

IMPLEMENTASI ALGORITMA *RUN LENGTH ENCODING* UNTUK PERANCANGAN APLIKASI KOMPRESI DAN DEKOMPRESI FILE CITRA

Cut Try Utari

Program Studi Magister Teknik Informatika
Fakultas Ilmu Komputer dan Teknik Informatika
Universitas Sumatera Utara
Jln. Civitas Akademi No. 9 Kampus USU Medan
E-mail : tarie.try91@gmail.com

Abstrak

Komunikasi data merupakan kegiatan yang sering dilakukan dalam bidang teknologi informasi. Komunikasi data itu sendiri berhubungan erat dengan pengiriman data dengan menggunakan sistem elektronik dari satu terminal komputer ke terminal komputer lain. Namun, yang terkadang menjadi kendala yaitu besarnya ukuran data / file yang akan dikirimkan. Ukuran file citra relatif besar dibandingkan data / file dengan format teks, yang membutuhkan memori yang besar dalam melakukan penyimpanan maupun dalam transmisi (pengiriman gambar) melalui media komunikasi. Semakin besar data / file maka akan semakin lama waktu yang dibutuhkan untuk melakukan transfer data dan sebaliknya semakin kecil data / file maka semakin cepat dalam melakukan transfer data. Pengiriman data dengan ukuran besar terkadang dapat beresiko tidak dapat tertampung pada media penyimpanan dan tersampaikan atau gagal terkirim serta dapat memperkecil kapasitas kosong pada memori media penyimpanan. Untuk itu perlu dikembangkan aplikasi untuk kompresi citra. Kompresi citra bertujuan untuk meminimalkan kebutuhan memori dalam mempresentasikan citra digital dengan mengurangi duplikasi data. Kompresi data berarti suatu teknik untuk memampatkan dalam citra sehingga memori yang dibutuhkan menjadi lebih sedikit daripada representasi citra data agar diperoleh data dengan ukuran yang lebih kecil dari pada ukuran aslinya sehingga lebih efisien dalam menyimpannya serta mempersingkat waktu pertukaran data tersebut. Kajian ini membahas kompresi citra yang memiliki kelompok derajat nilai keabu-abuan (grayscale) yang sama. Metode Run Length Encoding digunakan dalam kompresi citra dengan melakukan pengelompokan nilai derajat keabu-abuan yang sama dan menyimpannya dalam format rle yang tidak bisa dilihat. Agar file citra yang terkompres dapat dilihat kembali, maka dilakukan proses dekompresi dengan cara mengembalikan nilai derajat keabu-abuan yang sama atas citra tersebut. Dengan dibangunnya aplikasi ini, ukuran file citra dapat lebih kecil sehingga dapat mempercepat dalam proses pengiriman maupun penyimpanan.

Kata kunci : Komunikasi data, Transmisi, Kompresi Citra, Grayscale, Run Length Encoding.

1. Pendahuluan

Perkembangan teknologi komputer menjadikan hidup lebih fleksibel, mudah menjangkau daerah - daerah yang luas, dan otomatisasi di segala bidang. Informatika sebagai salah satu bidang dari ilmu pengetahuan dan teknologi yang berkembang dengan cepat dimaksudkan agar dapat berperan membantu manusia memperoleh kehidupan yang lebih sempurna.

Dalam bidang teknologi informasi, komunikasi data sangat sering dilakukan. Komunikasi data ini berhubungan erat dengan pengiriman data menggunakan sistem transmisi elektronik dari satu terminal komputer ke terminal komputer yang lain. Besarnya ukuran data terkadang menjadi kendala dalam proses pengiriman data.

Data dengan ukuran besar akan memakan waktu transfer yang lebih lama dibandingkan dengan data yang memiliki ukuran lebih kecil, terkadang ada resiko tidak dapat tertampung pada media penyimpanan dan tidak tersampainya, sehingga akan memperkecil kapasitas kosong dalam memori media penyimpanan. Oleh karena itu, manusia selalu berusaha untuk menemukan suatu cara alternatif untuk

menangani permasalahan tersebut, salah satunya dengan cara kompresi. Ada beberapa macam format data yaitu berupa teks maupun citra (images) [1].

