

Aplikasi Kompresi Citra Dengan MatlabR2015a Menggunakan Metode *Discrete Cosine Transform*(DCT) dan Kuantisasi

Julian dan Aqwam Rosadi Kardian

Sistem Informasi, STMIK Jakarta STI&K

Jl. BRI No. 17 Radio Dalam, Kebayoran Baru, Jakarta Selatan

ajulian@gmail.com, aqwam@jak-stik.ac.id

Abstrak

Kompresi gambar adalah salah satu proses yang bisa mengurangi ukuran gambar agar bisa pada proses menyimpan gambar dan mengirimkannya. Dalam penelitian ini digunakan *Image Compression JPEG Method Standard* untuk mengompres ukuran gambar. Proses Kompresi Metode JPEG melibatkan banyak metode dalam metode ini, seperti Quantization, *Discrete Cosine Transform* (DCT) dan beberapa *Entropy Encoding*. Metode Kompresi ini berbasis kompresi lossy, karena ketika proses kompresi sedang berjalan, ada beberapa informasi dalam gambar yang hilang. Aplikasi ini dibuat menggunakan pemrograman matlab R2015a. Media untuk membuat aplikasi kompresi image menggunakan GUI yang terdapat pada matlab Jenis citra yang dipakai sebagai uji coba adalah citra berwarna (*True Color*) dan berformat PNG dan BMP. Digunakan rasio pemampatan untuk perbandingan citra yang terkompresi dan yang belum terkompresi, dimana hal tersebut akan menunjukkan berapa besar citra yang berhasil termampatkan.

Kata Kunci : : *Aplikasi, kompresi citra, DCT dan Kuantisasi, Matlab*

Pendahuluan

Sejak ditemukannya alat-alat untuk menangkap suatu citra seperti camera, perkembangan teknologi tidak hanya terfokuskan pada peralatan yang digunakan untuk menangkap citra tersebut. Namun teknologi untuk mengolah suatu citra yang telah ditangkap juga merupakan hal yang sangat penting, karena citra yang dihasilkan oleh kamera belum tentu baik sesuai dengan kebutuhan manusia. Dalam pengiriman data, terkadang mengalami kesulitan terhadap ukuran file atau data yang dikirim, data citra yang dikirim memiliki ukuran yang terlalu besar sehingga terkadang hal tersebut menjadi kendala utama. Pengiriman data gagal dikarenakan server storage dari web yang disediakan tidak memuat ukuran file citra digital yang dikirim.

Duplikasi data ini tidak perlu disimpan berulang kali karena memboroskan penggunaan memori. Untuk itu penerapan kompresi terhadap citra diperlukan dengan tujuan meminimalkan kebutuhan memori untuk merepre-

sentasikan citra digital. Teknik dan algoritma kompresi dapat digunakan untuk menciptakan file duplikat dari citra yang akan di kompres sehingga ukurannya lebih kecil dari file aslinya. File tersebut kemudian dapat ditampilkan tanpa terlihat berkurang kualitas aslinya, dikarenakan indra manusia dapat hanya dapat menangkap citra dalam kualitas tertentu. Penerapan Kompresi citra dengan metode kompresi JPEG dilakukan dengan Matlab R2015a, dimana didalamnya terdapat *Discrete Cosine Transform* (DCT), Kuantisasi. Matlab memudahkan dalam kompresi citra, mendukung cara kerja metode kompresi JPEG, pada matlab terdapat perintah dan command yang mendukung dalam pengkompresian. Karena dalam memproses citra atau image dalam penerapannya kebanyakan berbasis matrix $M \times N$ [3].

Ditemukan beberapa kondisi, dimana citra digital dalam komputer perlu di re-size dalam ukuran yang lebih kecil dikarenakan:

1. Apakah kompresi dapat dilakukan den-

gan metode kompresi JPEG sehingga size citra menjadi lebih kecil dari citra aslinya.

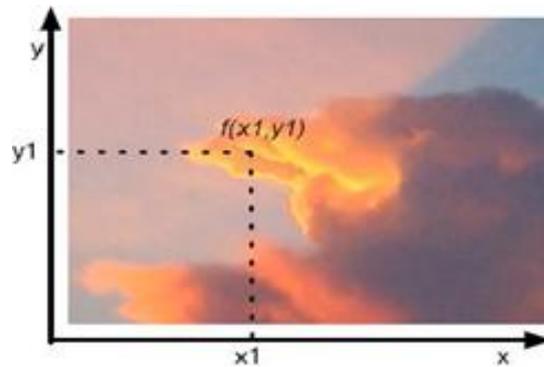
2. Bagaimana tingkat rasio file hasil kompresi citra yang dihasilkan dengan metode *lossy*.
3. Bagaimana hasil dari MSE dan PSNR file citra setelah terkompresi dan sebelum terkompresi.
4. Bagaimana penerapan metode kompresi JPEG pada citra.

Adapun yang menjadi tujuan penelitian ini adalah:

1. Melakukan kompresi citra dengan menggunakan aplikasi GUI dari matlab untuk merubah detail dan warna pada sebuah file citra digital sehingga menghasilkan citra yang memiliki ukuran file kecil, dan memiliki skala kebaikan citra yang baik.
2. Menerapkan metode kompresi JPEG standard untuk pemampatan citra digital.
3. Menjabarkan secara garis besar proses kompresi JPEG dengan singkat dan jelas.

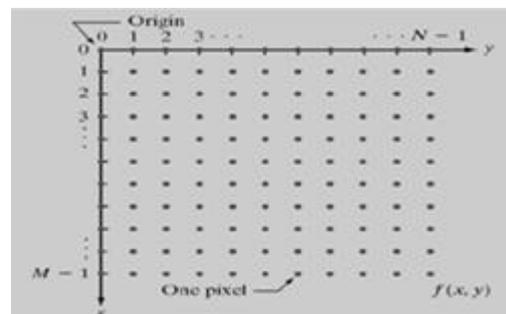
Sedangkan manfaat pada penelitian ini nantinya akan menurunkan size pada file citra digital, sehingga dapat menghemat penyimpanan memori pada komputer dan dalam pengiriman data melalui internet dapat menghemat waktu.

