menyusun bilangan menggunakan fungsi *zig-zag scanning* sebagai langkah terakhir pada proses kompresi.



Gambar 9. Zig-zag Scanning Kuantisasi

Matriks C yang terkuantisasi akan dikonversi oleh *encoder* ke data biner maka proses kompresi dapat dilakukan (menggunakan algoritma *Huffman*).

- Terakhir adalah proses dekompresi dimana ini merupakan proses untuk merekonstruksikan data hasil kompresi menjadi data yang dapat dikenali. Dan berikut adalah persamaan untuk matriks R :

$$\begin{aligned}
 R_{i,j} &= Q_{i,j} \times C_{i,j} \rightarrow N \\
 &= \text{round} (T'.R.T) + 128
 \end{aligned}$$

$$N = \begin{bmatrix}
 106 & 101 & 97 & 96 & 97 & 98 & 96 & 93 \\
 104 & 100 & 96 & 95 & 98 & 99 & 98 & 96 \\
 100 & 97 & 94 & 94 & 98 & 101 & 100 & 99 \\
 98 & 94 & 92 & 94 & 99 & 103 & 103 & 102 \\
 97 & 94 & 91 & 94 & 99 & 103 & 104 & 103 \\
 97 & 94 & 92 & 94 & 99 & 103 & 103 & 102 \\
 99 & 95 & 93 & 94 & 99 & 102 & 102 & 100 \\
 100 & 97 & 94 & 94 & 98 & 101 & 101 & 99
 \end{bmatrix}$$

Gambar 10. Matriks N

kompresi citra dengan citra sebelum dikompresi akan ditampilkan oleh sistem.

7. Simpan Citra

Setelah seluruh proses kompresi selesai pada tahap ini sistem akan otomatis menampilkan dan menyimpan gambar hasil kompresi pada file dimana file gambar asli sebelum dikompresi tersimpan.

8. Selesai

Proses selesai setelah sistem berhasil menampilkan gambar dan otomatis menyimpan gambar pada direktori komputer, itu adalah tahap akhir dari proses kompresi citra.

D. Analisis Sistem Ajuan

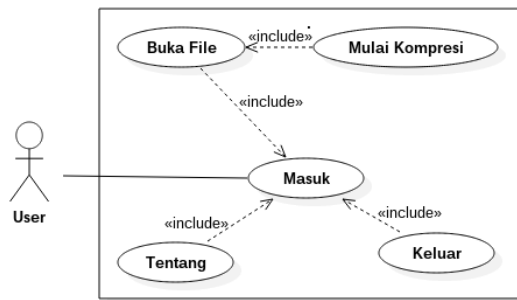
1. System Activities

a. Deskripsi Aktor

Tabel 3.1 Deskripsi Aktor

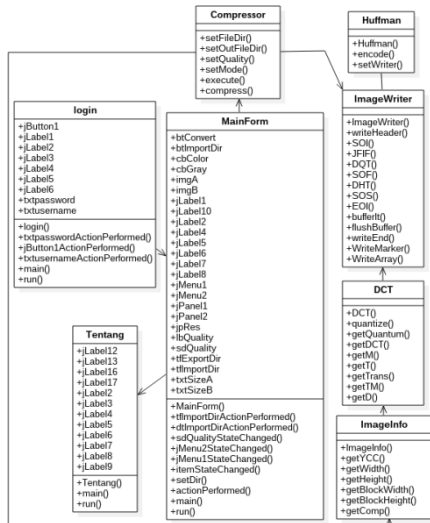
No	Aktor	Deskripsi
1	User	Pengguna sistem dapat melakukan semua proses yang ada pada sistem untuk dapat mengkompresi citra.

3. Use CaseDiagram



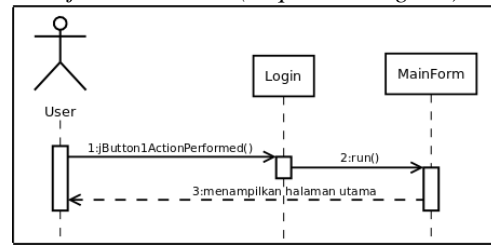
Gambar 14. Use Case Diagram

2. Class Diagram

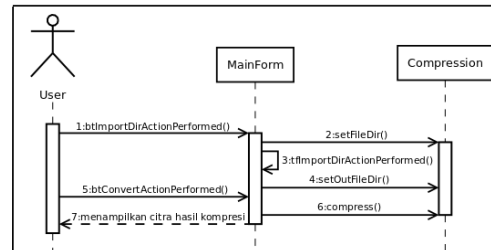


Gambar 15. Class Diagram

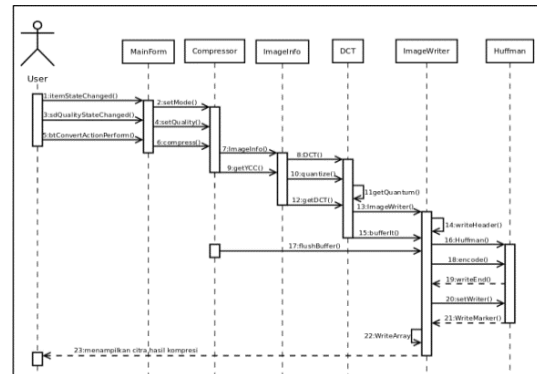
3. Object Interaction (Sequence Diagram)