Image adalah sebuah gambar, foto yang ditampilkan atau bentuk lain yang memberikan representasi visual tentang sebuah obyek atau pemandangan. Pada DIP (Digital Image Processing) sebuah gambar bilangan array 2 dimensi, yang setiap barisnya adalah representasi piksel pada gambar setiap barisnya. Ukuran gambar biasanya 256 x 256, 512 x 512, 1024 x 1024. minimum nilai piksel = 0 (hitam), maksimum = 255 (putih) dan bilangan antara 0 s/d 255 merepresentasikan derajat keabuan. Gambar berwarna dapat direpresentasikan dengan array 2D Red, green dan blue, 3D. Komputer membutuhkan memory lebih banyak untuk data ini rata-rata 3kali data [2].

Kompresi citra bertujuan untuk meminimalkan kebutuhan memori dalam mempresentasikan citra digital dengan mengurangi duplikasi data di semula. Kompresi data berarti suatu teknik untuk memampatkan dalam citra sehingga memori yang dibutuhkan menjadi lebih sedikit daripada representasi citra data agar diperoleh data dengan

ukuran yang lebih kecil dari pada ukuran aslinya sehingga lebih efisien dalam menyimpannya serta mempersingkat waktu pertukaran data tersebut [1].

Dengan kata lain, kompresi data sebenarnya adalah proses meminimalkan ukuran data atau berkas dengan mengurangi data yang berulang, karena umumnya pada sebuah data sering terjadi pengulangan. Data yang telah dikompres agar bisa digunakan kembali harus dikembalikan lagi seperti semula. Proses pengembalian sebuah data yang terkompresi menjadi seperti data aslinya disebut dengan dekompresi [1].

Pada kompresi data, terdapat dua tipe macam kompresi, yaitu *lossless compression* dan *lossy compression*. Pada *lossless compression*, semua informasi yang ada pada data akan kembali menjadi seperti aslinya dan tidak ada informasi yang hilang. Teknik ini biasanya digunakan untuk dokumen-dokumen, file executable, dan lainnya. Karena, kehilangan sebuah informasi merupakan hal yang fatal bagi file-file tersebut. Sedangkan pada *lossy compression*, tidak semua informasi yang ada akan kembali seperti semula. Hanya informasi-informasi inti yang dikembalikan. Hal ini terjadi, karena pada *lossy compression* informasi-informasi yang tidak berguna akan dihilangkan. walaupun ada informasi yang hilang, namun hal ini tidak terlalu disadari oleh pengguna. Teknik ini biasanya digunakan pada file video, gambar, suara yang mana file-file tersebut biasanya berukuran besar [1].

Dengan adanya kompresi diharapkan dapat menghemat biaya serta waktu yang dikeluarkan guna menambah fasilitas media penyimpanan data pada komputer serta mempercepat proses transfer data. Banyak teknik algoritma yang dapat digunakan untuk melakukan kompresi file, sebagai contoh adalah algoritma Huffman, algoritma LZW (Lempel-Ziv-Welch), algoritma DMC (Dinamic Markov Compression), Block-Shorting Lossless, Run-Length, Shannon-Fano, Arithmetic, PPM (Prediction by Partial Matching), Burrows-Wheeler Block Sorting, dan Half Byte, dan lain-lain [1]. Beberapa software kompresi yang banyak digunakan para pengguna komputer dewasa ini diantaranya adalah WinZip (menghasilkan format .zip) dan WinRAR (menghasilkan format .rar). Maka berdasarkan latar belakang yang telah dikemukakan di atas, judul yang diambil dalam penelitian ini adalah "Implementasi Algoritma Run Length Encoding Untuk Perancangan Aplikasi Kompresi Dan Dekompresi File Citra".

2. Tinjauan Pustaka

Pengertian Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran atau system perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal – sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan [2].

Citra dapat dikelompokkan menjadi dua bagian yaitu citra diam (*still images*) dan citra bergerak (*moving images*). Citra diam adalah citra tunggal yang tidak bergerak. Sedangkan citra bergerak adalah rangkaian citra diam yang ditampilkan secara beruntun (sekuensial) sehingga memberi kesan pada mata sebagai gambar yang bergerak. Setiap citra didalam rangkaian itu disebut *frame*.

