

PERANCANGAN APLIKASI KOMPRESI MENGGUNAKAN METODE DEFLATE.

Dalu Aditya Utomo^{1*}, Indra Kenedi², Juju Jumadi³

Program Studi Informatika, Universitas Dehasen Bengkulu^{1,2,3}

Daluaditya9@gmail.com^{1*}, Indrakenedi@gmail.com², Juju.jumadi@unived.ac.id³

ABSTRACT

Data compression or data compression is a field that focuses on forming a small output file from a large file. In other words, the field of data compression aims to overcome problems caused by the large size or capacity of the file. Compression is a way to reduce storage costs by eliminating redundancy or repetition that occurs in files or files. One compression method that is well known is the deflate compression method. Deflate is a compression algorithm that belongs to lossless compression.

From the results of the tests carried out, the results obtained that the largest compression ratio is owned by document files with RTF type with a value of 88.75%. The worst compression ratio is actually obtained from image files with jpeg type where the compression ratio is only 0.38%.

ABSTRAK

Kompresi data atau *data compression* merupakan bidang yang berfokus pada membentuk output file dengan ukuran yang kecil dari file yang berukuran besar. Dengan kata lain bidang kompresi data bertujuan untuk mengatasi masalah-masalah yang diakibatkan oleh besarnya ukuran atau kapasitas file. Kompresi merupakan sebuah cara untuk mengurangi biaya penyimpanan dengan menghilangkan redundansi atau perulangan yang terjadi pada berkas atau *file*. Salah satu metode kompresi yang cukup dikenal adalah metode kompresi deflate. Deflate merupakan algoritma kompresi yang tergolong kedalam kompresi *lossless*

Dari hasil pengujian yang dilakukan maka diperoleh hasil rasio kompresi terbesar dimiliki oleh file dokumen dengan tipe RTF dengan nilai adalah sebesar 88.75 %. Rasio kompresi terburuk justru di peroleh dari file gambar dengan tipe jpeg dimana rasio kompresi hanya diperoleh 0.38 %.

1. Pendahuluan

Inovasi – inovasi dalam bidang komputer banyak terhambat dikarenakan alasan kapasitas penyimpanan atau memori yang masih langka dan mahal. Inovasi perangkat lunak dan aplikasi melahirkan dilema baru, yaitu semakin besarnya ukuran file atau berkas yang dihasilkan. Ukuran file atau berkas yang besar bukan saja menjadi masalah dalam hal penyimpanan, namun juga berkembang menjadi masalah lain saat komunikasi antar komputer mulai berkembang yang ditandai dengan maraknya penggunaan internet di seluruh dunia. Pertukaran data bukan lagi menjadi masalah yang besar jika diperhatikan dari sudut pandang jarak, namun menjadi masalah yang sangat mengganggu jika di pandang dari sudut pandang ukuran data yang akan dikirimkan.

Kompresi data atau *data compression* merupakan bidang yang berfokus pada membentuk output file dengan ukuran yang kecil dari file yang berukuran besar. Dengan kata lain bidang kompresi data bertujuan untuk mengatasi masalah-masalah yang diakibatkan oleh besarnya ukuran atau kapasitas file. Dalam bidang kompresi data telah banyak muncul metode – metode yang dapat digunakan dalam kompresi data. Kompresi merupakan sebuah cara untuk mengurangi biaya penyimpanan dengan menghilangkan redundansi atau perulangan yang terjadi pada berkas atau *file* (Suarjaya, 2012). Salah satu metode kompresi yang cukup dikenal adalah metode kompresi deflate. Deflate merupakan algoritma kompresi yang tergolong kedalam kompresi *lossless* (Gopal, et al., 2011) . Algoritma Deflate banyak di-implementasikan pada dunia komputasi sejak lama yang mana salah satunya adalah kompresi Zip.

2. Tinjauan Pustaka

A. File

Sebuah *file* pada dunia komputer adalah sebuah potongan yang mengandung informasi yang tersedia bagi sistem operasi atau aplikasi komputer. *File* pada setiap komputer disimpan pada media penyimpanan seperti *hard drive*, *optical drive*, dan media penyimpanan lainnya. Beberapa file yang disimpan dan diorganisasi secara khusus dan spesifik biasanya disebut sebagai *file system* (Fisher, 2016).