Citra digital adalah gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses digitalisasi. Citra yang terlihat merupakan cahaya yang direfleksikan dari sebuah objek. Sumber cahaya menerangi objek lalu objek memantulkan kembali sebagian dari berkas cahaya tersebut dan pantulan cahaya ditangkap oleh alat-alat optik, missal mata manusia, kamera, scanner, sensor satelit, dan sebagainya, kemudian direkam. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal video seperti gambar pada monitor televisi, atau bersifat digital juga dapat dikelompokkan menjadi dua yaitu: citra tampak seperti foto/gambar, lukisan, dan yang tampak di layar monitor/televisi, hologram serta citra tidak tampak seperti data foto, gambar dalam file, dan citra yang direpresentasikan dalam fungsi matematis.[2]



Gambar 1: Citra Digital

Setiap pixel memiliki nilai (value atau number) yang menunjukkan intensitaskeabuan pada pixel tersebut. Derajat keabuan dimana merepresentasikan grey levelatau kode warna. Kisaran nilai ditentukan oleh bit yang dipakai dan akan menunjukkan resolusi aras abu-abu (grey level resolution). 1 bit –2 warna: [0,1] 4 bit –16 warna: [0,15] 8 bit –256 warna: [0,255] 24 bit –16.777.216 warna (true color) Kanal Merah -Red (R): [0,255] Kanal Hijau -Green (G): [0,255] Kanal Biru- Blue (B): [0,255] Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat(x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x,y, dan amplitudo f secara keseluruhan berhingga (finite) dan bernilai diskritmaka dapat dikatakan bahwa citra tersebut adalah citra digital. Pada gambar 2, menunjukkan posisi koordinat digital.



Gambar 2: . Koordinat Citra Digital

Nilai Digital dan banyak bit :

M = banyak pixel per baris (panjang)

N = banyak pixel per kolom (lebar)

b = banyak / besar bit pada suatu citra

Jenis Citra Digital

Pada suatu citra, citra memiliki ragam tersendiri, dan berbagai macam bentuk citra, antara lain :

1. Citra Biner Citra biner adalah citra yang setiap pixelnya hanya bernilai 0 (warna hitam) dan 1(warna putih).
2. Citra Skala Keabuan (*Grayscale*). Citra skala keabuan adalah citra yang setiap pixel mempunyai kemungkinan warna antara hitam(minimal) dan putih(maksimal).
3. Citra Warna(*True Color*) Citra warna adalah citra yang setiap pikselnya memiliki warna yang merupakan kombinasi antara tiga warna dasar, yaitu merah, biru, dan hijau(RGB) [5]

Format Citra Digital

Citra memiliki ekstensi atau file format tersendiri, dalam formatcitra, masing-masing citra memiliki keunggulannya, beberapa contoh format file citra pada.

1. Format Citra BMP (Bitmap) Merupakan format gambar yang paling umum dan merupakan format standardwindows. Ukuran filenya sangat besar karena mencapai ukuran megabytes. File ini merupakan format yang belum terkompresi dan menggunakan sistem warna RGB (Red, Green, Blue) di mana setiap warna pixelnya terdiri dari 3 komponen R,G, dan B yang dicampur menjadi satu. FileBMP sangat jarang digunakan di web (internet) karena ukurannya besar.
2. JPEG (Joint Photographic Expert Group) Format JPEG merupakan format yang paling terkenal sampai sekarang ini. Hal ini karena sifatnya yang berukuran kecil, dan bersifat portable. File ini sering digunakan pada bidang fotografi untuk menyimpan file foto. File ini bisa digunakan di web (internet).
3. GIF(Graphic Interchange Format) Format file GIF (Graphics Interchange Format) menggunakan maksimal 8 bit warna ($2^8 = 256$ warna) gambar ini menggunakan kompresi dengan LZW compression yang merupakan kompresi loseless yang berarti tidak ada data yang

dibuang. Format ini juga mendukung gambar transparansi dan animasi.

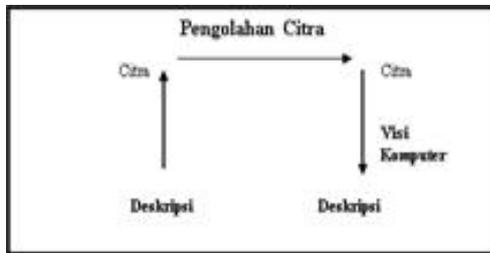
4. PDF (Portable Document Format) Format file PDF (Portable Document Format) digunakan untuk keperluan dokumen lintas sistem dan lintas program aplikasi dan mempunyai ukuran file relatif kecil.

Pengolahan Citra

Pengolahan citra adalah pemrosesan citra khususnya dengan menggunakan komputer menjadi citra yang kualitasnya lebih baik. Umumnya operasi pada pengolahan citra diterapkan pada citra bila perbaikan atau modifikasi citra perlu dilakukan untuk meningkatkan kualitas penampakan atau untuk menonjolkan beberapa aspek informasi yang terkandung di dalam citra, kemudian elemen di dalam citra perlu dikelompokkan, dicocokkan, atau diukur, sebagian citra perlu digabung dengan bagian citra yang lain. Sebelum melakukan pengolahan citra, citra yang akan diolah perlu dilakukan image pre-processing. Salah satu image pre-processing adalah mengubah citra berwarna menjadi citra keabuan.

Pada dasarnya ada tiga bidang yang menangani pengolahan data berbentuk citra, yaitu: grafika komputer, pengolahan citra, dan visi komputer. Bidang grafika komputer banyak dilakukan proses yang bersifat sintesis yang mempunyai ciri data masukan berbentuk deskriptif dengan keluaran hasil proses yang berbentuk citra. Sedangkan proses di dalam bidang visi komputer merupakan kebalikan dari proses grafika komputer. Bidang pengolahan citra merupakan proses pengolahan dan analisis citra dengan data masukan maupun data keluarannya berbentuk citra. Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Pengolahan citra bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin(dalam hal ini komputer) [4].

Teknik pengolahan citra transformasikancitra menjadi citra lain. Masukan adalah citra dan keluaran juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan. Hubungan antara ketiga bidang tersebut ditunjukkan pada gambar 3.



Gambar 3: Bidang berkaitan citra

Tujuan dan manfaat penelitian ini adalah melakukan kompresi citra dengan menggunakan aplikasi GUI dari matlab untuk merubah detail dan warna pada sebuah file citra digital sehingga menghasilkan citra yang memiliki ukuran file kecil, dan memiliki skala kebaikan citra yang baik, pada metode kompresi jpeg-standard untuk pemampatan citra digital.

Penelitian ini bertujuan juga menurunkan size pada file citra digital, sehingga dapat menghemat penyimpanan memori pada komputer dan dalam pengiriman data melalui internet sehingga dapat menghemat waktu.