Citra dalam format *BMP* lebih bagus dari pada citra dalam format *JPEG*, yang pada umumnya tidak dimampatkan sehingga tidak ada informasi yang hilang. Citra dalam format *BMP* ada tiga macam : citra biner, citra berwarna, dan citra hitam-putih (*grayscale*). Citra biner hanya mempunyai dua nilai keabuan, yaitu 0 dan 1. Warna yang terlihat pada citra bitmap merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau, dan biru. Setiap pixel disusun oleh tiga komponen warna : R (*red*), G (*green*), dan B (*blue*). Kombinasi dari tiga warna RGB tersebut menghasilkan warna yang khas untuk pixel yang bersangkutan [3].

Citra Digital

Citra dapat dikatakan sebagai citra digital jika citra tersebut disimpan dalam format digital (dalam bentuk file). Citra digital dihasilkan melalui proses digitalisasi terhadap citra kontinu. Citra adalah angka (*image is just a number*), dari segi estetika, citra atau gambar adalah kumpulan warna yang bisa terlihat indah, memiliki pola, berbentuk abstrak dan lain sebagainya. Citra digital dapat berupa foto udara, penampang lintang (*cross section*) dari suatu benda, gambar wajah, hasil tomografi otak dan lain sebagainya. Dari segi ilmiah, citra adalah gambar 3-dimensi (3D) dari suatu fungsi, biasanya intensitas warna sebagai fungsi spatial x dan y . Di computer warna dapat dinyatakan, misalnya sebagai angka dalam bentuk skala RGB. Karena citra adalah angka, maka citra dapat diproses secara digital [4].

Nilai suatu *pixel* memiliki nilai dalam rentang tertentu, dari nilai minimum sampai nilai maksimum. Jangkauan yang digunakan berbeda-beda tergantung dari jenis warnanya. Namun secara umum jangkauannya adalah 0 – 255. Citra dengan penggambaran seperti ini digolongkan ke dalam citra *integer*. Berikut adalah jenis-jenis citra berdasarkan nilai *pixel*-nya.

a. Citra Biner

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai *pixel* yaitu hitam dan putih. Citra biner juga disebut sebagai citra B&W (*black and white*) atau citra monokrom. Hanya dibutuhkan 1 bit untuk mewakili nilai setiap *pixel* dari citra biner. Citra biner sering kali muncul sebagai hasil dari proses pengolahan seperti segmentasi, pengambangan, morfologi ataupun *dithering*.



Gambar 1. Contoh Citra Biner

b. Citra Grayscale

Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap *pixel*-nya, dengan kata lain nilai bagian RED = GREEN = BLUE. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih. Tingkatan keabuan di sini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih. Citra *grayscale* berikut memiliki kedalaman warna 8 bit (256 kombinasi warna keabuan).



Gambar 2. Contoh Citra Grayscale

c. Citra Warna (True Color)

Setiap *pixel* dari citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = Red Green Blue). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte*, yang berarti setiap warna mempunyai gradiasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $2^8 \cdot 2^8 \cdot 2^8 = 2^{24} = 16$ juta warna lebih. Itulah sebabnya format ini dinamakan true color karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam.

Penyimpanan citra true color di dalam memori berbeda dengan citra grayscale. Setiap piksel dari citra grayscale 256 gradiasi warna diwakili oleh 1 *byte*. Sedangkan 1 piksel citra true color diwakili oleh 3 *byte*, dimana masing – masing *byte* merepresentasikan warna merah (Red), hijau (Green), dan biru (Blue) [2].



Gambar 3. Contoh Citra warna

Format File Citra

Ada banyak macam format file citra, dan format diantaranya yang sangat populer adalah *BMP*, *JPEG*, dan *GIF*. Setiap program pengolahan citra biasanya memiliki format citra tersendiri. Format dan metode dari suatu citra yang baik juga sangat bergantung pada jenis citranya. Setiap format file citra memiliki kelebihan dan kekurangan masing-masing dalam hal citra yang disimpan.

Citra tertentu dapat disimpan dengan baik (dalam arti ukuran file lebih kecil dan kualitas gambar tidak berubah) pada format file citra tertentu, apabila disimpan pada format lain kadang kala dapat menyebabkan ukuran file menjadi lebih besar dari aslinya dan kualitas citra dapat menurun. Oleh karena itu, untuk menyimpan suatu citra harus diperhatikan citra dan format file citra apa yang sesuai [5].

