



## Penerapan Algoritma Prefix Code Pada Kompresi File Gambar

Esteria Novebri Simanjuntak

Prodi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia  
Email: [esteria011inovia@gmail.com](mailto:esteria011inovia@gmail.com)

### ARTICLE INFO

#### Article History

Received : Nov 24, 2020  
Accepted : Mar 29, 2021  
Published : Mar 31, 2021

### CORRESPONDENCE

Email: [esteria011inovia@gmail.com](mailto:esteria011inovia@gmail.com)

### A B S T R A K

Kompresi data adalah suatu cara pengubahan data dari sekumpulan karakter diubah menjadi suatu kode dengan tujuan menghemat tempat penyimpanan. Bersamaan dengan berkembangnya jaman, perangkat yang semakin maju mengakibatkan ukuran data yang dihasilkan semakin meningkat dan terus bertambah besar, apabila ukuran datanya besar maka dapat menyebabkan pemborosan tempat penyimpanan data dan pemborosan pemakaian kuota pada saat sharing karena itu diperlukan suatu teknik untuk mengubah ukuran data tersebut agar menjadi lebih kecil. Penulis akan menerapkan cara kerjanya pada file gambar. Dalam penelitian ini dilakukan penerapan untuk memperkecil suatu ukuran file gambar kedalam bentuk aplikasi dengan algoritma Prefix code. Algoritma Prefix Code adalah salah satu algoritma kompresi data yang dapat memampatkan file gambar dengan yang mempunyai tujuan untuk memperkecil ukuran data pada gambar. Ada banyak jenis metode ataupun algoritma dalam kompresi gambar, tetapi dalam algoritma ini penulis hanya ingin mengetahui proses kompresi menggunakan algoritma Prefix Code menggunakan kode C1 dimana penulis telah berhasil mengkompres suatu file gambar dengan algoritma prefix code menggunakan kode C1 memiliki kemampuan mengkompres data gambar sebesar 37,5% pada suatu file gambar.

**Kata Kunci:** Kompresi; File; Gambar; Algoritma; Prefix Code

### A B S T R A C T

Data compression is a way of changing data from a set of characters to be converted into a code with the aim of saving storage space. Along with the development of the era, the more advanced devices result in the resulting data size increasing and continuing to increase, if the data size is large it can cause waste of data storage space and waste of quota usage when sharing because it requires a technique to change the size of the data so that to be smaller. The author will apply how it works on image files. In this study, an application to reduce the size of an image file into an application with the Prefix code algorithm. The Prefix Code algorithm is a data compression algorithm that can compress image files with the aim of reducing the size of the data in the image. There are many types of methods or algorithms in image compression, but in this algorithm the writer only wants to know the compression process using the Prefix Code algorithm using the C1 code where the author has successfully compressed an image file with the prefix code algorithm using the C1 code has the ability to compress 37 image data, 5% in an image file.

**Keywords:** Compression; File; Image; Algorithm; Prefix Code

## 1. PENDAHULUAN

Pada saat ini perkembangan ilmu pengetahuan dan teknologi yang semakin pesat membuat file citra dua dimensi menjadi sangat dibutuhkan. Oleh karena itu, dapat dilihat dari kualitas citra dan semakin tingginya resolusi kamera digital saat ini sehingga menghasilkan resolusi citra yang tinggi dengan ukuran yang tinggi. Sementara itu untuk penyimpanan citra sendiri memerlukan *storage* yang cukup besar dan sehingga membuat *storage* cepat penuh, saat pertukaran data informasi juga menguras waktu dan *bandwidth*

Secara umum citra ialah suatu gambar, foto ataupun tampilan dua dimensi lainnya yang menggambarkan suatu visualisasi objek. Citra sebagai keluaran suatu sistem 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 penyimpan. Kompresi citra adalah waktu pengiriman data pada saluran komunikasi lebih singkat dan membutuhkan ruang memori dalam *storage* yang lebih sedikit[1].

Kompresi data adalah proses mengubah aliran data *input* (sumber aliran atau data mentah asli) menjadi aliran data lain (*output*, atau aliran terkompresi) yang memiliki ukuran lebih kecil. Teknik kompresi yang ada sekarang

memungkinkan citra dikompresi sehingga ukurannya menjadi jauh lebih kecil dari pada ukuran asli. Sayangnya, rasio yang dihasilkan kompresi citra sangat rendah karna banyaknya aplikasi yang memerlukan kompresi tanpa cacat, seperti pada aplikasi radiografi, kompresi citra hasil diagnosa medis atau gambar satelit, dimana kehilangan gambar sekecil apapun akan menyebabkan hasil yang tak diharapkan.