Metode

Operasi Pengolahan Citra

Operasi yang dilakukan dalam pengolahan citra banyak ragam. Namun, secara umum, pada pengolahan citra terdapat enam jenis operasi pengolahan, yaitu:

1. Peningkatan kualitas citra (*image enhancement*) Jenis operasi ini bertujuan untuk memperbaiki kualitas citra dengan cara memanipulasi parameter citra. Operasi peningkatan kualitas citra:
 - (a) Perbaikan kontras gelap/terang
 - (b) Perbaikan tepi objek (*edge enhancement*)
 - (c) Penajaman (*sharpening*)
 - (d) Pemberian warna semu (*pseudo coloring*)
 - (e) Penapisan derau (*noise filtering*)
2. Restorasi citra (*image restoration*) Operasi ini bertujuan meminimumkan cacat pada citra. Tujuan restorasi citra hampir sama dengan operasi peningkatan kualitas citra. Bedanya, pada restorasi citra penyebab degradasi gambar diketahui. Operasi restorasi citra:

- (a) Penghilangan kesamaran (*deblurring*).
- (b) Penghilangan derau (*noise*).

3. Kompresi citra (*image compression*) Jenis operasi ini dilakukan agar citra dapat direpresentasikan dalam bentuk yang lebih kompak sehingga memerlukan memori yang lebih sedikit. Hal penting yang harus diperhatikan dalam kompresi citra adalah citra yang telah dikompresikan harus tetap mempunyai kualitas gambar yang bagus.
4. Segmentasi citra (*image segmentation*) Operasi ini adalah suatu tahap pada proses analisis citra yang bertujuan untuk memperoleh informasi yang ada di dalam citra tersebut dengan membagi citra ke dalam daerah terpisah dimana setiap daerah adalah homogen dan mengacu pada sebuah kriteria keseragaman yang jelas.
5. Analisis citra (*image analysis*) Jenis operasi ini bertujuan menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsinya. Proses segmentasi diperlukan untuk melokalisasi objek yang diinginkan. Operasi analisis citra, diantaranya [4]:

- (a) Pendeteksian tepi objek (*edge detection*).
- (b) Ekstraksi batas (*boundary*).
- (c) Representasi daerah (*region*).
- (d) Rekonstruksi citra.

Klasifikasi Teknik Kompresi Citra

Kompresi citra (*images compression*) mempunyai tujuan meminimalkan kebutuhan memori untuk merepresentasikan sebuah citra digital. Prinsip umum yang digunakan pada proses kompresi citra digital adalah mengurangi duplikasi data didalam citra sehingga memori yang dibutuhkan untuk merepresentasikan citra menjadi lebih sedikit dari pada citra digital aslinya. Terdapat dua proses utama dalam permasalahan kompresi citra digital.

Lossless Compression

Pada kompresi jenis ini informasi yang terkandung pada citra hasil sama dengan informasi pada citra asli. Citra hasil proses kompresi dapat dikembalikan secara sempurna menjadi citra asli, tidak terjadi kehilangan informasi, dan tidak terjadi kesalahan informasi. Oleh karena itu metode ini disebut juga *error free compression*. Pada kompresi *lossless*, karena harus mempertahankan kesempurnaan informasi, sehingga hanya terdapat proses *coding* dan *decoding*, tidak terdapat proses kuantisasi.

Kompresi *lossless* cocok diterapkan pada berkas basis data (*database*), *spread sheet*, berkas *word processing*, citra biomedis. Namun ratio kompresi (Rasio kompresi yaitu, ukuran file yang dikompresi dibanding yang tak terkompresi dari file) dengan metode ini sangat rendah. Pada gambar 4 contoh citra dengan *Lossless Compression*.



Gambar 4: Citra dengan Lossless

Lossy Compression

Lossy kompresi adalah suatu metode untuk mengkompresi data dan mendekompresinya, data yang diperoleh mungkin berbeda dari yang aslinya tetapi cukup dekat perbedaannya. *Lossy* kompresi ini paling sering digunakan untuk kompres data multimedia (audio, gambar diam). Format kompresi *lossy* mengalami generation loss yaitu jika melakukan berulang kali kompresi dan dekompresi file akan menyebabkan kehilangan kualitas secara progresif. hal ini berbeda dengan kompresi data *lossless*. ketika pengguna yang menerima file terkompresi secara *lossy* (misalnya untuk mengurangi waktu *download*) file yang diambil dapat sedikit berbeda dari yang asli dilevel bit ketika tidak dapat dibedakan oleh mata dan telinga manusia untuk tujuan paling praktis. Metode kompresi *lossy* menghasilkan rasio kompresi yang lebih besar daripada metode

lossless. Misal terdapat *image* asli berukuran 12,249 bytes, kemudian dilakukan kompresi dengan JPEG kualitas 30 dan berukuran 1,869 bytes berarti *image* tersebut 85% lebih kecil dan ratio kompresi 15%. Contoh metode *lossy* adalah metode CS&Q (*coarser sampling and/or quantization*), jpeg, dan mpeg, citra dengan *Lossy Compression*.



Gambar 5: Citra dengan *Lossy Compression*

Kompresi Citra

Proses kompresi merupakan proses mereduksi ukuran suatu data untuk menghasilkan representasi citra digital yang padat atau mampat namun tetap dapat mewakili kuantitas informasi yang terkandung pada data tersebut. Pada citra, video, dan audio, kompresi mengarah pada minimisasi jumlah bit rate untuk representasi digital. Pada beberapa literatur, istilah kompresi sering disebut juga *source coding*, *data compression*, *bandwidth compression*, dan *signal compression*.

Data dan informasi adalah dua hal yang berbeda. Pada data terkandung suatu informasi. Tujuan dari kompresi data tiada lain adalah untuk mengurangi data berlebihan tersebut sehingga ukuran data menjadi lebih kecil dan lebih ringan dalam proses transmisi.

Jenis Pendekatan Kompresi Citra

Pendekatan kompresi citra dalam merupakan strategi atau teknik untuk melakukan kompresi citra, dalam pendekatannya, adalah :

1. Pendekatan Statistik Kompresi atau pemampatan citra didasarkan pada frekuensi kemunculan derajat keabuan pixel di dalam seluruh bagian gambar. Contoh metode: *Huffman Coding*.
2. Pendekatan Ruang Kompresi atau pemampatan citra didasarkan pada hubungan spasial antara pixel-pixel di dalam suatu kelompok yang memiliki derajat keabuan yang sama di dalam suatu daerah

di dalam gambar. Contoh: *Run-Length Encoding*.

3. Pendekatan Kuantisasi Pemampatan citra yang dilakukan dengan mengurangi jumlah derajat keabuan yang ada. Contoh :Metode pemampatan kuantisasi
4. Pendekatan Fraktal Pemampatan citra yang didasarkan pada kenyataan bahwa kemiripan bagian-bagian di dalam citra dapat dieksploitasi dengan suatu matriks transformasi. Contoh :Fractal *Image Compression*.

Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) atau disebut dengan Transformasi Cosinus Diskrit adalah model transformasi fourier yang mengacu pada fungsi diskrit dengan mengambil bagian *cosines* dari eksponensial kompleks, dan hasilnya juga diskrit. *Discrete Cosine Transform* (DCT) 1D (satu dimensi) didefinisikan sebagai berikut : [6]

$$C(u) = \sqrt{\frac{2}{N}} \alpha(u) \sum_{x=0}^{N-1} f(x) \cos(\pi(2x+1)u/N) \quad (1)$$

untuk $u = 0, 1, 2, \dots, N-1$

Dengan cara yang sama, DCT balik dapat didefinisikan sebagai berikut :

$$F(x) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} \alpha(u) C(u) \cos\left(\frac{\pi(2x+1)u}{2N}\right) \quad (2)$$

Dengan $\alpha(u)$ dinyatakan sebagai berikut.

$$\alpha(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{untuk } u=0 \\ 1 & \text{untuk } u \neq 0 \end{cases} \quad (3)$$

Format *image* jpeg menggunakan DCT untuk mengolah tiap 8x8 blok pixel data. Untuk *image* dengan banyak komponen warna (YUV misalnya), maka DCT diterapkan pada 8x8 pixel untuk setiap komponen.[7]

Persamaan dasar 2D DCT sebagai berikut :

$$f(x, y) = \frac{2}{\sqrt{MN}} \alpha(u) \alpha(v) C(x, y) \quad (4)$$

$$\sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \cos\left[\frac{\pi(2x+1)u}{2M}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (5)$$

Persamaan dasar 2D DCT untuk 8x8 standar JPEG didefinisikan $M = N = 8$, dimana M dan N adalah baris dan Kolom matriks pada citra yang sudah di bentuk ke dalam blok 8x8.

Perancangan Aplikasi

Rancangan aplikasi untuk teknik kompresi JPEG ditunjukkan pada Gambar dibawah, pada blok diagram ini menunjukkan proses secara garis besar bagaimana proses kompresi ini bekerja.



Gambar 6: Diagram Perancangan

Aplikasi Kompresi jpeg Aplikasi yang dirancang akan menghasilkan keluaran berupa citra terkompresi yang sesuai dengan faktor kualitas yang sudah ditentukan dan dipilih. Percobaan akan dilakukan beberapa kali dengan factor kualitas tertentu, sehingga akan mendapatkan hasil kompresi dan data yang berbeda. Nantinya hasil dan data kompresi akan dibuatkan dalam bentuk tabel sehingga akan memperlihatkan hasil kompresi dengan faktor kualitas yang terbaik.

Hasil Pembahasan

Rasio Kompresi Citra Rasio kompresi citra adalah ukuran citra dalam bentuk presentase

yang telah berhasil dikompresi atau dimampatkan, sehingga dapat diketahui berapa besar hasil kompresi atau pemampatan yang dilakukan. Secara matematis rasio kompresi citra dapat dihitung sebagai berikut :

$$RasioKompresi = \left(\frac{UkuranCitraAsli}{UkuranCitrahasilKompresi} \right) \times 100 \quad (6)$$

Pengukuran Error Kompresi Citra

Informasi yang hilang akibat kompresi sebenarnya seminimal mungkin sehingga kualitas hasil kompresi bagus. Tetapi biasanya kualitas kompresi bagus bila proses kompresi menghasilkan pengurangan memori yang tidak begitu besar, demikian sebaliknya. Dalam kompresi citra terdapat standar pengukuran error kompresi yaitu :

1. MSE (Mean Square Error) MSE dapat dituliskan secara matematis, yaitu :

$$MSE = \frac{1}{MN} \sum_{y=1}^M \sum_{x=1}^N [I(x,y) - I'(x,y)]^2 \quad (7)$$

Dimana: $I(x,y)$ adalah nilai pixel di citra asli $I'(x,y)$ adalah nilai pixel pada hasil citra kompresi M,N adalah dimensi *image*.

2. PSNR (Peak Signal to Noise Ratio) PSNR dapat dituliskan secara matematis, yaitu :

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \quad (8)$$

Nilai MSE yang rendah akan lebih baik, sedangkan nilai PSNR yang tinggi akan lebih baik.

Kompresi citra (*image compression*) merupakan salah satu proses penting yang banyak aplikasi dalam pengolahan citra, namun tekniknya cukup rumit. Kompresi citra diperlukan karena kemampuannya dalam memampatkan file citra yang dapat menghemat penggunaan memori. Adapun proses yang dilakukan dalam kompresi citra JPEG yaitu :

1. Melakukan *Color Sampling* atau *Color Transform*.
2. Membagi pixel citra dengan ukuran 8x8 blok grup.

3. Setiap blok citra akan dilakukan perubahan nilai pixel dengan koefisien dari matriks persegi $M \times N$
4. Kuantisasi, yaitu membuang informasi dari hasil DCT *forward* dan DCT *Inverse*, setelah nilai DCT dikuantisasi dengan tabel kuantisasi standar dengan kualitas kompresi 50%, kemudian nilai-nilai pixel tersebut akan dilakukan Pengkodean entropi.
5. Entropy *coding* atau Pengkodean ini dilakukan secara zig-zag, dalam kompresi JPEG biasanya hal ini dapat dilakukan dengan metode Huffman atau RLE.[1]

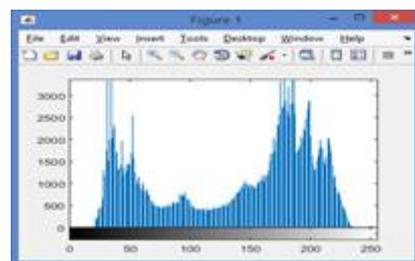
Uji Coba

Uji coba diterapkan pada komputer HP pavilion 14-e017tx dengan spesifikasi Processor Intel® Core™ i5-3230M CPU 2.60GHz dengan RAM sebesar 4.00 Gb atau setara dengan 4096 Mb, serta menggunakan program perhitungan matematis yaitu Matlab R2015a. Dalam uji coba ini, citra yang digunakan sebagai bahan uji coba kucing.bmp ditunjukkan pada gambar 7.



Gambar 7: Citra kucing.bmp

Pada gambar 8 terlihat hasil histogram citra kucing.bmp dimana penyebaran nilai histogram yang merata, dengan citra yang memiliki nilai intensitas keabuan citra yang maksimal.



Gambar 8: Histogram pada citra Kucing.bmp

Citra berikutnya yaitu dilakukan pengujian pada citra kucing2.png, dipilih karena nilai der-

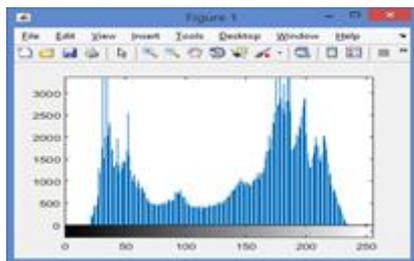
ajar keabuan citra ini memiliki nilai yang cukup merata.

Gambar 9 adalah contoh dari gambar yang akan digunakan sebagai uji coba dengan format PNG.