Kompresi Data

Kompresi data berarti sebuah proses mengkodekan informasi menggunakan bit atau information-bearing unit yang lain yang lebih rendah daripada representasi data yang tidak terkodekan dengan suatu sistem encoding tertentu.

Kompresi data juga diartikan sebagai teknik untuk memampatkan data agar diperoleh data dengan ukuran yang lebih kecil daripada ukuran aslinya sehingga lebih efisien dalam menyimpannya atau mempersingkat waktu pertukaran data tersebut [1].

Dekompresi

Sebuah data yang sudah dikompres tentunya harus dapat dikembalikan lagi ke bentuk aslinya, prinsip ini dinamakan dekompresi. Untuk dapat merubah data yang terkompres diperlukan cara yang berbeda seperti pada waktu proses kompres dilaksanakan. Jadi pada saat dekompres terdapat catatan header yang berupa *byte-byte* yang berisi catatan mengenai isi dari file tersebut [6].

Algoritma Run Length Encoding (RLE)

Algoritma RLE menggunakan pendekatan ruang. Algoritma ini cocok digunakan untuk memampatkan citra yang memiliki kelompok – kelompok piksel berderajat keabu – abuan yang sama. Metode ini dilakukan dengan menyatakan seluruh baris citra menjadi sebuah baris run, lalu menghitung run – length untuk setiap derajat keabu – abuan yang berurutan. Contoh, andaikan string “ABBABABACAACDDD” akan di-encode dengan algoritma RLE maka langkahnya adalah sebagai berikut.

KODE = (A,1)(B,2)(A,1)(B,1)(A,1)(B,1)(A,1)(C,1)(A,2)(C,1)(D,3)

Contoh lain, misalnya sebuah citra grayscale 3 bit berukuran 10 x 10 piksel akan di – encode dengan algoritma RLE maka :

Tabel 1. Matriks Citra 10 x 10 piksel

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 2 | 2 | 7 | 7 | 7 | 7 | 2 | 2 | 2 |
| 4 | 4 | 1 | 1 | 1 | 7 | 7 | 7 | 7 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 7 | 7 | 7 | 7 | 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 4 | 5 | 5 | 5 | 5 | 2 | 2 | 2 | 2 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 |
| 7 | 7 | 7 | 6 | 6 | 6 | 6 | 2 | 2 | 2 |
| 4 | 4 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 |
| 7 | 7 | 7 | 7 | 7 | 7 | 4 | 4 | 4 | 4 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 4 | 4 | 2 | 2 | 5 | 5 | 5 |

Citra belum terkompres

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 2 | 7 | 7 | 7 | 7 | 2 | 2 | 2 | 2 |
| 4 | 4 | 1 | 1 | 1 | 7 | 7 | 7 | 7 | 7 |
| 7 | 7 | 7 | 7 | 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 4 | 5 | 5 | 5 | 5 | 2 | 2 | 2 | 2 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 |
| 7 | 7 | 7 | 6 | 6 | 6 | 6 | 2 | 2 | 2 |
| 4 | 4 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 |
| 7 | 7 | 7 | 7 | 7 | 7 | 4 | 4 | 4 | 4 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 4 | 4 | 2 | 2 | 5 | 5 | 5 |

Hasil Kompresi

- (2,2) (7,4) (2,4)
- (4,2) (1,3) (7,5)
- (7,4) (3,6)
- (4,2) (5,4) (2,4)
- (5,4) (6,6)
- (7,3) (6,4) (2,3)
- (4,2) (3,5) (5,3)
- (7,6) (4,4)
- (0,10)
- (1,3) (4,2) (2,2) (5,3)

Hasil pengodean :

2 2 7 4 2 4 4 2 1 3 7 5 7 4 3 6 4 2 5 4 2 4 5 4 6 6 7
3 6 4 2 3 4 2 5 5 3 7 6 4 4 0 10 1 3 4 2 2 2 5 3

Semuanya = 52 piksel

Ukuran citra sebelum dikompres = 10 x 10 x 3 bit = 300 bit

Ukuran citra setelah dikompres = 52 x 3 bit = 156

$$\text{Rasio pemampatan} = \left(100\% - \frac{156}{300} \times 100\% \right)$$

= 48 %, artinya 48% dari citra semula telah dimampatkan.

