

BAB IV

IMPLEMENTASI DAN ANALISIS

4.1 Perancangan Program

4.1.1 Perangkat Pendukung

Untuk mengimplementasikan metode kompresi Huffman dan *Dynamic Marcov Compression* ke dalam bentuk program diperlukan beberapa perangkat pendukung di antaranya perangkat keras (*hardware*) dan perangkat lunak (*software*).

4.1.1.1 Perangkat Keras

Spesifikasi perangkat keras yang digunakan penulis untuk melakukan perbandingan kinerja metode kompresi Huffman dan *Dynamic Markov Compression* ini adalah sebagai berikut:

1. *Processor* : Pentium(R) 4 CPU 2,66GHz
2. *Memory* : RAM 248 MB
3. *Hard disk* : 20 GB
4. *Monitor* : 15,4"
5. *Keyboard dan mouse*

4.1.1.2 Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan penulis untuk melakukan perbandingan kinerja metode kompresi Huffman dan *Dynamic Markov Compression* ini adalah sebagai berikut:

1. Sistem Operasi : *Windows XP*
2. Bahasa pemrograman : *Delphi*
3. Sistem Basis data : *Microsoft Access*

4.1.2 Perancangan Antarmuka

Sebelum memasuki tahap implementasi metode kompresi Huffman dan *Dynamic Markov Compression*, haruslah dibuat suatu perancangan yang bertujuan untuk memberikan gambaran tentang program aplikasi yang akan dibuat. Perancangan dalam hal ini adalah perancangan tampilan antarmuka (*user interface*).

Penyajian suatu program dengan menggunakan tampilan antarmuka selain membuat program menjadi lebih menarik, juga dapat memudahkan pengguna dalam mengoperasikan program tersebut. Selanjutnya akan dijelaskan gambaran tentang rancangan tampilan program yang akan dibuat.

Rancangan pertama adalah membuat menu untuk menampilkan informasi mengenai file yang akan dikompresi, file hasil kompresi dan data-data yang berkaitan dengan proses pengompresian file tersebut. Data-data tersebut akan di tampilkan dalam bentuk tabel. Ilustrasi rancangan dapat dilihat pada Gambar (4.1).

Perbandingan Kompresi Huffman dan DMC

Input Data Sampel | Grafik Rasio | Grafik Kecepatan Kompresi

sampel

Sampel Huffman :

Nama File	Ukuran Asli	Ukuran Hasil	Rasio	kecepatan Kompresi

Sampel DMC :

Nama File	Ukuran Asli	Ukuran Hasil	Rasio	kecepatan Kompresi

Input Data Hapus Data

Gambar 4.1 Rancangan Tampilan Data Sampel Hasil Kompresi

Tabel pada rancangan antarmuka pertama terdiri dari 5 kolom. Kolom pertama adalah kolom “Nama File”, kolom ini berisi nama file yang akan dikompresi beserta ekstensinya. Kolom kedua adalah kolom “Ukuran Asli”, kolom ini berisi ukuran dari file yang akan dikompresi dengan satuan *Byte*. Kolom ketiga adalah kolom “Ukuran Hasil”, kolom ini berisi ukuran file hasil kompresi dengan satuan *byte*. Kolom keempat adalah kolom “Rasio”, kolom ini berisi rasio kompresi dalam persen, rumus untuk mendapatkan rasio kompresi adalah:

$$\text{Rasio} = \frac{\text{Ukuran Hasil}}{\text{Ukuran Asli}} \times 100\%$$

Kolom terakhir adalah kolom “Kecepatan”, satuan untuk kecepatan adalah KB/s dan rumus untuk mendapatkan kecepatan adalah:

$$\text{Kecepatan} = \frac{\text{Waktu kompresi}}{\text{Ukuran Asli}}$$

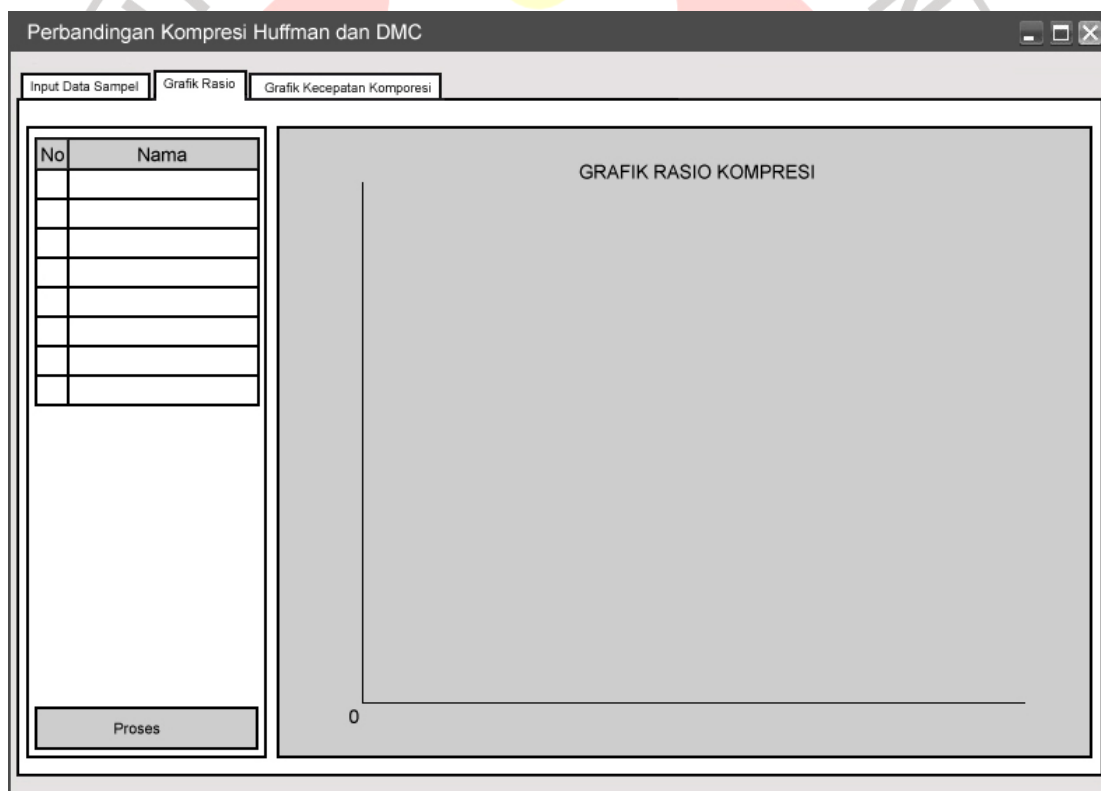
Rancangan kedua adalah membuat tampilan untuk meng-*input* sampel file yang akan dikompresi. Rancangan ini akan muncul dengan mengklik tombol “Input Data” pada rancangan pertama. Dalam rancangan ini file yang di-*input* terdiri dari satu objek. *Input* yang diterima berupa file dalam bentuk apapun. Ilustrasi rancangan dapat dilihat pada Gambar (4.2)

Gambar 4.2 Rancangan Tampilan Tambah Data Sampel

Peng-*input*-an file dilakukan dengan mengklik tombol “Browse”, setelah meng-*input* file maka selanjutnya akan muncul alamat file yang akan dikompresi dan ukuran file tersebut. Pada gambar terdapat dua tombol “PROSES”, yaitu tombol “PROSES” untuk metode kompresi Huffman dan tombol “PROSES” untuk metode *Dynamic Markov Compression*, dengan mengklik tombol ini maka proses pengompresian dimulai, dan setelah proses selesai data-data yang berkaitan

dengan proses pengompresian yaitu “Ukuran”, “Rasio”, dan “Kecepatan Kompresi” secara otomatis akan terisi. Selanjutnya data-data tersebut akan disimpan dalam basis data dan akan ditampilkan pada rancangan antarmuka pertama dengan mengklik tombol “Simpan”. Dan untuk menghapus data-data pada tabel pada rancangan pertama dilakukan dengan mengklik tombol “Hapus”.

Rancangan ketiga adalah membuat tampilan untuk grafik perbandingan rasio kompresi kedua metode. Ilustrasi rancangan dapat dilihat pada Gambar (4.3).