B. Pengkodean Data

Data disimpan di komputer pada memori utama untuk diproses. Sebuah karakter data disimpan dalam memori utama menempati posisi 1 byte. Pada saat ini komputer merepresentasikan 1 byte terdiri dari 8 bit, yang mana tiap karakter direpresentasikan kedalam 8 bit pada memori utama (Hartono, 2015)

C. Kompresi

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 (Utari, 2016)

Kompresi adalah pengubahan data kedalam bentuk yang memerlukan *bit* yang lebih sedikit, biasanya dilakukan agar data dapat disimpan atau dikirimkan dengan lebih efisien. Kebalikan dan proses kompresi, yaitu dekompresi. Dekompresi sendiri merupakan proses untuk mengembalikan data baru yang telah dihasilkan oleh proses kompresi menjadi data awal. Data yang telah dilakukan kompresi dapat digunakan jaringan yang lebih rendah yang sesuai dengan kapasitas data yang telah dilakukan proses kompres (Firmansah & Setiawan, 2015).

Kompresi data dapat dibagi menjadi dua jenis yaitu *lossles* dan *lossy*. Kompresi *lossles* merupakan kompresi dimana data yang telah dikompresi dapat kembali menjadi data semula, sedangkan kompresi *lossy* akan menghasilkan data kompresi yang tidak dapat dikembalikan kembali menjadi data semula

1. Kompresi Lossy

Kompresi Lossy adalah suatu metode untuk mengkompresi data, 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) (Purwoto & Ardiyanto , 2015)

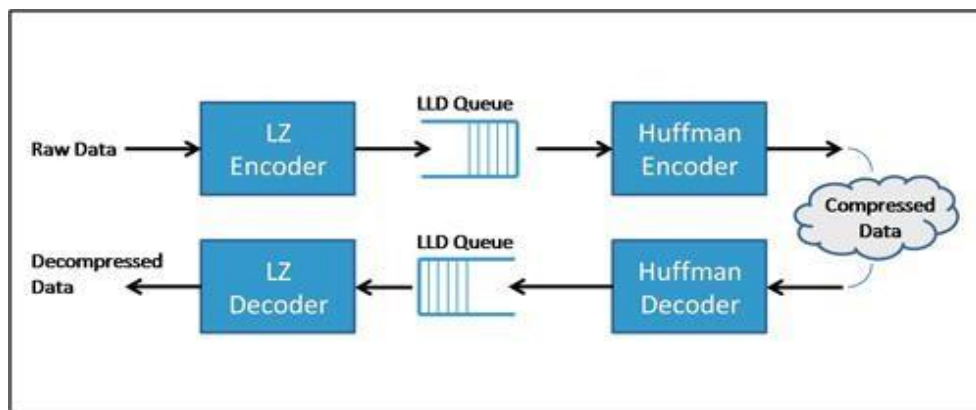
2. Kompresi Lossless

Kompresi Lossless diperlukan untuk data teks dan file, seperti catatan bank, artikel teks dll. 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 (Purwoto & Ardiyanto , 2015).

D. Metode Deflate

Metode kompresi deflate merupakan metode kompresi yang pada dasarnya menggunakan kombinasi metode LZ77 dan algoritma *Huffman Coding* (Aguilera, 2015) . Algoritma Deflate terdiri dari dua bagian yaitu penghilangan redundansi dan

pengkodean entropy (Brown, 2016). Penghilangan redundansi pada deflate merupakan bagian dari metode Lempel Ziv. Penghilangan redundansi dilakukan dengan mencari tiga atau lebih byte yang muncul secara berurutan yang identik dengan urutan byte saat ini yang kemudian akan digantikan dengan variable referensi yang sesuai.