Metode RLE sangat cocok untuk citra biner sedangkan untuk citra grayscale, sering kali pengodean hasil kompresi lebih besar dibanding sebelum terkompres. Metode RLE dapat dikombinasikan dengan metode Huffman untuk meningkatkan rasio kompresi. Mula – mula lakukan kompresi RLE, lalu hasilnya dimampatkan lagi dengan metode Huffman [2].

3. Metode Penelitian

Langkah – langkah proses program

Adapun proses program kompresi dan dekompresi RLE ini adalah sebagai berikut :

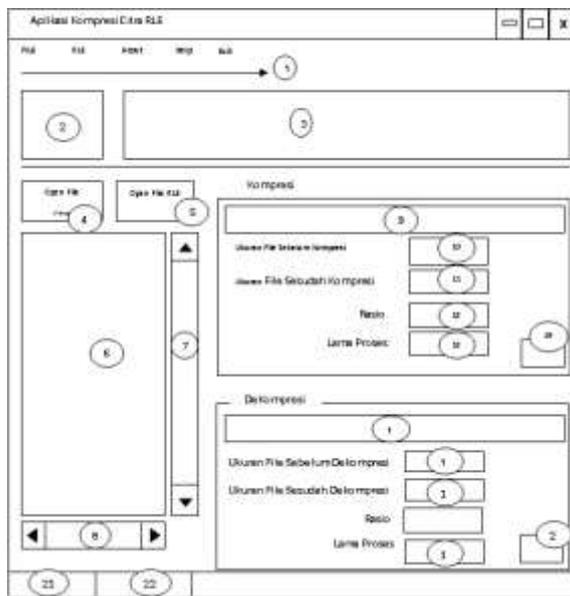
- a. Tahap pertama citra akan direpresentasikan dalam bentuk matriks.
- b. Kemudian berdasarkan representasi citra dalam bentuk matriks, maka akan disusun suatu senarai berantai (link list) dari baris pertama sampai baris terakhir. Setiap nomor warna merupakan satu simpul dalam senarai.
- c. Berdasarkan senarai berantai diatas, maka akan dilakukan kompresi metode RLE dengan Algoritma sebagai berikut :
 1. Ambil dua byte pertama dari sederetan karakter yang berurutan dalam suatu senarai berantai citra digital.
 2. Lakukan perulangan untuk perintah selanjutnya.
 3. Mengecek apakah kedua data bytes yang diambil apakah sama.
 4. Jika sama maka kedua bytes tersebut dikeluarkan, dan dihitung pengulangan yang terjadi untuk dicatat pada counter.
 5. Keluaran nilai pencatat counter dengan karakter khusus (#).
 6. Lakukan update terhadap pointer yang menunjuk pada deretan data yang terakhir.
 7. Ambil dua byte berikutnya.
 8. Ulangi langkah no 2.
 9. Jika tidak sama byte pertama di dikeluarkan, byte yang kedua di jadikan byte pertama, dan ambil byte berikutnya sebagai byte kedua kemudian lakukan update terhadap pointer yang menunjuk data byte yang terakhir dan ulangi langkah no. 2.

Rancangan Aplikasi Kompresi Citra RLE

Keterangan :

1. Menu, Ada beberapa menu pada rancangan aplikasi kompresi citra RLE ini,yaitu menu File, RLE, About, Help dan Exit.
 - a. Menu File yang terdiri dari 2 sub menu yaitu : Open, dan exit.
 - b. Menu RLE yang terdiri dari 2 sub menu yaitu Kompres dan Dekompres.
 - c. Menu About untuk menampilkan form About.
 - d. Menu Help untuk menampilkan form Help.
 - e. Menu Exit untuk keluar dari halaman Aplikasi Kompresi Citra RLE dan kembali ke Tampilan Utama.
2. Picture Box tempat menampilkan gambar.
3. Label untuk menampilkan judul aplikasi.
4. Command Button Open File Citra untuk memanggil file citra yang berformat JPG dan BMP.
5. Menu Open File RLE untuk memanggil file citra yang berformat RLE.
6. Picture untuk menampilkan citra yang berformat JPG dan BMP.
7. Vscroll untuk menggeser ke atas dan ke bawah posisi citra pada picture box.