Gambar 9: Citra kucing2.png

Histogram citra memiliki nilai derajat keabuan yang sedang, dapat dilihat pada gambar 10.



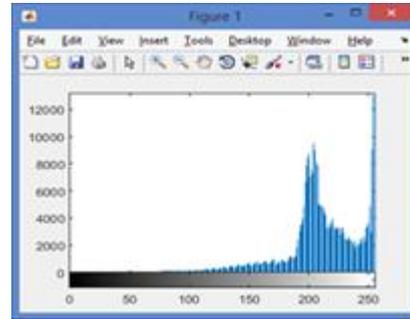
Gambar 10: Histogram pada Citra Kucing2.png

Pada sampel berikutnya yaitu winter.bmp yang dapat dilihat pada gambar 11, dimana citra ini digunakan karena object didalam citra berbeda dengan citra sebelumnya, sebab citra ini memiliki penyebaran nilai terpusat pada satu titik.



Gambar 11: Winter.bmp

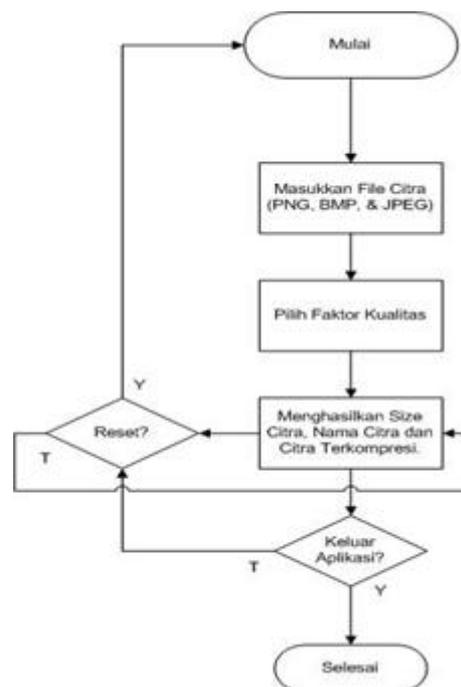
Histogram citra winter.bmp terlihat pada gambar 12



Gambar 12: Histogram pada Citra winter.bmp

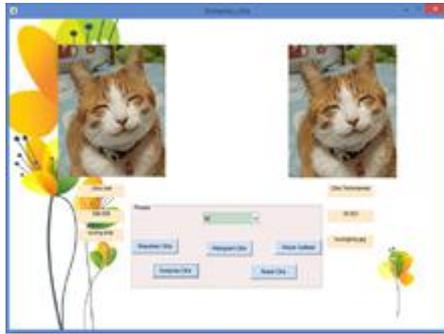
Flowchart Program

Flowchart berikut merupakan bagaimana alur program bekerja pada aplikasi yang telah dirancang.



Gambar 13: Flowchart Aplikasi Kompresi. JPEG

Pada gambar 14, merupakan hasil uji coba dari aplikasi yang berhasil menampilkan GUI (Graphical User Interface) dari sebelumnya, adalah tampilan aplikasi secara keseluruhan, terdapat button input atau untuk memasukkan file citra yang akan dikompresi, histogram sebagai button untuk menampilkan histogram pada citra, dan beberapa static text untuk menampilkan nama citra, dan size file dari citra itu sendiri.



Gambar 14: Tampilan GUI Aplikasi Kompresi JPEG

Hasil Uji Coba Citra

Hasil uji coba citra dilakukan pada 3 sampel citra yang sudah ditentukan, masing-masing citra memiliki format file berbeda yakni, PNG dan BMP. Hasil pengujian menunjukkan size atau ukuran sebelum pemampatan, sesudah pemampatan, dan berupa perbandingan antara citra dengan faktor kualitas 10-100.

Hasil Uji Coba Citra Kucing.BMP

Dalam pengujian aplikasi, dilakukan pada 3 citra dengan format PNG dan BMP, dimana citra ini adalah Kucing.bmp, kucing2.png, dan winter.bmp. Pada tabel 1, adalah tabel citra kucing.bmp dengan ukuran file sebesar 572 KB. Pada tabel ini dipaparkan *size* citra dari tiap faktor kualitas pada aplikasi yang sudah dirancang.

Tabel 1: Hasil Kompresi Citra Kucing.bmp dengan faktor kualitas 10-100

No	Size File Citra/Ukuran(Byte)	Faktor Kualitas	Ukuran Asli Citra(Byte)
1	7.214	10	586.606
2	11.008	20	586.606
3	15.155	30	586.606
4	17.195	40	586.606
5	30.623	50	586.606
6	34.785	60	586.606
7	36.902	70	586.606
8	39.977	80	586.606
9	55.918	90	586.606
10	114.015	100	586.606

Dari tabel 1 hasil kompresi citra kucing.bmp, dengan faktor kualitas 10 maka akan menghasilkan citra yang memiliki *size* dengan ukuran yang sangat kecil, namun kualitas dari gambar dapat digolongkan ke dalam kategori buruk, terlihat dari gambar yang banyak

menimbulkan noise atau derau, ini dikarenakan faktor kualitas menunjukkan tingkat kualitas pemampatan dengan tujuan untuk mendapatkan beberapa variasi hasil dari kompresi. Hasil kompresi citra berhasil menurunkan file ukuran hingga lebih kecil dari file aslinya. Pada gambar 15, adalah hasil dari citra yang terkompresi dengan faktor kualitas dari 10 hingga 100.



Gambar 15: Perbandingan hasil citra kucing .jpg terkompresi

Hasil Uji Coba Citra Kucing2.PNG

Pada bagian kedua ini adalah hasil kompresi pada citra kucing2.png dengan ukuran file sebesar 326 Kb, terlihat pada tabel 2, dimana hasil kompresi JPEG ini dengan faktor kualitas 10-100. Hasil uji coba didapatkan hasil kompresi sebesar 105.708 byte pada citra kucing2.png dengan faktor kualitas 100, citra asli sebelum dikompresi sebesar 586.606 byte, hal ini menunjukkan kompresi berhasil menekan angka atau besar ukuran file, namun tetap tidak mengurangi kualitas citra. Tetapi pada citra hasil kompresi dengan faktor kualitas yang dipilih 10, citra menghasilkan keluaran dengan ukuran file yang sangat minim, berkisar 6.58Kb atau setara dengan 6.743 byte dan dapat dilihat, citra yang dikompresi dengan faktor kualitas sebesar 10-30 menampakkan sedikit derau

atau noise yang mana citra tersebut tergolong pada citra dengan kualitas yang kurang bagus. Hasil perbandingan citra kucing2.png, terlihat pada gambar 16.