Algoritma *Prefix code* adalah jenis sistem kode yang dibedakan berdasarkan kepemilikannya atas "properti awalan", yang mensyaratkan bahwa tidak ada keseluruhan kata kode dalam sistem yang merupakan awalan (segmen awal) dari setiap kata kode lain dalam sistem [2].

## 2. METODOLOGI PENELITIAN

### 2.1 Perancangan Aplikasi

Perancangan aplikasi adalah sebuah proses penerapan suatu sistem yang akan dikerjakan dengan menggunakan teknik yang bervariasi guna untuk melaksanakan suatu fungsi bagi pengguna aplikasi agar mencapai suatu sasaran yang dituju . Perancangan aplikasi lebih di asumsikan sebagai perancangan yang melibatkan logika modul-modul yang akan dibuat dalam perangkat lunak. Perancangan mengenai aplikasi melibatkan perancangan logika dan algoritma dari suatu perangkat lunak[3].

### 2.2 Kompresi

Kompresi adalah pengubahan data yang berupa kumpulan karakter menjadi bentuk kode dengan tujuan untuk menghemat kebutuhan tempat penyimpanan dan waktu transmisi data[4]. Data-data yang bergabung dalam setiap kelompok dapat dianggap berkarakter sama/mirip sehingga data-data dalam kelompok yang sama dapat dikompresi dengan diwakili oleh indeks prototipe dari setiap kelompok. Setiap objek direpresentasikan dengan indeks prototipe yang dikaitkan dengan sebuah kelompok[5]. Secara umum, metode kompresi data dapat dibagi kedalam dua kelompok yaitu kompresi *lossy* dan kompresi *lossless*. Teknik kompresi data yang bersifat *lossy* mengijinkan terjadinya kehilangan sebagian data tertentu dari pesan tersebut, sehingga dapat menghasilkan rasio kompresi yang tinggi. Apabila citra terkompresi direkonstruksi kembali maka hasilnya tidak sama dengan citra aslinya, tetapi informasi yang tekandung tidak sampai berubah atau hilang. Kompresi data *lossy* ini efektif jika diterapkan pada penyimpanan data analog yang digitalisasi seperti gambar, vidio, dan suara[6]. Teknik kompresi data yang bersifat *lossless* ini ialah mempertahankan kesempurnaan data sesuai dengan data aslinya, sehingga hanya terdapat proses *coding* dan *decoding*, tidak terdapat proses kuantitasi. Citra hasil proses kompresi dapat dikembalikan secara sempurna menjadi citra asli, tidak terjadi kehilangan informasi. Oleh karna itu metode ini disebut juga *error free compression*. Kompresi ini dapat diterapkan pada berkas yang berbasis data ( *database*), *spread sheet*, berkas *word processing*, citra biomedis, dan lain sebagainya [6].

### 2.3 Format File Gambar

Format *file* gambar yang digunakan saat ini terdiri dari beberapa jenis. Format- format ini digunakan untuk menyimpan gambar dalam sebuah *file*. Setiap format memiliki karakteristik masing- masing yaitu *Bitmap (.bmp)*, *Tagged Image Format(.tif, .tiff)*, *Portable Network Graphics(.png)*, *Joint Photographic Experts Group (JPEG)*, *Graphics Interchange Format (.gif)*.

### 2.4 Algoritma Prefix Code

*Prefix Code* adalah jenis sistem kode yang dibedakan oleh kepemilikan "*prefix property*", yang mensyaratkan bahwa tidak ada kata kode keseluruhan dalam sistem yang merupakan awalan (awal segmen) dari kode kata lain dalam sistem, Terdapat empat kode dalam *prefix code* yaitu C1, C2, C3, C4 dan kode yang penulis gunakan adalah kode C1 [2]. Untuk mendapatkan nilai kode C1, C2, C3, dan C4 sebelumnya harus mengetahui nilai dari *Unary Code*,

$B(n)$  dan  $\bar{B}(n)$ . *Unary Code* dari bilangan positif  $n$  didefinisikan sebagai  $n - 1$  diikuti oleh satu 0 atau alternatifnya seperti angka nol  $n - 1$  diikuti dengan satu.