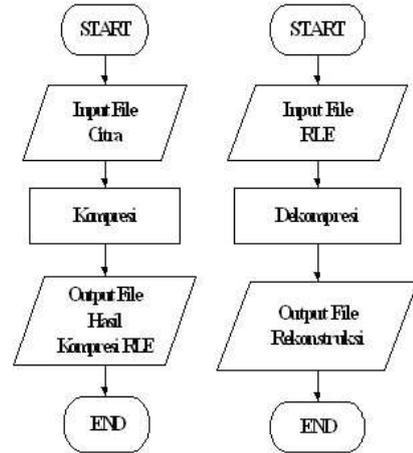
8. Hscroll untuk menggeser ke kiri dan ke kanan posisi citra pada picture box.
9. Label untuk menampilkan lokasi file citra JPG dan BMP tersimpan.
10. TextBox Label untuk menampilkan ukuran file sebelum di-kompres.
11. TextBox untuk menampilkan ukuran file sesudah di-kompres.
12. TextBox untuk menampilkan rasio kompres.
13. TextBox untuk menampilkan lama proses kompres.
14. Command Button Kompres untuk melakukan kompresi file citra yang berformat JPG dan BMP.
15. Label untuk menampilkan lokasi file citra RLE tersimpan.
16. TextBox untuk menampilkan ukuran file sebelum dekompres.



Gambar 4. Rancangan Form Aplikasi Kompresi Citra RLE

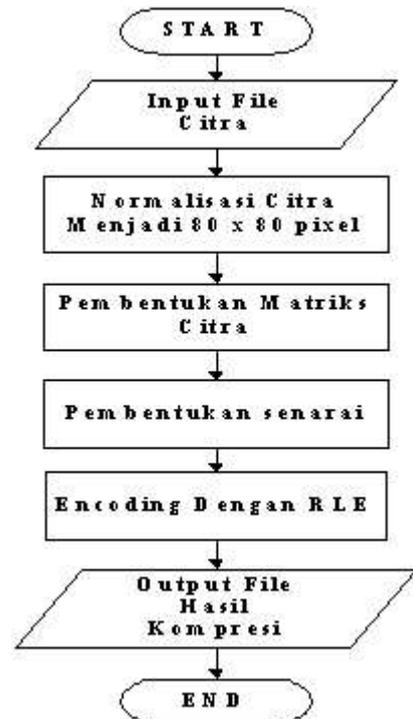
Flowchart Sistem Kompresi dan dekompresi RLE

Flowchart sistem kompresi adalah bagan yang menunjukkan alur kerja atau proses kompresi dan dekompresi dengan menggunakan algoritma Run Length Encoding. Rancangan flowchart sistem kompresi dan dekompresi RLE dapat dilihat pada gambar 5.



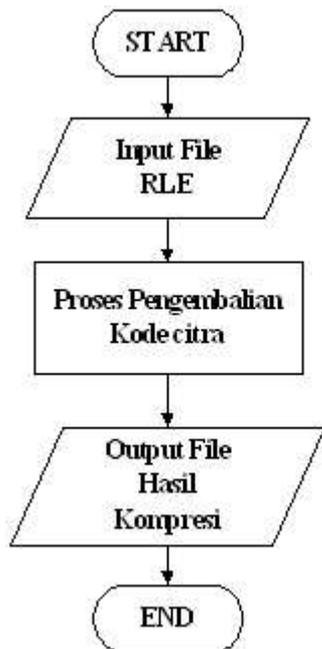
Gambar 5. Flowchart Sistem Kompresi dan Dekompresi RLE

a. Flowchart Proses Kompresi



Gambar 6. Flowchart Proses Kompresi RLE

b. Flowchart Proses Dekompresi



Gambar 7. Flowchart Proses Dekompresi

4. Hasil Dan Pembahasan

Hasil Penelitian

Adapun hasil penelitian yang diperoleh penulis setelah melakukan implementasi adalah tampilan hasil dari sebuah aplikasi yang mampu mengimplementasikan algoritma Run Length Encoding (RLE) untuk melakukan kompresi dan dekompresi file citra.

Tampilan hasil program yang diperoleh antara lain :

a. Tampilan Form Utama

Tampilan form utama muncul pertama kali ketika program dieksekusi.