Gambar 1 Proses Kompresi Deflate

Pada dasarnya kompresi deflate menggunakan kombinasi *Huffman* dan *LZ77*, yang dapat dijabarkan sebagai berikut :

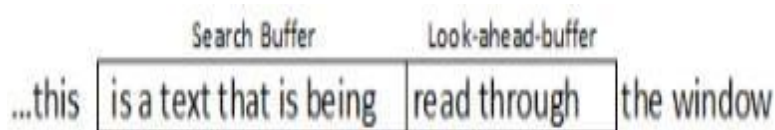
1. Huffman Encoding

Cara pembentukan kode Huffman adalah dengan membentuk pohon biner, yang dinamakan pohon Huffman, sebagai berikut :

- a. Pilih dua simbol dengan peluang paling kecil. Kedua simbol tadi dikombinasikan sebagai simpul orangtua sehingga menjadi simbol 2 karakter dengan peluang yang dijumlahkan
- b. Selanjutnya pilih dua simbol berikutnya termasuk simbol baru yang mempunyai peluang terkecil.
- c. Prosedur yang sama dilakukan pada dua simbol berikutnya yang mempunyai peluang terkecil

2. LZ77 Encoding

Algoritma kompresi LZ77 melakukan kompresinya dengan mengganti porsi data dengan suatu referensi untuk menyamakan data yang sudah dilewati oleh *encoder* dan *decoder*. Algoritma ini menggunakan suatu “*sliding window*” yang terdiri dari *search buffer* dan *look-ahead buffer*. *Search buffer* digunakan sebagai kamus, sedangkan *look-ahead buffer* merupakan buffer yang berisi string yang akan dikompresi. *Search buffer* merupakan buffer yang sudah melalui tahap kompresi, namun digunakan sebagai kamus. Kedua buffer tersebut dapat dilihat pada contoh berikut



