

# Kompresi Data Menggunakan Metode FELICS (*Fast Efficient and Lossless Images Compression System*) Terhadap Citra PNG (*Portable Network Graphics*)

Yan Daniel Hutapea<sup>1</sup>, Nur Wulan<sup>2\*</sup>

<sup>1,2</sup> Teknik Informatika, Universitas Harapan Medan, Medan, Indonesia

<sup>1</sup>yandanielhutapea88@gmail.com, <sup>2</sup>nurwulanstth@gmail.com

**Abstrak**—Pada saat ini perkembangan ilmu pengetahuan dan teknologi informasi yang semakin pesat membuat file citra dua dimensi menjadi sangat dibutuhkan. Pengiriman file baik berupa file teks maupun file citra merupakan hal yang sudah biasa dilakukan. Pengiriman file citra dengan ukuran file yang besar membutuhkan waktu yang lebih lama dan bandwidth lebih besar. Untuk mengatasi hal ini perlu dilakukan kompresi atau pemampatan terhadap file citra tersebut. Metode FELICS merupakan salah satu metode yang dapat digunakan untuk kompresi file citra, Metode FELICS menyajikan sistem yang lebih sederhana untuk kompresi citra, berjalan lebih cepat dan hanya kehilangan efisiensi kompresi yang minimal. Berdasarkan hasil uji coba terhadap lima file .png dengan ukuran masing-masing file 5.249.494 byte menjadi 2.041.208 byte, 3.124.201 byte menjadi 1.060.781 byte, 10.473.459 byte menjadi 3.065.010 byte, 804.728 byte menjadi 200.215 byte, 503.111 byte menjadi 158.711 byte. Uji coba ini menghasilkan compression rasio antara 24,87% sampai dengan 38,00%.

**Kata Kunci:** FELICS, Kompresi, Citra.

**Abstract**—At present, the rapid advancement of sciences and information technology has made two-dimensional image files highly demanded. In this era, sending files, whether in text or image format, has become a common practice. Sending large image files requires more time and bandwidth. To address this, compression is necessary for image files. The FELICS method is one of the techniques that can be used for image file compression. FELICS present a simpler system for image compression, operates faster, and only incurs minimal compression efficiency loss. Based on test results for five .png files with sizes of 5.249.494 bytes reduced to 2.041.208 bytes, 3.124.201 bytes reduced to 1.060.781 bytes, 10.473.459 bytes reduced to 3.065.010 bytes, 804.728 bytes reduced to 200.215 bytes, and 503.111 bytes reduced to 158.711 bytes. This results in compression ratios ranging from 24.87% to 38.00%.

**Keywords:** FELICS, Compression, Image.

## 1. PENDAHULUAN

Pesatnya kemajuan ilmu pengetahuan dan teknologi membuat gambar dua dimensi semakin penting. Seperti yang sering diamati dalam kehidupan sehari-hari, banyak telepon selular yang menawarkan kamera resolusi tinggi untuk menghasilkan kualitas gambar yang lebih baik. Namun, kualitas gambar yang lebih tinggi juga memerlukan ruang penyimpanan yang lebih besar. Ketika ruang penyimpanan menjadi penuh, maka memakan waktu selama proses pertukaran data. Sejalan dengan hal tersebut, muncul solusi dari permasalahan tersebut, yaitu melalui kompresi data. Tujuan kompresi adalah untuk memperkecil ukuran data. Kompresi data sangat penting saat ini karena menghemat penyimpanan data, mengurangi waktu transmisi data, dan meminimalkan kebutuhan bandwidth. Namun, prinsip dasar yang mendasari setiap metode tetap sama: mengompresi data dengan menghilangkan redundansi dari data asli. Citra digital merupakan representasi fungsi intensitas cahaya dalam bentuk diskrit pada bidang dua dimensi[1]. Kompresi berarti memperkecil ukuran atau satuan pembawa informasi dengan menggunakan bit atau satuan pembawa informasi yang lebih rendah dari representasi data yang tidak dikodekan dengan sistem pengkodean tertentu, sehingga data yang dikompresi menjadi lebih kecil sehingga dapat mempersingkat waktu dalam pengiriman data [2].

Gambar sebagai keluaran suatu sistem perekam data dapat berbentuk optik seperti foto, atau digital yang dapat langsung disimpan pada pita magnetik. Gambar adalah representasi dari suatu objek. Citra digital merupakan salah satu jenis bukti digital yang mempunyai risiko perubahan yang tinggi sehingga mengakibatkan kerusakan dan hilangnya informasi secara signifikan [3]. Gambar digital terdiri dari potongan-potongan kecil berukuran piksel yang membentuk suatu gambar yang dapat dilihat oleh mata manusia. Setiap piksel tidak hanya mewakili 1 titik pada suatu gambar, melainkan suatu bagian berupa kotak yang merupakan bagian terkecil (sel). Semakin tinggi resolusi gambar maka semakin tinggi pula tingkat detail gambar tersebut [4]. Citra merupakan kumpulan titik-titik yang mempunyai identitas tertentu sehingga membentuk suatu kesatuan utuh yang mempunyai makna, baik secara artistik maupun hakiki. Gambar yang baik adalah gambar yang dapat menampilkan keindahan gambar (artistik) serta kejelasan gambar untuk keperluan analisis dan keperluan lainnya (intrinsik)[5]. Gambar merupakan salah satu bentuk informasi yang dibutuhkan manusia selain teks, suara dan video. Informasi yang terkandung dalam