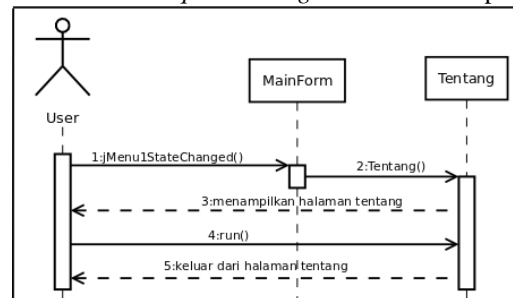
Gambar 16. Sequence Diagram Masuk



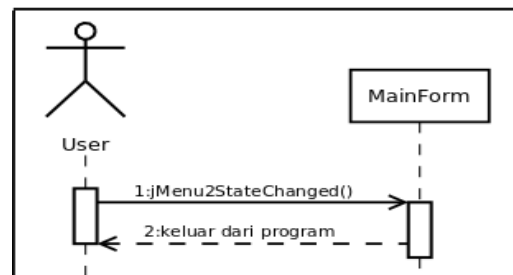
Gambar 17. Sequence Diagram Buka File



Gambar 18. Sequence Diagram Mulai Kompresi

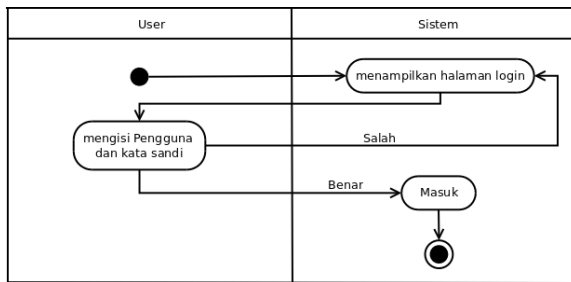


Gambar 19. Sequence Diagram Tentang

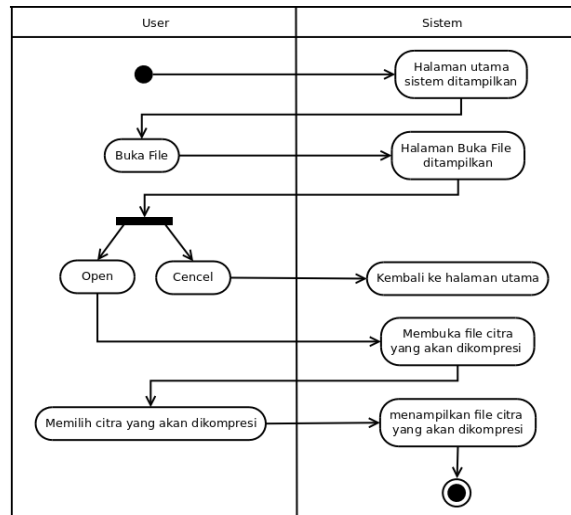


Gambar 20. Sequence Diagram Keluar

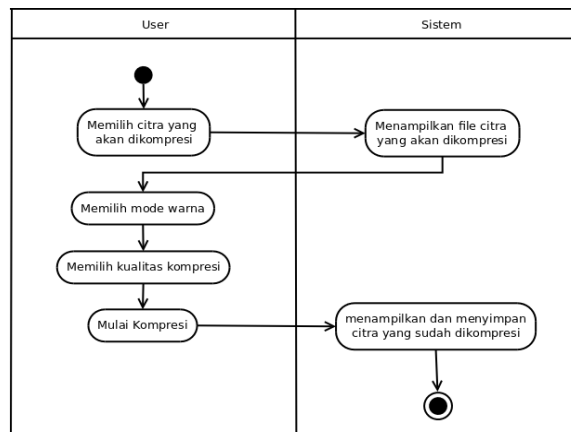
4. Object Behavior (Activity Diagram)