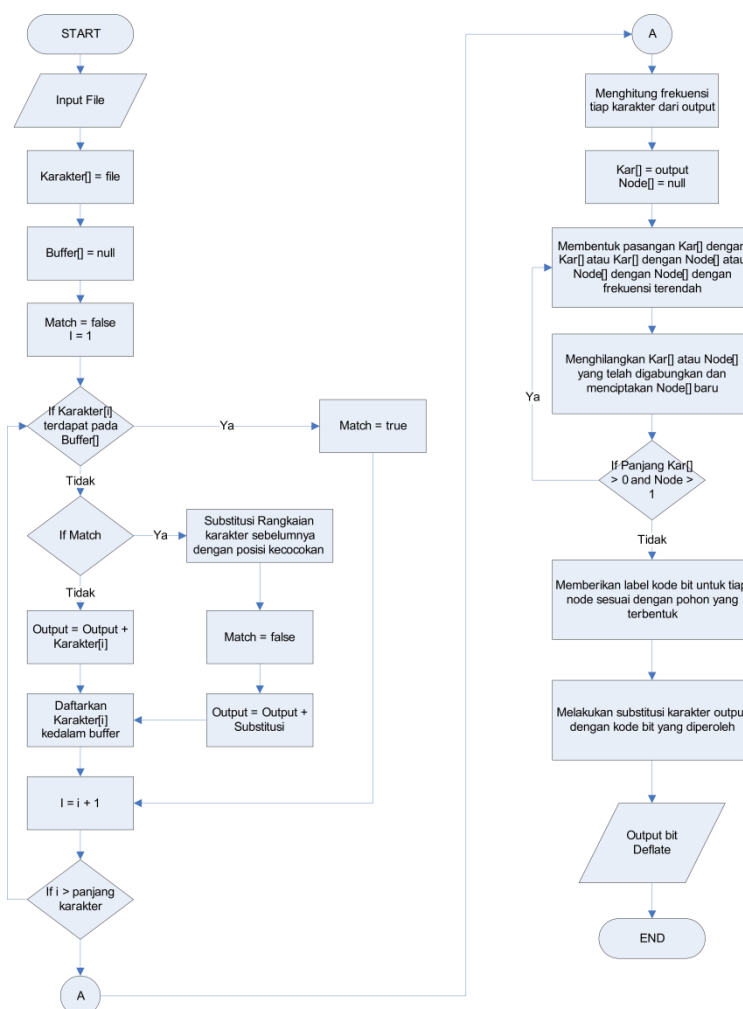
Gambar 2 Search Buffer Pada LZ77

3. Metode Penelitian

A. Metode Penelitian Applied Research

Dalam melakukan penelitian ini, penulis menggunakan metode terapan (*applied research*). Penelitian terapan dilakukan berkenaan dengan kenyataan-kenyataan praktis, penerapan, dan pengembangan ilmu pengetahuan yang dihasilkan oleh penelitian dasar dalam kehidupan nyata. Penelitian terapan berfungsi untuk mencari solusi tentang masalah-masalah tertentu. Tujuan utama penelitian terapan adalah pemecahan masalah sehingga hasil penelitian dapat dimanfaatkan untuk kepentingan manusia baik secara individu atau kelompok.

B. Flowchart Proses Deflate



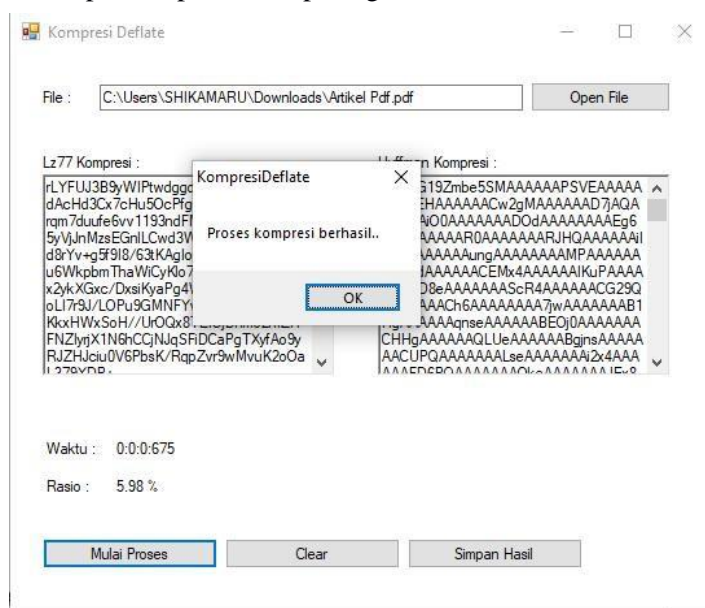
Gambar 3 Flowchart Proses Deflate

3. Hasil dan pembahasan

Penelitian kompresi data dengan menerapkan metode *deflate* yang dilakukan ini menghasilkan sebuah sistem aplikasi yang dapat digunakan untuk meng-kompresi dan dekompresi file. Adapun hasil dari penelitian ini terdiri dari beberapa *form* yaitu *form* utama, *form* kompresi dan *form* dekompresi.

A. Interface Form Kompresi

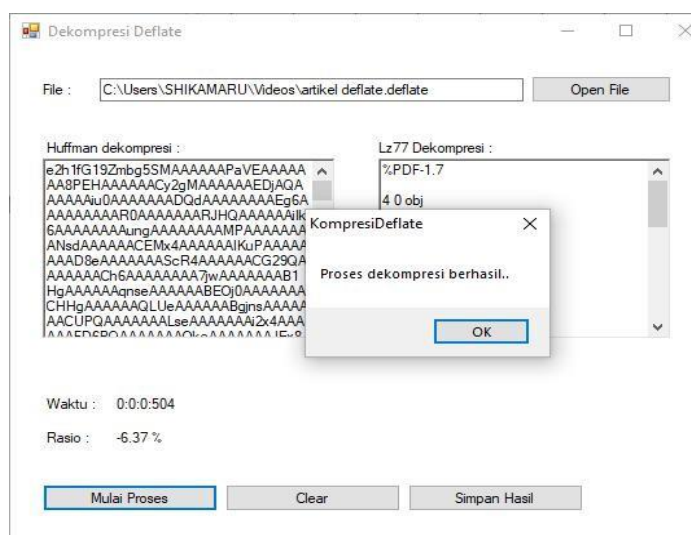
Pada *interface form* kompresi mempunyai beberapa *button* yang berfungsi untuk melakukan proses kompresi. Hasil dari proses kompresi akan disimpan didalam pada *drive* komputer untuk nanti bisa digunakan kembali pada proses dekompresi. Adapun tampilan *form* kompresi dapat dilihat pada gambar berikut ini :