$B(n)$  untuk menunjukkan representasi biner dari bilangan bulat  $n$ , sementara  $\bar{B}(n)$  untuk menunjukkan nilai  $B(n)$  tanpa bit dengan menghilangkan nilai bit 1 depan setiap bilangan  $B(n)$ . Setelah menemukan nilai  $B(n)$  dan  $\bar{B}(n)$  barulah mencari nilai kode C1 dengan contoh  $n = 5 = 101_2$ , ukuran  $B(5)$  adalah 3 lalu mulai dengan kode unari 110 (atau 001) dan menambahkan  $\bar{B}(5) = 01$ , dengan demikian kode lengkapnya adalah 110 | 01 (atau 001 | 10) Untuk mencari nilai kode C2, C3, dan C4 bisa dilihat pada tabel 1 dibawah ini.

Tabel 1. Tabel Ketentuan Prefix Code

n	Unary	B(n)	$\bar{B}(n)$	C1	C2	C3	C4
1	0	1		0	0	0	0
2	10	10	1	10 0	100	100 0	10 0
3	110	11	00	10 1	110	100 1	11 0
4	1110	100	01	110 00	10100	110 00	10 100 0
5	11110	101	10	110 01	10110	110 01	10 101 0

6	111110	110	11	110 10	11100	110 10	10 110 0
7	1111110	111	000	110 11	11110	110 11	10 111 0
8		1000	001	1110 000	1010100	10100 000	11 1000 0
9		1001	010	1110 001	1010110	10100 001	11 1001 0
10		1010	011	1110 010	1011100	10100 010	11 1010 0
11		1011	100	1110 011	1011110	10100 011	11 1011 0
12		1100	101	1110 100	1110100	10100 100	11 1100 0
13		1101	110	1110 101	1110110	10100 101	11 1101 0

Sumber : David Solomon, Giovanni Motta, 2010[2]

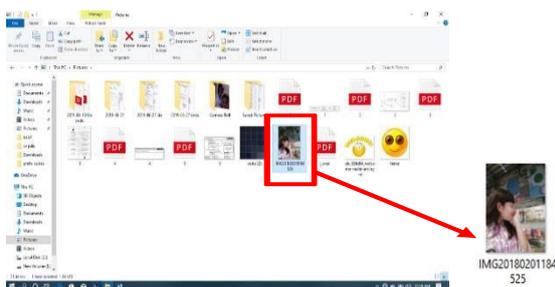
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisa Masalah

Analisa terhadap sistem merupakan tahap yang sangat penting untuk mengetahui proses yang terjadi di dalam aplikasi yang akan dirancang, dalam hal ini peneliti menggunakan algoritma *prefix code* untuk mengkompresi *file* gambar. Pengompresan *file* berguna untuk mengurangi ruang penyimpanan serta mempercepat proses pengiriman suatu *file*. Dalam penelitian ini yang akan dibahas adalah bagaimana cara kerja algoritma *prefix code* dalam mengkompresi file gambar dengan menggunakan analisis kompresi. File gambar yang akan digunakan dalam penelitian ini adalah *file* \*.Jpg. File gambar yang berekstensi \*.Jpg mempunyai ukuran yang cukup besar, semakin besar ukuran gambar dan kualitas gambar, maka semakin besar pula tempat penyimpanannya, dan proses transmisi yang dibutuhkan juga semakin besar. Dalam melakukan proses kompresi *file* gambar sebelumnya harus dilakukan analisa terhadap *file* gambar.

#### 3.2 Kompresi Dengan Algoritma Prefix Code

Penulis akan mengkompresi sebuah gambar dengan menggunakan algoritma *Prefix code*. Sebelum *file* dikompresi, terlebih dahulu dilakukan pembacaan biner yang terdapat pada *file* gambar untuk mendapatkan data berupa data biner. Membaca biner yang terdapat pada *file* gambar menggunakan aplikasi *Binery Viewer* untuk mencari nilai biner pada file gambar. Berikut adalah contoh file gambar yang akan di kompresi pada sampel C:/IMG20180201184525 dibawah ini :



Gambar 1. File Gambar

Berdasarkan contoh diatas terdapat nilai *biner viewer*, adapun nilai *hexadecimal* dari contoh file gambar pada sampel C:/IMG20180201184525 tersebut adalah :

Tabel 2. Tampilan Nilai Hexal Berdasarkan *Biner Viewer*

00	01	00	00	00	82	01	28
00	03	00	00	00	01	00	02

Bila dikodekan menggunakan kode C1, langkahnya sebagai berikut :

1. Berdasarkan pada tabel di atas, didapatkan beberapa nilai *pixel* yang sama. Sebelum proses kompresi citra, langkah awal adalah membaca nilai *pixel* citra kemudian membuat tabel nilai *pixel* yang diurutkan dari nilai frekuensi terbesar (nilai *pixel* yang sama) ke terkecil. Urutan nilai *pixel* dapat dilihat pada tabel di bawah ini :