Tabel 2: Hasil Kompresi Citra Kucing2.png dengan faktor kualitas 10-100

No	Size File Citra/Ukuran (Byte)	Faktor Kualitas	Ukuran Asli Citra (Byte)
1	6.743	10	334.774
2	8.729	20	334.774
3	10.526	30	334.774
4	12.081	40	334.774
5	13.570	50	334.774
6	15.225	60	334.774
7	18.039	70	334.774
8	22.378	80	334.774
9	33.314	90	334.774
10	105.708	100	334.774



Gambar 16: Perbandingan hasil citra kucing2.png terkompresi

Hasil Uji Coba Citra Winter.bmp

Pada uji sampel ke tiga adalah citra winter.bmp terlihat pada tabel 3, kompresi citra berhasil dilakukan dengan faktor kualitas yang sudah ditentukan, dari 10 hingga 100.

Tabel 3: Hasil Kompresi Citra Winter.bmp dengan faktor kualitas 10-100.

No	Size File Citra/Ukuran (Byte)	Faktor Kualitas	Ukuran Asli Citra (Byte)
1	13.559	10	1.244.216
2	22.422	20	1.244.216
3	28.950	30	1.244.216
4	35.839	40	1.244.216
5	41.446	50	1.244.216
6	47.502	60	1.244.216
7	59.915	70	1.244.216
8	73.093	80	1.244.216
9	110.576	90	1.244.216
10	331.151	100	1.244.216

Gambar 17 memperlihatkan hasil kompresi citra winter.bmp beserta ukuran file yang sudah terkompresi. Hal yang serupa pada citra winter.bmp, setelah terkompresi pada faktor kualitas 10-30 citra terdapat banyak begitu derau atau noise dimana, hal ini juga dialami pada citra kucing.bmp dan kucing2.png



Gambar 17: Perbandingan hasil citra winter.bmp terkompresi

Dengan demikian faktor kualitas menentukan seberapa besar citra yang akan terkompresi dan seberapa baik hasil dan kualitas citra yang terkompresi, pada citra yang terkompresi dengan faktor kualitas 50-100 menghasilkan

citra yang minim ukuran namun, memiliki kualitas yang baik.

Rasio Kompresi citra ditampilkan dalam bentuk tabel dan angka sehingga nantinya akan dibuat simpulan akhir dari tabel 4.

Hasil PSNR, MSE dan Rasio Kompresi Citra Uji Coba

Tabel 4. Merupakan hasil dari pengukuran Kualitas Citra, dimana PSNR dan MSE, serta

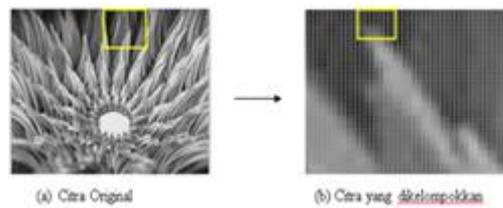
Tabel 4: Pengukuran Kualitas Citra

Faktor Kualitas	Kucing.bmp			Kucing2.PNG			Winter.bmp		
	PSNR (dB)	MSE	RK (%)	PSNR	MSE	RK (%)	PSNR	MSE	RK (%)
10	27.09	126.82	81.31	30.66	55.74	49.64	28.29	96.27	91.76
20	29.01	81.54	53.30	34.26	24.34	38.35	30.27	60.99	55.49
30	29.97	65.44	38.70	36.07	16.06	31.80	31.38	47.22	41.68
40	30.41	59.11	34.11	37.23	12.28	27.71	32.06	40.38	34.71
50	30.95	52.21	19.15	38.02	10.24	24.67	32.64	35.40	30.02
60	35.01	20.47	16.86	38.79	8.58	21.98	33.16	31.40	26.19
70	42.06	4.04	15.90	39.87	6.69	18.55	33.87	26.63	21.86
80	42.37	3.76	14.68	41.35	4.76	14.95	34.94	20.81	17.02
90	45.01	2.05	10.50	43.66	2.79	10.04	37.09	12.69	11.25
100	50.89	0.52	5.14	48.64	0.88	3.16	41.29	4.83	3.75

Pada citra kucing.bmp yang terkompresi menjadi JPEG, dengan faktor kualitas 10 menghasilkan nilai PSNR dibawah 30 dB (Decibel) hal ini mengindikasikan citra ini memiliki kualitas yang rendah dan ini berpengaruh pada, lain halnya dengan citra kucing2.png citra ini mendapatkan PSNR senilai 30.667 dB, dan pada citra Winter.bmp didapat PSNR dibawah 30 dB atau berkisar 28.295 dB. Faktor kualitas yang rendah akan memberikan proses kuantisasi yang banyak menghilangkan data pada citra setelah di kompresi sehingga menjadikan nilai PSNR semakin rendah tetapi nilai rasio kompresi semakin tinggi. Kualitas citra yang baik ketika nilai PSNR lebih tinggi dari nilai MSE. Proses Kompresi Gambar *Discrete Cosine Transform* (DCT) bekerja dengan memisahkan gambar ke bagian frekuensi yang berbeda. Pada langkah kuan-

tisasi dimana proses kompresi terjadi, sehingga nantinya frekuensi dan informasi yang kurang penting dibuang. Diasumsikan suatu citra *grayscale* digunakan sebagai contoh kasus untuk DCT, berikut ini adalah algoritma *Discrete Cosine Transform*:

1. Gambar dibagi menjadi beberapa bagian blok, dan setiap blok memiliki ordo 8x8 atau 8x8 pixel.



Gambar 18: Citra Sampel *Grayscale*

2. Data matriks original dikurangi dengan 128, hal ini disebut sebagai *color transform* atau *color conversion*, algoritma DCT bekerja pada rentang [-128 127] dengan ketentuan pengolahan citra digital pada citra berwarna

3. Lakukan pengurangan pada tiap baris dan kolom matrix original dengan titik nol dari [0 255] maka matriks original dikurangi dengan 128 didapatkan matriks sebagai berikut:

$$M = \begin{pmatrix} 26 & -5 & -5 & -5 & -5 & -5 & -5 & 8 \\ 64 & 52 & 8 & 26 & 26 & 26 & 8 & -18 \\ 126 & 70 & 26 & 26 & 52 & 26 & -5 & -5 \\ 111 & 52 & 8 & 52 & 52 & 38 & -5 & -5 \\ 52 & 26 & 8 & 39 & 38 & 21 & 8 & 8 \\ 0 & 8 & -5 & 8 & 26 & 52 & 70 & 26 \\ -5 & -23 & -18 & 21 & 8 & 8 & 52 & 38 \end{pmatrix}$$

Gambar 19: Matriks M

Kemudian, cari nilai untuk Matriks *Discrete Cosine Transform* untuk Matriks T yang nantinya matriks T ini akan dikalikan dengan Matriks M, dan matriks T ini akan di transpose menjadi matriks T^t .