Gambar 4 *Form Proses Kompresi*

B. Interface Form Dekompresi

Interface for dekompresi berfungsi untuk dekompresi file yg telah di kompresi sebelumnya. Adapun proses dekompresi seperti gambar berikut ini :



Gambar 5. Interface Form Dekompresi

5. Pengujian Sistem

Pengujian dilakukan untuk memperoleh akurasi dan efektivitas kinerja dari metode *deflate* dalam melakukan kompresi dan dekompresi file. Pengujian dilakukan dengan melakukan proses kompresi dan dekompresi dengan ekstensi file yang berbeda. Berikut tabel hasil pengujian.

Tabel 1 Pengujian Kompresi Aplikasi

Nama File	Sebelum Kompresi	Setelah Kompresi	Ratio (%) Kompresi
Cara Pakai Gamu.doc	470 KB	443 KB	5.77
Leger_K2013_XSOS1.xls	22 KB	19 KB	17.56
Coding Sederhana .txt	3 KB	1 KB	70.84
Contoh Daftar Isi.rtf	124 KB	14 KB	88.75
Artikel .Pdf	600 KB	406 KB	32.3
Tanpa Batas Waktu .mp3	6.066 KB	5.920 KB	2.41
Café Pantai .JPEG	42.8 KB	42.7	0.38
Sekolah.BMP	149 KB	56 KB	62.22
Google Chrome .exe	1.266 KB	1.177 KB	6.98

6. Kesimpulan

Dari pengujian yang telah dilakukan maka diperoleh hasil yang dapat dilihat bahwa program aplikasi kompresi menggunakan algoritma deflate yang dibangun dapat memberikan hasil yang cukup baik, dimana file secara umum dapat dikompresi dengan baik.

Rasio kompresi yang diperoleh dari hasil pengujian sebelumnya di peroleh bahwa rasio terbesar dimiliki oleh file dokumen dengan tipe RTF dimana rasio yang diperoleh adalah sebesar 88.75 %. Rasio kompresi terburuk justru di peroleh dari file gambar juga namun dengan tipe jpeg dimana rasio kompresi hanya diperoleh 0.38 %. Untuk jenis file lainnya rasio kompresi bervariasi bergantung pada tipe format dari file tersebut.

REFERENSI

- Aguilera, P. (2015). *Comparison of different image compression formats*. ECE 533 Project Report.
- Brown, R. D. (2016). *Reconstructing Corrupted Deflated Files*. ELSEVIER
- Firmansah, L., & Setiawan, B. E. (2015). Kompresi Data Audio Lossless format FLAC menjadi Audio Lossy format MP3 dengan Algoritma Huffman Shift Coding. e-Proceeding of Engineering : Vol.2, No.3 Desember 2015 ISSN : 2355-9365
- Fisher, T. (2016, November 28). *File*. Retrieved November 28, 2016, from PC Support About. Com: <http://pcsupport.about.com/od/termsf/g/file-definition.htm>
- Hartono, J. (2015). *Pengenalan Komputer*. Yogyakarta: Penerbit Andi
- Purwoto, B., & Ardiyanto, D. (2015). Kompresi Citra Dengan Menggunakan Metode Delta Modulation. Jurnal Emitor Vol. 14No. 01 ISSN 1411-8890, 1-11
- Sutanta, E. (2017). *Sistem Basis Data Edisi Revisi*. Yogyakarta: Graha Ilmu.
- Utari, C. (2016). Implementasi Algoritma Run Length Encoding Untuk Perancangan Aplikasi Kompresi Dan Dekompresi File Citra. Jurnal TIMES , Vol. V No 2 : 24-31 ISSN : 2337