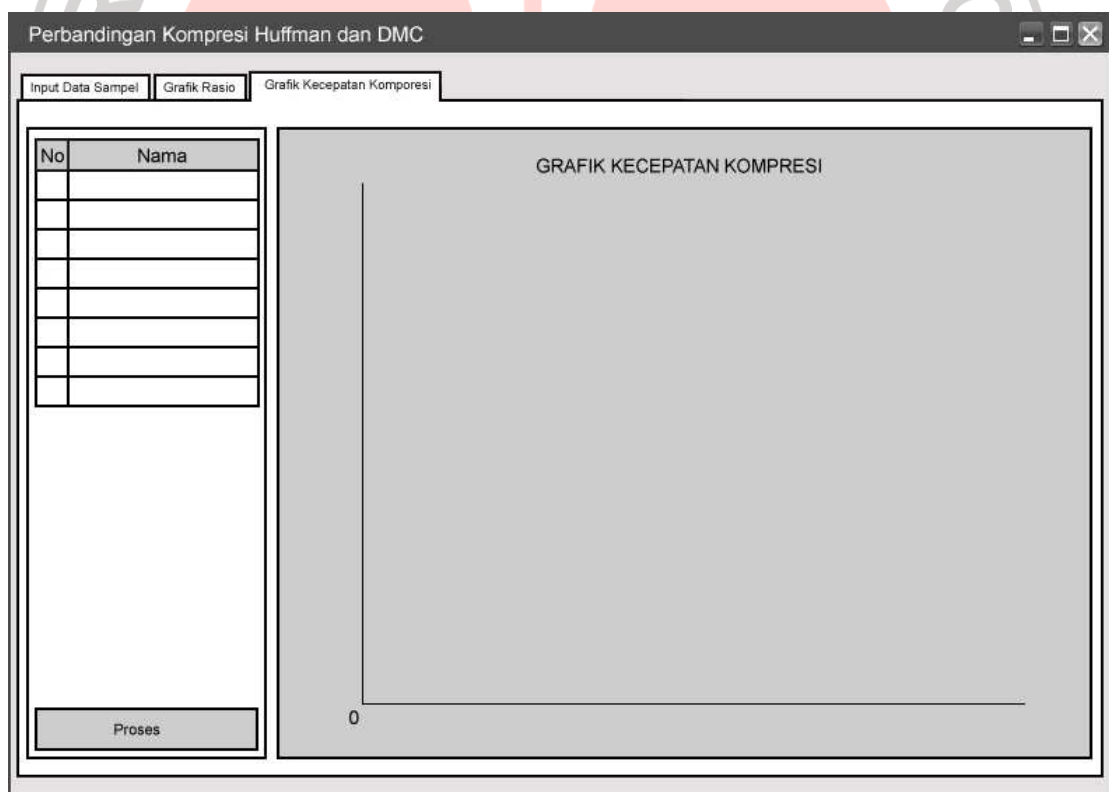
Gambar 4.3 Rancangan Tampilan Grafik Rasio Kompresi

Pada bagian kiri gambar terdapat tabel yang terdiri dari dua kolom, tabel ini akan terisi dengan mengklik tombol “Proses”. Kolom pertama adalah kolom

“No”, kolom ini berisi nomor urut file berdasarkan abjad. Kolom kedua adalah kolom “Nama”, kolom ini berisi nama-nama file yang terdapat pada tabel pada rancangan pertama.

Pada bagian kanan gambar akan ditampilkan grafik rasio kompresi. Grafik ini didapat dengan mengklik tombol “Proses”. Ordinatnya berupa nomor urut file berdasarkan abjad dan absisnya berupa kecepatan kompresi dalam persen.

Rancangan keempat adalah membuat tampilan untuk grafik perbandingan kecepatan kedua metode kompresi. Ilustrasi rancangan dapat dilihat pada Gambar (4.4).



Gambar 4.4 Rancangan Tampilan Grafik Kecepatan Kompresi

Pada bagian kiri gambar terdapat tabel yang terdiri dari dua kolom, tabel ini akan terisi dengan mengklik tombol “Proses”. Kolom pertama adalah kolom “No”, kolom ini berisi nomor urut file berdasarkan abjad. Kolom kedua adalah kolom “Nama”, kolom ini berisi nama-nama file yang terdapat pada tabel pada rancangan pertama.

Pada bagian kanan gambar akan ditampilkan grafik kecepatan kompresi. Grafik ini didapat dengan mengklik tombol “Proses”. Ordinatnya berupa nomor urut file berdasarkan abjad dan absisnya berupa rasio kompresi dalam *KB/s*.

4.2 Pembuatan Program

Bahasa pemrograman yang digunakan dalam pembuatan program ini adalah bahasa pemrograman *Delphi*. Dengan demikian, file kode program yang dibuat disimpan dengan ekstensi **.pas*. Setelah kode program tersebut dibuat, langkah selanjutnya adalah mengompilasi file-file tersebut sehingga akan dihasilkan file dengan ekstensi **.exe*. Dari file berekstensi **.exe* ini program aplikasi dapat langsung dijalankan

Semua kode *Delphi* yang dibuat memiliki peran dan fungsi tersendiri yang saling berhubungan antara satu file yang satu dengan yang lainnya. Dengan demikian, jika kode pada suatu file diubah, maka akan berpengaruh pada file-file yang lainnya. Kesalahan penulisan kode pada suatu file dapat mengakibatkan file yang berhubungan dengan file tersebut tidak dapat dikompilasi, akibatnya program tidak dapat dijalankan.

Adapun penjelasan dari file-file yang digunakan dalam program aplikasi ini adalah sebagai berikut:

1. *huffman.pas*

File ini berfungsi dalam proses kompresi dan dekompresi oleh metode kompresi Huffman.

2. *DMC.pas*

File ini berfungsi dalam proses kompresi dan dekompresi oleh metode kompresi *Dynamic Markov Compression*.

3. *UMain.pas*

File ini adalah file utama yang berfungsi untuk menjalankan program.