Tabel 3. Pixel Yang Belum Dikompresi

No	Hexal	Biner	Frek	Bit	Frek x Bit
1	00	00000000	9	8	72
2	01	00000001	3	8	24
3	82	10000010	1	8	8
4	28	00101000	1	8	8
5	03	00000011	1	8	8
6	02	00000010	1	8	8

No	Hexal	Biner	Frek	Bit	Frek x Bit
				Total Bit	128 bit

Proses selanjutnya adalah melakukan kompresi nilai dari citra *hexal* sampel dengan nilai kode *C1* yang di dapat dari tabel diatas. Adapun Proses kompresi citra sampel dapat dilihat pada tabel di bawah ini :

**Tabel 4.** Hexal Yang Sudah Dikompresi Dengan Kode C1 Pada Algoritma Prefix Code

n	Nilai Hexal	Frek	Codeword (C1)	Bit	Frek x Bit
1	00	9	0	1	9
2	01	3	100	3	9
3	82	1	101	3	3
4	28	1	11000	5	5
5	03	1	11001	5	5
6	02	1	11010	5	5
				Total Bit	36
					Bit

a. Ratio of compression (Rc)

RC = Jumlah bit sebelum dikompresi / Jumlah bit sesudah dikompresi

$$RC = 128$$

$$RC = 2,6666$$

b. Compression ration (Cr)

CR = Ukuran data sesudah dikompresi / Ukuran data sebelum dikompresi

$$C_R = \frac{6 \text{ byte}}{16 \text{ byte}} \times 100\%$$

$$CR = 37,5\%$$

c. Space Savings (SS) SS

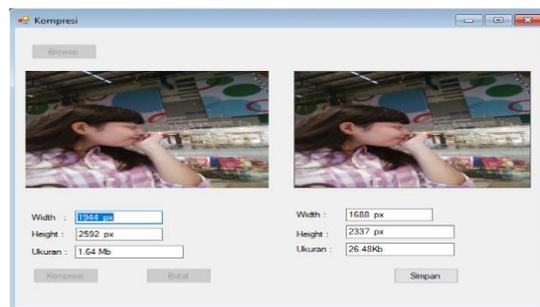
$$SS = 100\% - CR$$

$$SS = 100\% - 37,5\% = 62,5\%$$

### 3.3 Implementasi

1. *Form* Kompresi

*Form* kompresi yaitu aplikasi yang dirancang untuk mengkompresi gambar atau memperkecil ukuran gambar dari gambar aslinya.



**Gambar 2.** Form Kompresi

2. *Form* Dekompresi

*Form* dekomposisi yaitu aplikasi yang dirancang untuk mengembalikan data gambar yang sudah dikompres ke data gambar semula atau data gambar aslinya.



**Gambar 3.** Form Dekompresi

#### 4. KESIMPULAN

Hasil penelitian yaitu untuk proses dekompresi akan dilakukan pada tahapan pengurangan bit akhir gambar dan sistem akan melakukan proses dekompresi dengan menggunakan Algoritma *Prefix Code*. Pada form kompresi sistem akan melakukan proses kompresi dengan cara memilih gambar yang akan di kompresi dengan menggunakan Algoritma *Prefix Code* dan akan tersimpan sesuai tempat penyimpanan yang dipilih.

#### REFERENCES

- [1] T. Sutoyo, E. Mulyanto, V. Suhartono, O. D. Nurhayati, and Wijanarto, *Teori Pengolahan Citra Digital.pdf*. Yogyakarta: ANDI, 2009.
- [2] M. David, Salomon dan Giovanni, *Handbook of Data Compression*. Springer London Dordrech Heidelberg New York: Springer, 2010.
- [3] S. Rizky, *Konsep Dasar Rekayasa Perangkat Lunak (Software Reengineering)*. Jakarta: Prestasi Pustaka, 2011.
- [4] Juansyah Andi, "Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System ( A-GPS ) Dengan Platfrom Android," *J. Ilm. Komput. dan Inform.*, vol. 1, no. 1, pp. 1–8, 2015.
- [5] S. D. Nasution, "Perancangan Aplikasi Kompresi File Teks Dengan," vol. 1, no. 1, pp. 154–156, 2016.
- [6] E. Prasetyo, *DATA MINING - Konsep dan Aplikasi Meggunakan MATLAB*, 11 ed. Andi Offset, 2012.