Gambar 8. Form Utama

b. Tampilan Form Aplikasi Kompresi RLE



Gambar 9. Form Aplikasi Kompresi RLE

c. Tampilan Hasil Kompresi

Tampilan Hasil Kompresi berfungsi untuk menampilkan hasil kompresi file citra pada text box berupa ukuran file sebelum dan sesudah kompresi, rasio kompresi serta lama proses kompresi. Bentuk tampilan Hasil Kompresi dapat dilihat pada gambar 10 berikut ini.



Gambar 10. Tampilan Hasil Kompresi

d. Tampilan Open File RLE

Tampilan Open File RLE berfungsi untuk melakukan pencarian file citra hasil kompresi yang berekstensi RLE dari disk komputer untuk ditampilkan pada picture box dan selanjutnya dilakukan proses. Bentuk tampilan Open File RLE dapat dilihat pada gambar 11 berikut ini.



Gambar 11. Tampilan Open File RLE

e. Tampilan Hasil Dekompresi

Tampilan Hasil Dekompresi berfungsi untuk menampilkan hasil dekompresi file citra pada text box berupa ukuran file sebelum dan sesudah kompresi, rasio dekompresi serta lama proses dekompresi. Bentuk tampilan hasil dekompresi dapat dilihat pada gambar 12. berikut ini.



Gambar 12. Tampilan Hasil Dekompresi

Hasil Pengujian

Adapun hasil – hasil pengujian kompresi dan dekompresi citra dapat dilihat pada tampilan gambar dibawah ini.

a. Hasil Kompresi dan Dekompresi Citra Format JPG



Gambar 13. Hasil Kompresi&Dekompresi Citra Format JPG

Keterangan :

Hasil Kompresi

Ukuran file sebelum kompresi: 103.38Kb

Ukuran file setelah kompresi: 91.53Kb

Ratio : 11.47 %

Lama Proses: 18 detik

Hasil Dekompresi

Ukuran file sebelum dekompresi: 91.53Kb

Ukuran file setelah dekompresi: 103.38Kb

Ratio: 3.21 %

Lama Proses: 5 detik

b. Hasil Kompresi dan Dekompresi Citra Grayscale format BMP



Gambar 14. Hasil Kompresi dan Dekompresi Citra Grayscale format BMP

Keterangan :

Hasil Kompresi

Ukuran file sebelum kompresi: 258.28Kb

Ukuran file setelah kompresi : 181.61 Kb

Ratio: 5 %

Lama Proses: 16 detik

Hasil Dekompresi

Ukuran file sebelum dekompresi: 181.61Kb

Ukuran file setelah dekompresi: 258.28 Kb

Ratio: 3.21 %

Lama Proses: 5 detik

c. Hasil Kompresi dan Dekompresi Citra Grayscale Format JPG



Gambar 15. Hasil Kompresi dan Dekompresi Citra Grayscale Format JPG

Keterangan :

Hasil Kompresi

Ukuran file sebelum kompresi: 160.39 Kb

Ukuran file setelah kompresi: 160.35 Kb

Ratio: 2 %

Lama Proses: 27 detik

Hasil Dekompresi

Ukuran file sebelum dekompresi: 160.35 Kb

Ukuran file setelah dekompresi: 160.39 Kb

Ratio: 2.85 %

Lama Proses: 8 detik

Pembahasan

Aplikasi RLE ini menerapkan Algoritma RLE dalam melakukan pengkompresian dan dekompresi file citra. File citra yang berekstensi JPG dan BMP akan dikompres agar ukuran file yang tadinya besar akan menjadi lebih kecil dan hasil kompresi tersebut akan disimpan dalam format .rle. Hasil file citra yang telah dikompres tidak dapat dibuka atau dilihat gambarnya. Maka untuk bisa membuka atau melihat kembali gambar tersebut gambar tersebut harus dikembalikan kebentuk semula yaitu dengan cara dilakukannya dekompresi file citra. Setelah dilakukannya dekompresi maka file citra akan kembali kedalam format JPG dan BMP.