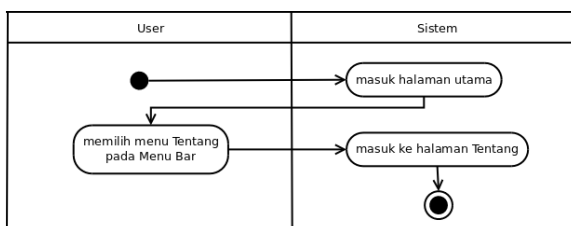
Gambar 21. Activity Diagram Masuk



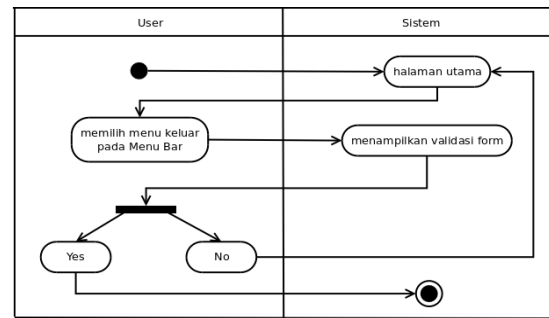
Gambar 22. Activity Diagram Buka File



Gambar 23. Activity Diagram Mulai Kompresi



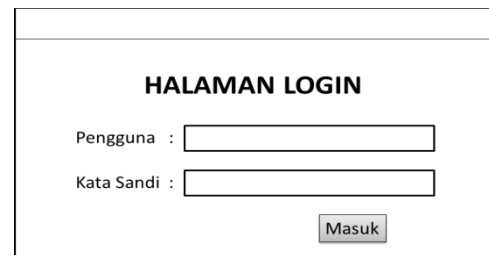
Gambar 24. Activity Diagram Tentang



Gambar 25. Activity Diagram Keluar

C. Desain Antar Muka

1. Desain Antar Muka Halaman Masuk



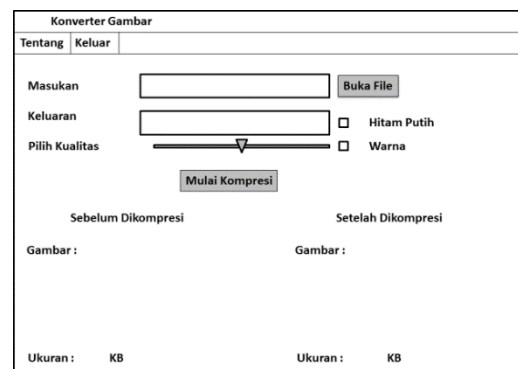
Gambar 26. Desain Antar Muka Halaman Masuk

2. Tampilan Halaman Masuk



Gambar 27. Tampilan Halaman Masuk

3. Desain Antar Muka Halaman Utama



Gambar 28. Desain Antar Muka Halaman Utama

4. Tampilan Halaman Utama



Gambar 29. Tampilan Halaman Utama

IV. SIMPULAN DAN SARAN

A. Simpulan

Dari hasil pembangunan aplikasi kompresi citra ini maka dapat diambil beberapa kesimpulan, yaitu :

1. Dengan menggabungkan metode Discrete Cosine Transform (DCT) dan Algoritma *Huffman* dapat mengkompresi gambar dengan maksimal. Dengan teknik *Lossy Compression* kompresi citra yang dihasilkan sedikit mengurangi warna namun tampak tidak terlihat perbedaannya dengan citra asli sebelum dikompresi.
2. Hasil kompresi citra dapat dianalisa dengan membandingkan citra asli dengan hasil citra yang telah dikompresi.
3. Hasil kompresi tergantung pada pemilihan kualitas kompresi yang diinginkan. Jika kita memilih kompresi dengan kualitas standar, maka citra hasil kompresi dengan citra yang asli tidak akan terlihat perbedaannya namun pengurangan ukuran *bytes* tidak terlalu drastis. Tetapi apabila kita memilih kualitas kompresi

rendah, maka ukuran *bytes* pada citra akan berkurang namun kualitas gambar hasil kompresi akan terlihat perbedaannya dengan citra asli.

B. Saran

Pada penelitian ini penulis menyadari bahwa masih banyak kekurangan. Apabila aplikasi ini akan dikembangkan maka dapat ditambahkan fungsinya sebagai aplikasi *photo editor* seperti menambahkan *effect, border, draw, text, sticker* dan lain-lain.

DAFTAR PUSTAKA

- [1] Herdiyeni, Yeni. 2007. *Kompresi Citra*. Bogor: Departemen Ilmu Komputer IPB.
- [2] Hidayat, Erwin Yudi., Erika Devi Udayanti. 2011. *Hybrid Watermarking Citra Digital Menggunakan Teknik DWT-DCT SVD*. Semarang : in *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*.
- [3] Juma'in., Yuliana Melita. 2011. *Kompresi Gambar atau Citra Menggunakan Discrete Cosin Transform*. Jurnal Teknika : Vol. 3 No. 2. ISSN No. 2085-0859.
- [4] Rafsyam, Yenniwarti. 2010. *Kompresi Citra Menggunakan Teknik Lossy Dengan Metode Algoritma JPEG*. Elektron : Vol. 2 No. 1. ISSN No. 2085-6989.
- [5] Satzinger J, Jackson R, dan Stephen Burd. 2007. *Analysis System And Design In A Changing Wordl*. Canada: Thompson Course Technology.
- [6] Yahya, Kurnia., Yuliana Melita. 2011. *Aplikasi Kompresi Citra Digital Menggunakan Teknik Kompresi Jpeg dengan Fungsi GUI pada Matlab*. Jurnal Teknika : Vol. 3 No. 2. ISSN No. 2085-0859