4. *dB.mdb*

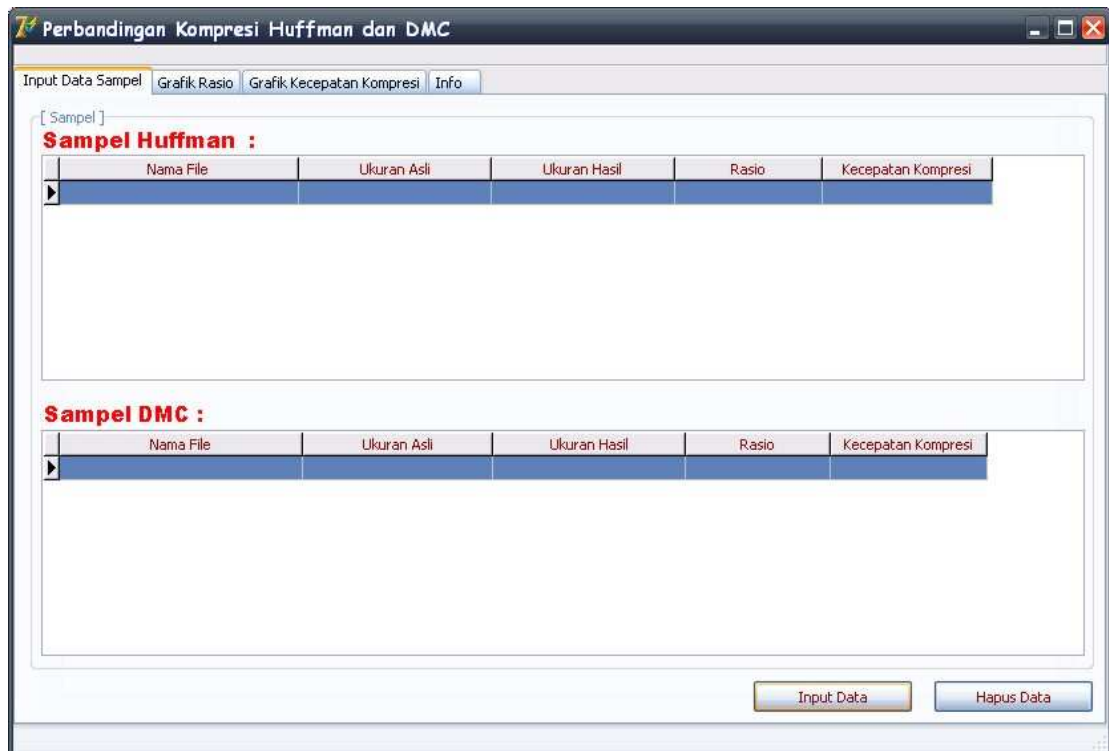
File basis data yang akan menyimpan data-data proses pengompresian file.

4.3 Pengujian Program

Pengujian adalah hal terpenting dalam menilai keberhasilan suatu program aplikasi, dan bertujuan untuk mengetahui kesalahan dan kekurangan pada sistem yang dibuat. Selanjutnya akan ditampilkan rincian program.

4.3.1 Menu Utama

Pada menu utama ditampilkan beberapa menu yang terdapat pada program. Menu-menu tersebut ditampilkan dalam bentuk *tab* yaitu *tab* Input Data Sampel, *tab* Grafik Rasio dan *tab* Grafik kecepatan Kompresi seperti ditampilkan pada Gambar (4.5).



Gambar 4.5 Menu Utama

4.3.1.1 Menu Input Data Sampel

Pada menu ini akan ditampilkan data-data proses pengompresian seperti nama file, ukuran asli, ukuran hasil dan rasio dan kecepatan kompresi. Di dalam menu tersebut terdapat beberapa tombol fungsi, yaitu:

1. Tombol “Input Data” : Untuk membuka submenu “Tambah Data Sampel”.
2. Tombol “Hapus Data” : Untuk menghapus data-data proses pengompresian file dari basis data.

Untuk memulai proses kompresi *input*-kan file yang akan dikompresi dengan mengklik tombol “Input Data”. Sehingga muncul submenu “Tambah Data

Sampel”, seperti ditampilkan pada Gambar (4.6). Gambar (4.6) menunjukkan proses kompresi untuk file *aeheur.dll*.