Tinjau citra 10×10 pixel dengan 8 derajat keabuan yang dinyatakan sebagai matriks derajat keabuan sebagai berikut :

0 0 0 0 2 2 2 2 2
0 0 0 1 1 1 1 2 2 2
1 1 1 1 1 1 1 1 1 1
4 4 4 4 3 3 3 3 2 2
3 3 3 5 5 7 7 7 7 6
2 2 6 0 0 0 0 1 1 0
3 3 4 4 3 2 2 2 1 1
0 0 0 0 0 0 0 0 1 1
1 1 1 1 0 0 0 2 2 2
3 3 3 2 2 2 1 1 1 1

semuanya ada 100 buah nilai.

Pasangan nilai untuk setiap baris runyang dihasilkan dengan metode kompresi RLE:

(0, 5), (2, 5)
(0, 3), (1, 4), (2, 3)
(1, 10)
(4, 4), (3, 4), (2 2)
(3, 3), (5, 2), (7, 4), (6, 1)
(2, 2), (6, 1), (0, 4), (1, 2), (0, 1)
(3, 2), (4, 2), (3, 1), (2, 2), (1, 2)
(0, 8), (1, 2)
(1, 4), (0, 3), (2, 3)
(3, 3), (2, 3), (1, 4)

semuanya ada 31 pasangan nilai atau $31 \times 2 = 62$ nilai. Ukuran citra sebelum dikompres (1 derajat keabuan = 3 bit) adalah $100 \times 3 \text{ bit} = 300 \text{ bit}$, sedangkan ukuran citra setelah dikompres (derajat keabuan = 3 bit, run length = 4 bit):

$(31 \times 3) + (31 \times 4) \text{ bit} = 217 \text{ bit}$.

Ratio = $((300-217)/300) \times 100 \% = 27,67 \%$, yang artinya 27.67% dari citra semula telah dikompres.

Dari file hasil pengujian yang telah dilakukan kita tinjau perolehan Ratio kompresi RLE nya, sebagai contoh

File Gambar Kupu-Kupu yang Berformat JPG Berikut :

Ukuran file sebelum kompresi: 103.38 Kb

Ukuran file setelah kompresi: 91.53 Kb

Ratio: $((103,38-91,53)/103,38) \times 100 \%$
 $=11,47 \%$

5. Kesimpulan

Kesimpulan yang menjadi hasil dari penulisan penelitian ini adalah :

1. Perangkat lunak kompresi/dekompresi citra dengan Algoritma Run Length Encoding (RLE) dapat melakukan kompresi file citra dengan format bmp maupun jpg.
2. Kompresi citra dengan metode Run Length Encoding (RLE) dapat memperkecil ukuran file, sehingga dapat menghemat ruang tempat penyimpanan (Storage).
3. Perangkat lunak kompresi citra RLE ini dapat berjalan dengan baik untuk citra yang memiliki derajat keabu-abuan (grayscale) yang sama.
4. Perangkat lunak kompresi citra RLE ini tidak cocok untuk kompresi citra yang memiliki RGB yang tinggi.
5. Perangkat lunak kompresi citra RLE ini berguna untuk mengubah ukuran file citra yang semula

berukuran besar bisa diperkecil sehingga dapat membantu dalam pengiriman file citra agar tidak menghabiskan waktu yang cukup lama dalam melakukan pengiriman data citra dan tidak menghabiskan banyak tempat pada media penyimpanan data.

6. Daftar Pustaka

- [1] Subarkah, A.F. 2010. Rancang Bangun Aplikasi Kompresi File Menggunakan Metode LZW berbasis Java. UIN Malang : Teknik Informatika.
- [2] Sutoyo, T., Mulyanto, E., Suhartono, V., Nurhayati, D.O., Wijanarto.2009. Teori Pengolahan Citra Digital. Yogyakarta: Andi.
- [3] Priyanto, Budi. 1992. Dasar-dasar Pengolahan Citra. Jakarta : Elexmedia Komputindo.
- [4] Menovita. A. 2010. Implementasi Algoritma Huffman Pada Kompresi Citra BMP. USU : Ilmu Komputer.
- [5] Murni, Aniati dan Setiawan, Suryana. 1992. Pengantar Pengolahan Citra. Jakarta : Elexmedia Komputindo.
- [6] Munir, Rinaldi. 2004. Pengolahan Citra Digital dengan Pendekatan Algoritmik. Jakarta : Informatika.