$$T(i, j) = 1/\sqrt{N}, \text{ jika } i = 0$$

$$\sqrt{\frac{2}{N}} \cos\left(\frac{(2j+1)i\pi}{2N}\right) \text{ jika } i \neq 0$$

Dengan rumusan diatas, matriks T akan dihitung mulai dari blok T(0,0) sampai T(7,7). Berikut ini adalah perhitungan nilai DCT untuk Matriks T :

$$T = \begin{pmatrix} 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 \\ 0.4904 & 0.4157 & 0.2778 & 0.0975 & -0.0975 & -0.2778 & -0.4157 & -0.4904 \\ 0.4619 & 0.2916 & -0.1913 & -0.4619 & -0.4619 & -0.1916 & 0.1916 & 0.4619 \\ 0.4157 & -0.0975 & -0.4904 & -0.2778 & 0.2778 & 0.4904 & 0.0975 & -0.4157 \\ 0.3536 & -0.3536 & -0.3536 & 0.3536 & 0.3536 & -0.3536 & -0.3536 & 0.3536 \\ 0.2778 & -0.4904 & 0.0975 & 0.4157 & -0.4157 & -0.0975 & -0.4904 & -0.2778 \\ 0.1913 & -0.4619 & 0.4619 & -0.1880 & -0.1913 & 0.4619 & -0.4619 & 0.1913 \\ 0.0975 & -0.2778 & 0.4157 & -0.4904 & 0.4904 & -0.4157 & 0.2778 & -0.0975 \end{pmatrix}$$

Gambar 20: Matriks T

Selanjutnya matriks T akan dilakukan *transpose* menjadi matriks T^t , ditunjukkan pada gambar 21.

$$T^t = \begin{pmatrix} 0.3536 & 0.4904 & 0.4619 & 0.4157 & 0.3536 & -0.2778 & 0.1913 & 0.0975 \\ 0.3536 & 0.4157 & 0.2916 & -0.0975 & -0.3536 & -0.4904 & -0.4619 & -0.2778 \\ 0.3536 & 0.2778 & -0.1913 & -0.4904 & 0.3536 & 0.0975 & 0.4619 & 0.4157 \\ 0.3536 & 0.0975 & -0.4619 & -0.2778 & 0.3536 & 0.4157 & -0.1913 & -0.4904 \\ 0.3536 & -0.0975 & -0.4619 & 0.2778 & 0.3536 & -0.4157 & -0.1913 & 0.4904 \\ 0.3536 & -0.2778 & 0.1913 & 0.4904 & -0.3536 & -0.0975 & -0.4619 & -0.4157 \\ 0.3536 & -0.4157 & 0.1913 & 0.4904 & -0.3536 & -0.4904 & -0.4619 & -0.2778 \\ 0.3536 & -0.4904 & 0.4619 & -0.4157 & 0.3536 & -0.2778 & 0.1913 & -0.0975 \end{pmatrix}$$

Gambar 21: Matriks T^t

4. Dengan menggunakan persamaan *Discrete Cosine Transform*, dimana untuk mencari matriks D dan matriks tersebut akan digunakan untuk proses kuantisasi lanjutan. Untuk mencari matriks D digunakan persamaan $D = T \cdot M \cdot T^t$. Setelah dilakukan perkalian pada matriks T, M dan T^t , didapatkan matriks D pada gambar 22

$$D = \begin{pmatrix} 162.3 & 40.6 & 20 & 72.3 & 30.3 & -109.1 & -19.3 & -11.5 \\ 30.5 & 108.4 & 10.5 & 32.3 & 27.7 & -11.5 & 18.4 & -2.0 \\ -94.1 & -60 & 32.3 & -43.4 & -31.3 & 43.7 & -3.6 & 7.1 \\ -38.6 & -83.4 & -5.4 & -22.2 & -15.5 & 21.3 & -1.4 & 3.5 \\ -31.3 & 17.9 & -5.5 & -12.4 & 34.3 & 31.8 & 11.5 & -6 \\ -29.7 & 40.4 & -7.6 & -2.6 & 4.5 & 26.5 & 20.5 & 2.5 \\ 5.2 & -2 & 32.2 & 6.9 & -18.5 & -19.4 & 7.6 & 11.9 \\ -10 & 11.2 & 7.8 & -16.3 & 21.5 & 10.6 & 3.9 & 10.7 \end{pmatrix}$$

Gambar 22: Matriks D

Matriks D berisi koefisien DCT, yang kemudian akan dikuantisasi dengan level kuantisasi yang dipilih, atau dengan faktor kualitas yang telah dipilih.

5. Matriks D yang berisi koefisien DCT, data yang terletak pada kiri atas merupakan korelasi dari frekuensi-frekuensi rendah dari data original. Sedangkan pada kanan bawah merupakan korelasi frekuensi tinggi dari data original. Selanjutnya dilakukan proses kuantisasi dengan Quality 50, dimana hasil terlihat pada gambar 23

$$Q50 = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 41 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}$$

Gambar 23: Matriks Q50

Kuantisasi dilakukan dengan persamaan

berikut ini, dimana round berarti mendekati nilai hasil pembagian ke dalam pembulatan integer yang paling dekat.

$$C(i,j) = \text{round}(Dij/Qij) \quad (9)$$

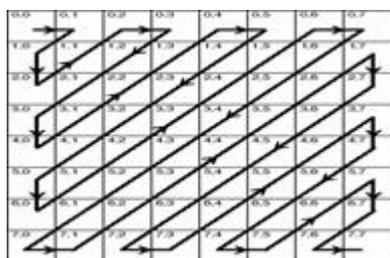
Maka dari perhitungan diatas didapatkan matriks C dengan nilai yang telah terkuantisasi dan dapat dilihat pada gambar24

$$C = \begin{pmatrix} 10 & 3 & 2 & 4 & 1 & -2 & 0 & 0 \\ 2 & 9 & 0 & 1 & 1 & -0 & 0 & -0 \\ -4 & 1 & 0 & -1 & -0 & 0 & -0 & 0 \\ -2 & -4 & -0 & -0 & -0 & 0 & -0 & 0 \\ 1 & 0 & -0 & 0 & 0 & 0 & 0 & -0 \\ -3 & 1 & -0 & -0 & 0 & 0 & 0 & 0 \\ 0 & -0 & 0 & 0 & -0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Gambar 24: Matriks C

Citra original yang telah terkuantisasi dan sebelum terkuantisasi dapat dilihat perbandingannya, pada citra original dengan citra matriks C.

- Susun bilangan menggunakan fungsi zig-zag scanning dimana ini adalah langkah terakhir pada proses kompresi, proses ini ditunjukkan pada gambar 25.