Gambar 4.6 Menu Tambah Data Sampel

Pada submenu ini terdapat beberapa tombol

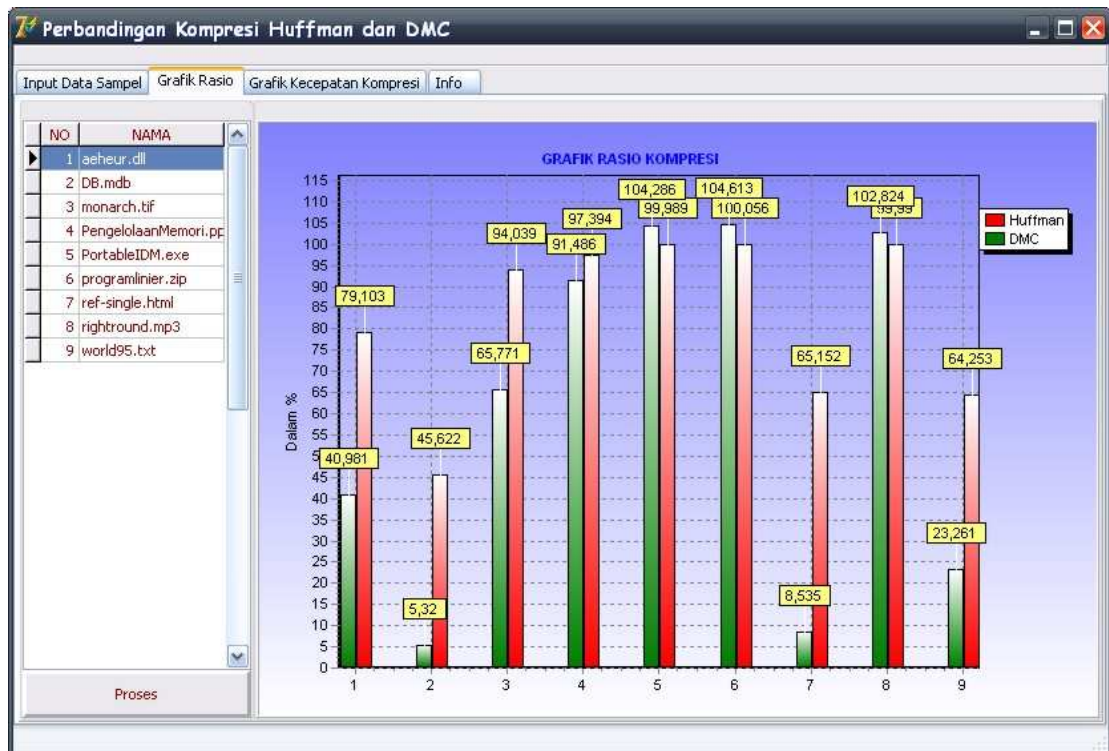
1. Tombol “Browse” : Untuk meng-*input* file yang akan dikompresi
2. Tombol “Proses” : Untuk mengkompresi file.
3. Tombol “Simpan” : Untuk menyimpan data-data yang berkaitan dengan proses kompresi file ke dalam basis data, dan data-data tersebut akan ditampilkan pada menu utama.
4. Tombol “Batal” : Untuk keluar dari menu Tambah Data Sampel dan kembali ke menu Input Data Sampel.

File yang telah dikompresi secara otomatis akan tersimpan di folder kompresi. File hasil kompresi metode kompresi Huffman akan disimpan dengan ekstensi *.huf*, sedangkan file hasil kompresi metode *Dynamic Markov Compression* akan disimpan dengan ekstensi *.dmc*.

4.3.1.2 Menu Grafik Rasio Kompresi

Pada menu ini akan ditampilkan salah satu data proses kompresi yaitu rasio kompresi dalam bentuk grafik. Bagian kiri adalah daftar file-file yang telah dikompresi, dan bagian kanan menampilkan grafik rasio, sehingga dapat dilihat perbedaan rasio antara metode kompresi Huffman dan *Dynamic Markov Compression*. Pada menu ini terdapat satu tombol fungsi yaitu tombol Proses yang berfungsi untuk memproses basis data menjadi grafik. Tampilan menu ini dapat dilihat pada gambar (4.7).

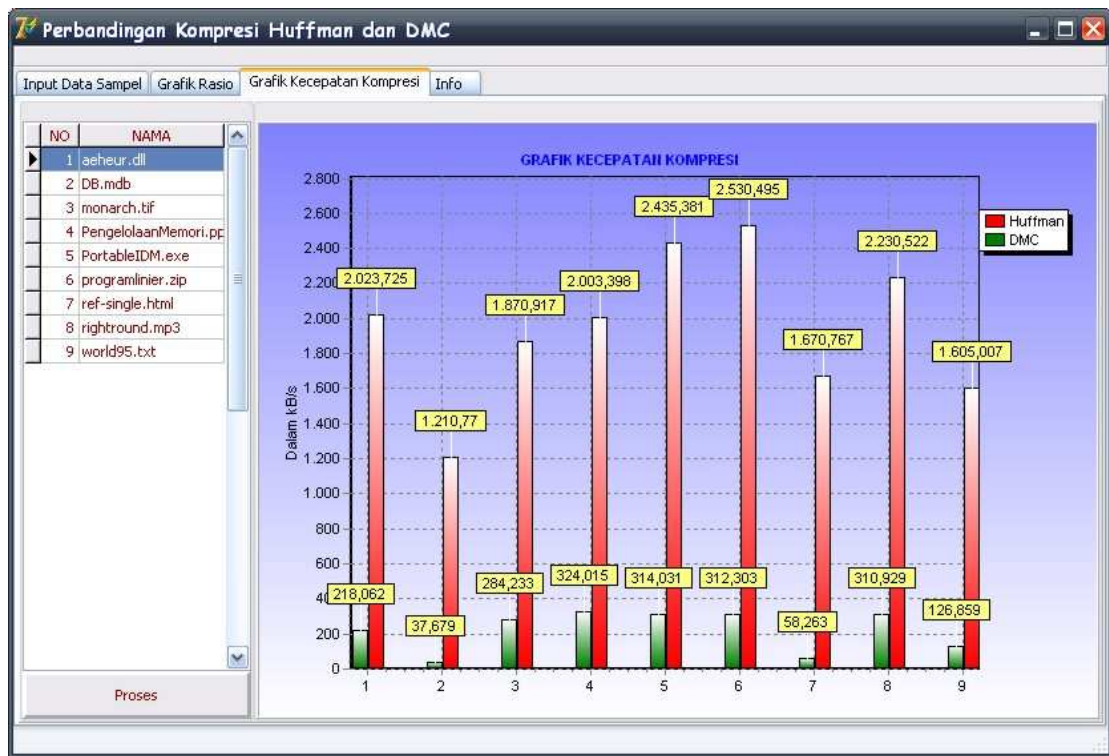
Grafik pada gambar (4.7) didapat setelah mengkompresi beberapa file. Nama file-file tersebut ditampilkan pada tabel di bagian kiri gambar.



Gambar 4.7 Menu Grafik Rasio

4.3.1.3 Menu Grafik Kecepatan Kompresi

Pada menu ini akan ditampilkan salah satu data proses kompresi yaitu kecepatan kompresi dalam bentuk grafik. Bagian kiri adalah daftar file-file yang telah dikompresi, dan bagian kanan menampilkan grafik kecepatan kompresi, sehingga dapat dilihat perbedaan kecepatan kompresi antara metode kompresi Huffman dan *Dynamic Markov Compression*. Pada menu ini terdapat satu tombol fungsi yaitu tombol “Proses” yang berfungsi untuk memproses basis data menjadi grafik. Tampilan menu ini dapat dilihat pada gambar (4.8). Grafik pada gambar (4.8) didapat setelah mengkompresi beberapa file. Nama file-file tersebut ditampilkan pada tabel di bagian kiri gambar.



Gambar 4.8 Menu Grafik Kecepatan Kompresi

4.4 Perbandingan Metode Kompresi Huffman dan *Dynamic Markov Compression*

Dalam penelitian ini, digunakan 9 golongan besar kasus uji yang dipandang cukup memadai untuk mewakili sebagian besar tipe file yang ada, yaitu:

1. File aplikasi (*Excel, Acrobat Reader, Flash, Corel Draw, PowerPoint, Font Window, dan Help*).
2. File yang sebelumnya telah dikompresi.
3. File *object/* file biner (file *com*, file sistem, file hasil kompilasi *C, Pascal*, dan *Java*, file *DLL*)
4. File basis data (*Access, DBase, Paradox, MySQL*)

5. File *executable*, (file dengan ekstensi *.*exe*)
6. File gambar (file *jpg*, file *bitmap*, file *gif*, file *png*, file *tif*, file *ani*, dan file *wmf*)
7. File multimedia (file *asf*, file *mpeg/mpg*, file *mp3*, file *mov*, file *midi*, file *avi*, file *wav*)
8. File *source code* (*pascal*, *html*, *c*, *cpp*, *java*, *prolog*, *css*, *vbs*, *js*, *xml*, *php*, *lisp*)
9. File teks (file *rtf*, file *doc*, file *inf*, file *txt*, file *ini* – konfigurasi Windows)

File-file ini kemudian dikompresi dengan menggunakan kedua metode yaitu Huffman dan *Dynamic Markov Compression*. Berikut adalah data-data mengenai file sampel yang akan dikompresi.

Tabel 4.1
Data File Sampel

No	Nama File	Ekstensi File	Tipe File	Ukuran File (dalam <i>Byte</i>)
1	PengelolaanMemori	ppt	File aplikasi	1.285.632
2	programlinier	zip	File yang telah dikompresi.	4.385.418
3	aeheur	dll	File <i>object</i> / file biner	1.798.520
4	DB	mdb	File basis data	3.399.680
5	PortableIDM	exe	File <i>executable</i>	2.586.666
6	monarch	tif	File gambar	1.179.784
7	righround	Mp3	File multimedia	1.496.832
8	ref-single	html	File <i>source code</i>	2.402.858
9	world95	txt	File teks	3.005.020

Data-data mengenai proses kompresi Huffman untuk file-file pada tabel (4.1) dapat dilihat pada Tabel (4.2).