Gambar 25: Metode Zig-Zag Scanning

Matriks C yang telah terkuantisasi akan dikonversi oleh encoder dapat disebut juga dengan proses pengkodean entropi. Matriks C ini akan diubah ke data biner (0110101...) koefisien DCT terkuantisasi sehingga bit yang paling kiri berisikan nilai-nilai yang tidak 0, dan yang paling kanan berisikan baris bit yang bernilai0. Setelah diurutkan, maka proses kompresi dapat dilanjutkan dengan algoritma Huffman atau RLE.

- Dilakukan proses dekompresi untuk merekonstruksi hasil kompresi, ini merupakan proses dimana hasil akan ditampilkan dan menjadi data yang dapat dikenali. Hitung dengan persamaan sebagai:

$$R(i,j) = Q(i,j) \times C(i,j)$$

Setelah matriks Q50 dikalikan dengan koefisien matriks C, lakukan perhitungan dengan persamaan berikut :

$$N = \text{round}(T' \cdot R \cdot T) + 128$$

Didapatkan matriks R setelah dilakukan perhitungan terhadap matriks Q50 dengan matriks C, seperti pada gambar 26.

$$R = \begin{pmatrix} 160 & 33 & 20 & 64 & 24 & -40 & 0 & 0 \\ 24 & 108 & 0 & 19 & 26 & -0 & 0 & -0 \\ -84 & -52 & 0 & -24 & -0 & 0 & -0 & 0 \\ -84 & -68 & -0 & -0 & -0 & 0 & -0 & 0 \\ 18 & 0 & -0 & 0 & 0 & 0 & 0 & -0 \\ -24 & 35 & -0 & -0 & 0 & 0 & 0 & 0 \\ 0 & -0 & 0 & 0 & -0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Gambar 26: Matriks R

Matriks T' dikalikan dengan matriks R, kemudian hasil dari perkalian tersebut dikalikan dengan matriks T, dan hasilnya ditambahkan dengan 128. Didapatkan hasil matriks N dan ini adalah hasil akhir dari proses yang sudah dikompresi dalam blok 8x8, hasil pada gambar27.

$$N = \begin{pmatrix} 155 & 131 & 113 & 126 & 146 & 133 & 132 & 135 \\ 193 & 165 & 138 & 145 & 164 & 147 & 140 & 136 \\ 237 & 204 & 168 & 164 & 176 & 155 & 137 & 125 \\ 234 & 204 & 168 & 162 & 178 & 161 & 144 & 127 \\ 177 & 158 & 136 & 142 & 173 & 174 & 171 & 159 \\ 130 & 122 & 109 & 116 & 149 & 158 & 164 & 156 \\ 116 & 118 & 112 & 118 & 149 & 162 & 176 & 173 \\ 131 & 137 & 132 & 131 & 153 & 169 & 177 & 175 \end{pmatrix}$$

Gambar 27: Matriks N

Pada hasil yang didapat, dilakukan perbandingan matriks Original dengan matriks N, dimana nilai koefisien tiap blok berubah dan ditunjukkan pada gambar 28 (a) dan gambar 28 (b)

134	123	123	123	123	123	123	136
192	180	136	154	154	154	136	110
234	198	154	154	180	154	123	123
239	180	136	180	180	166	123	123
180	154	136	167	166	149	136	136
123	136	123	136	154	180	198	154
123	105	110	149	136	156	180	166
110	136	123	123	123	136	154	136

Gambar 28: (a) Matriks Original

155	131	111	124	146	133	132	135
193	165	138	145	164	147	140	136
237	204	168	164	176	153	137	123
234	204	168	162	178	161	144	127
177	158	136	142	173	174	171	159
130	122	109	116	149	158	164	156
116	118	112	118	149	162	176	173
151	137	132	131	153	163	177	175

Gambar 29: Matriks N

Pada gambar 28 dan 29 adalah perbandingan Matriks Original dengan Matriks N yang merupakan hasil akhir dari proses kompresi yang dilakukan, contoh perhitungan tersebut dilakukan dengan kompresi faktor kualitas 50 persen, atau dengan kuantisasi 50.

Penutup

Dari penelitian yang dilakukan, pada 3 citra berbeda sebagai sampel uji coba, disimpulkan bahwa :

1. Kompresi yang dilakukan pada citra PNG berhasil menurunkan *size*/ ukuran citra secara signifikan, ditunjukkan ukuran citra asli 334.774 menjadi 105.708 (byte) atau setara dengan 103Kb.
2. Aplikasi kompresi citra yang dirancang sudah dapat berjalan sesuai dengan yang diharapkan dan dapat mengkompresi citra dengan baik sesuai faktor kualitas yang ada.
3. Tingkat rasio file citra masing-masing memiliki tingkatan yang berbeda, ini dikarenakan faktor kualitas kompresi citra yang berbeda. Dengan faktor kualitas 10 maka kompresi yang dilakukan akan semakin tinggi dan mempengaruhi nilai rasio kompresi.

4. Hasil PSNR dan MSE mengindikasikan kualitas citra yang baik, dimana nilai PSNR lebih tinggi dibandingkan dengan nilai MSE.

Saran dari penelitian yang telah dilakukan yaitu, dimasa yang akan datang dapat dilakukan pengembangan untuk kompresi citra menggunakan metode yang lain dan bisa membandingkan metode mana yang lebih cocok pada citra dengan format citra yang ada.

Daftar Pustaka

- [1] Marvin Ch. Wijaya dan Agus Prijono, "Pengolahan Citra Digital Menggunakan Matlab", Bandung: Informatika Bandung, 2007.
- [2] Aqwam Rosadi, Sunny Arief Sudiro dan Sarifuddin Madenda, "Efficient Implementation of Mean Formula for Image Processing using FPGA Device", Proceeding of International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE 2016), ISBN:978-602-60280-0-6, Yogyakarta, 2016
- [3] Liliana, Andy Febrico Bintoro dan Iwan Njoto Sandjaja, "Pembuatan Aplikasi Kompresi Image dengan Metode Fast Discrete Cosine Transform", Jurnal Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra, 2012.
- [4] Darma Putra, "Pengolahan Citra Digital", Yogyakarta. Penerbit: ANDI, 2011.
- [5] Sigit R, Basuki, A., Ramadijanti, N., dan Pramadihanto, D., "Step by step Pengolahan Citra Digital", Yogyakarta: Andi, 2005.
- [6] Thomas Wiegand and Heiko Schwarz, "Source Coding: Part I of Fundamentals of Source and Video Coding", Foundations and Trends in sample Vol. 1, No 1, p-p : 1-217, 2011.
- [7] Dewobroto Wiryanto, "Aplikasi Rekayasa Konstruksi dengan Visual Basic 6.0", Jakarta, Penerbit Elex Media Komputindo, 2005.