Tabel 4.2
Data Proses Kompresi Huffman

No	nama_file	Ukuran asli (dalam Byte)	Ukuran hasil kompresi (dalam Byte)	Rasio (dalam %)	Kecepatan (dalam kB/s)
1	aeheur.dll	1.798.520	1.422.679	79,103	2.023,725
2	DB.mdb	3.399.680	1.550.996	45,622	1.210,770
3	monarch.tif	1.179.784	1.109.454	94,039	1.870,917
4	PengelolaanMemori.ppt	1.285.632	1.252.124	97,394	2.003,398
5	PortableIDM.exe	2.586.666	2.586.375	99,989	2.435,381
6	programlinier.zip	4.385.418	4.387.878	100,056	2.530,495
7	ref-single.html	2.402.858	1.565.509	65,152	1.670,767
8	righround.mp3	1.496.832	1.496.680	99,990	2.230,522
9	world95.txt	3.005.020	1.930.824	64,253	1.605,007
Rata-rata				82,470	1.940,260
Standar Deviasi				19,125	397,882

Dari tabel di atas dapat dilihat bahwa rata-rata rasio kompresi untuk metode kompresi Huffman cukup besar yaitu 82,470%. Dan standar deviasi rasio kompresi untuk metode kompresi Huffman adalah 19,125%, standar deviasi yang cukup kecil menunjukkan bahwa rasio kompresi untuk metode ini cukup merata untuk semua jenis file.

Hasil kompresi dengan metode kompresi Huffman menunjukkan rasio yang baik pada file *object*, basis data, *source code* dan teks. Sedangkan untuk kompresi file gambar, file aplikasi, file executable, file hasil kompresi, dan file multimedia metode kompresi Huffman menunjukkan hasil yang buruk, bahkan dapat mencapai lebih dari 100 %. Metode kompresi Huffman bekerja optimal pada file-file berbentuk teks, hal ini dikarenakan file yang berbentuk teks mengandung banyak simbol dan banyak terjadi pengulangan simbol, sehingga kode Huffman bisa dibuat menjadi lebih pendek. Untuk file-file lain seperti misalnya file multimedia, jarang terjadi pengulangan simbol, sehingga kode Huffman yang

dibuat cenderung lebih panjang. Hal ini dapat menyebabkan rangkaian bit yang dihasilkan menjadi lebih panjang daripada rangkaian bit file sebelum dikompresi.

Rata-rata kecepatan kompresi menunjukkan bahwa metode kompresi Huffman mengompresi dengan kecepatan yang sangat baik yaitu 1.940,260 *kB/s*. Standar deviasi kecepatan kompresi untuk metode kompresi Huffman adalah 397,882 *kB/s* Standar deviasi yang cukup kecil menunjukkan bahwa kecepatan kompresi metode kompresi Huffman cukup merata untuk semua jenis file.

Tabel selanjutnya yang akan ditampilkan adalah tabel data-data mengenai proses kompresi *Dynamic Markov Compression* untuk file-file pada tabel (4.1). Data-data tersebut dapat dilihat pada Tabel (4.2).

Tabel 4.3
Data proses kompresi *Dynamic Markov Compression*

No	nama_file	Ukuran asli(dalam Byte)	Ukuran hasil kompresi (dalam Byte)	Rasio (dalam %)	Kecepatan (dalam <i>kB/s</i>)
1	aeheur.dll	1.798.520	737.050	40,981	218,062
2	DB.mdb	3.399.680	180.860	5,320	37,679
3	monarch.tif	1.179.784	775.956	65,771	284,233
4	PengelolaanMemori.ppt	1.285.632	1.176.176	91,486	324,015
5	PortableIDM.exe	2.586.666	2.697.522	104,286	314,031
6	programlinier.zip	4.385.418	4.587.728	104,613	312,303
7	ref-single.html	2.402.858	205.086	8,535	58,263
8	rightround.mp3	1.496.832	1.539.101	102,824	310,929
9	world95.txt	3.005.020	698.993	23,261	126,859
Rata-rata				58,806	220,905
Standar Deviasi				40,149	109,875

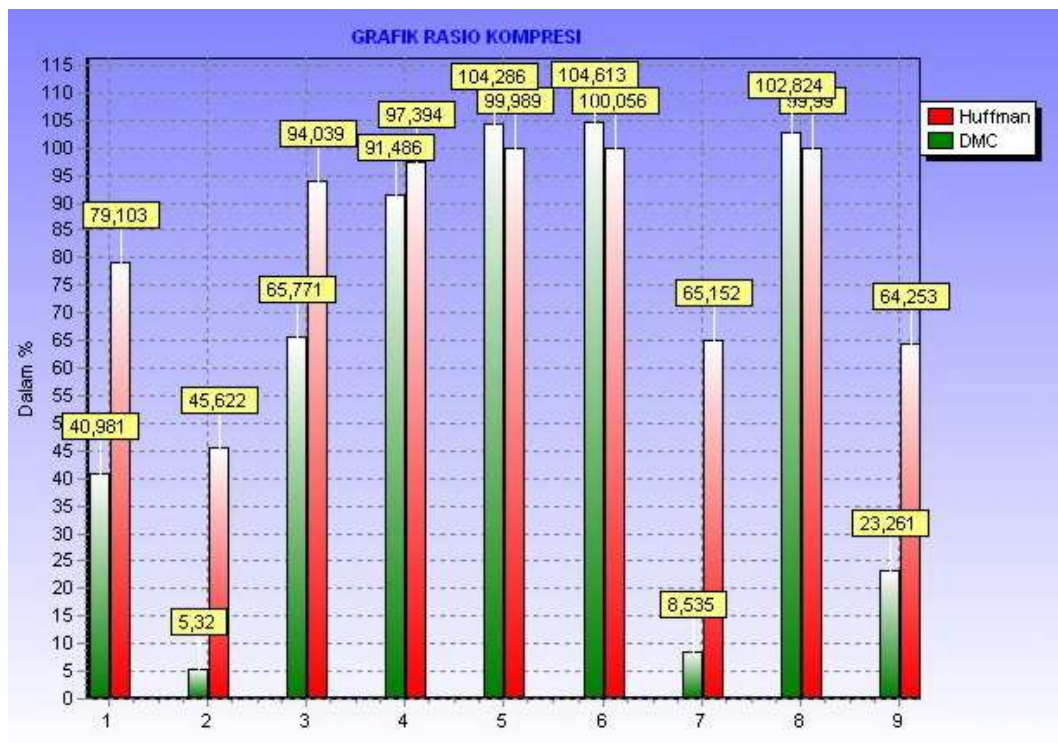
Dari tabel di atas dapat dilihat bahwa rata-rata rasio kompresi untuk metode kompresi Huffman cukup kecil yaitu 58,806 %. Dan standar deviasi rasio kompresi untuk metode kompresi Huffman adalah 40,149 %, standar deviasi yang

sangat besar menunjukkan bahwa rasio kompresi untuk metode ini tidak merata untuk setiap jenis file. Terdapat perbedaan yang sangat signifikan antara satu file dengan file yang lain.

Hasil kompresi dengan metode *Dynamic Markov Compression* menunjukkan hasil yang sangat baik pada file *object*, file basis data, file gambar, file *source code* dan file teks, yaitu dapat mencapai kurang dari 50%. Sama dengan metode kompresi Huffman untuk kompresi file aplikasi, file *executable*, file hasil kompresi, dan file multimedia, metode *Dynamic Markov Compression* menunjukkan hasil yang buruk yaitu dapat mencapai lebih dari 100%. Metode ini mengalami penurunan kinerja untuk file-file yang berbentuk hasil kompresi, begitu juga file multimedia karena file multimedia umumnya merupakan file hasil kompresi.

Rata-rata kecepatan kompresi menunjukkan bahwa metode *Dynamic Markov Compression* mengompresi dengan kecepatan yang tidak terlalu baik yaitu 220,905 *kB/s*. Standar deviasi kecepatan kompresi untuk metode kompresi Huffman adalah 109,875 *kB/s*. Standar deviasi yang cukup kecil menunjukkan bahwa kecepatan kompresi metode *Dynamic Markov Compression* cukup merata untuk semua jenis file. Namun kecepatan kompresi mengalami penurunan yang cukup signifikan pada file basis data dan file *source code*. Metode *Dynamic Markov Compression* mengorbankan kecepatan kompresi untuk mendapatkan rasio hasil kompresi yang baik.

Grafik perbandingan rasio kompresi dan kecepatan kompresi dari kedua metode tersebut dalam masing-masing kasus uji ditunjukkan secara lengkap dalam Gambar (4.9) dan (4.10).



Gambar 4.9 Grafik Rasio Kompresi

Dari gambar (4.9) dapat dilihat bahwa rasio kompresi untuk metode *Dynamic Markov Compression* cenderung lebih baik daripada metode kompresi Huffman. Walaupun secara rata-rata metode *Dynamic Markov Compression* menunjukkan hasil yang lebih baik, beberapa tipe file menunjukkan hasil yang tidak lebih baik daripada rasio hasil metode kompresi Huffman. Namun terdapat beberapa tipe file yang menunjukkan perbedaan yang jauh antara rasio kompresi metode Huffman dan *Dynamic Markov Compression*.



Gambar 4.10 Grafik Kecepatan Kompresi

Dari gambar (4.10) dapat dilihat bahwa kecepatan mengkompresi metode kompresi Huffman jauh lebih baik dari pada *Dynamic Markov Compression*. Hal ini dikarenakan metode *Dynamic Markov Compression* bekerja pada level bit sedangkan metode kompresi Huffman bekerja pada level *Byte*.