suatu gambar dapat diinterpretasikan berbeda-beda antara manusia satu dengan yang lainnya[1]. Grayscale merupakan suatu proses pengolahan citra untuk mengubah warna gambar yang mempunyai nilai RGB diubah menjadi gambar skala abu-abu[6]. Gambar adalah data yang berformat gambar dan biasanya mempunyai ukuran gambar yang besar, semakin bagus gambar yang dihasilkan maka semakin besar pula kapasitas gambarnya. Di era digitalisasi, banyak orang yang memberikan informasi melalui gambar karena informasi dengan menggunakan gambar lebih mudah dipahami dan efektif[7]. Pengolahan citra digital memiliki keunggulan antara lain: tidak merusak objek, pemrosesan cepat dan mudah[8]. Portable Network Graphics (PNG) adalah salah satu format untuk menyimpan gambar yang menggunakan metode kompresi tanpa kehilangan bagian apa pun dari gambar tersebut. Selain itu, format ini mendukung transparansi pada browser[9]. PNG adalah solusi kompresi gambar foto kuat yang berisi lebih banyak data warna dibandingkan jenis lainnya (24 bit RGB + alpha). Kompresi file ini menggunakan kompresi yang tidak menghilangkan data[10]. Format .png (Portable Network Graphics) merupakan format yang menggunakan metode kompresi lossless dan sering digunakan pada website karena ukurannya yang relatif lebih kecil sehingga mempercepat akses pada website itu sendiri, selain itu format ini juga mendukung transparansi pada browser[11]. Tujuan penelitian ini adalah untuk menganalisa prosedur kompresi dan mengetahui hasil kompresi terhadap citra dengan format .png. Metode FELICS berhasil melakukan kompresi gambar JPEG dengan kapasitas 1801 sebelum kompresi, dan setelah kompresi menghasilkan 1144 bit sehingga menghasilkan rasio kompresi sebesar 60%. Dari hasil implementasi, metode FELICS dapat diterapkan untuk mengkompresi gambar JPEG dengan berbagai ukuran kapasitas, dengan kecepatan kompresi rata-rata 0,4 detik[12].

## 2. METODE PENELITIAN

### 2.1 Metode FELICS

Metode FELICS merupakan singkatan dari Fast Efficient and Loseless Image Compression System. Metode ini merupakan metode kompresi yang dirancang untuk gambar skala abu-abu. Metode FELICS menghadirkan sistem kompresi gambar lossless yang lebih sederhana yang berjalan lebih cepat dan hanya kehilangan efisiensi kompresi minimal. Prinsip FELICS adalah mengkodekan piksel cache dengan kode ukuran variabel berdasarkan nilai dari dua piksel tetangga yang terlihat sebelumnya. Untuk mengkodekan gambar, langkah pertama adalah menghapus dua piksel pertama tanpa melakukan pengkodean, lalu ikuti langkah selanjutnya

Metode FELICS merupakan singkatan dari Fast Efficient and Loseless Image Compression System. Metode ini merupakan metode kompresi yang dirancang untuk gambar skala abu-abu. Metode FELICS menghadirkan sistem kompresi gambar lossless yang lebih sederhana yang berjalan lebih cepat dan hanya kehilangan efisiensi kompresi minimal. Prinsip FELICS adalah mengkodekan piksel cache dengan kode ukuran variabel berdasarkan nilai dari dua piksel tetangga yang terlihat sebelumnya. Untuk mengkodekan suatu gambar, langkah pertama adalah menghilangkan dua piksel pertama tanpa pengkodean, kemudian ikuti langkah selanjutnya, yaitu:

- 1) Memilih piksel P dan menemukan tetangga terdekatnya N1 dan N2.
- 2) Menghitung  $L = \min(N1 ; N2)$ ,  $H = \max(N1 ; N2)$ , dan  $\Delta = H - L$ 
  - a. If  $L \leq P \leq H$ , menggunakan satu bit untuk menyandikan *IN-RANGE*; lalu menggunakan kode biner yang disesuaikan untuk mengkodekan P-L di  $[0, \Delta]$ .
  - b. If  $P < L$ , menggunakan satu bit untuk menyandikan *OUT-OF-RANGE*, dan satu bit untuk menyandikan *BELOW-RANGE*. Kemudian menggunakan kode *Golomb-Rice* untuk mengkodekan integer non-negatif  $L - P - 1$ .
  - c. Jika  $P > H$ , menggunakan satu bit untuk menyandikan *OUT-OF-RANGE*, dan satu bit untuk menyandikan *ABOVE-RANGE*. Kemudian menggunakan kode Golomb Rice untuk mengkodekan integer non-negatif  $P - H - 1$ .

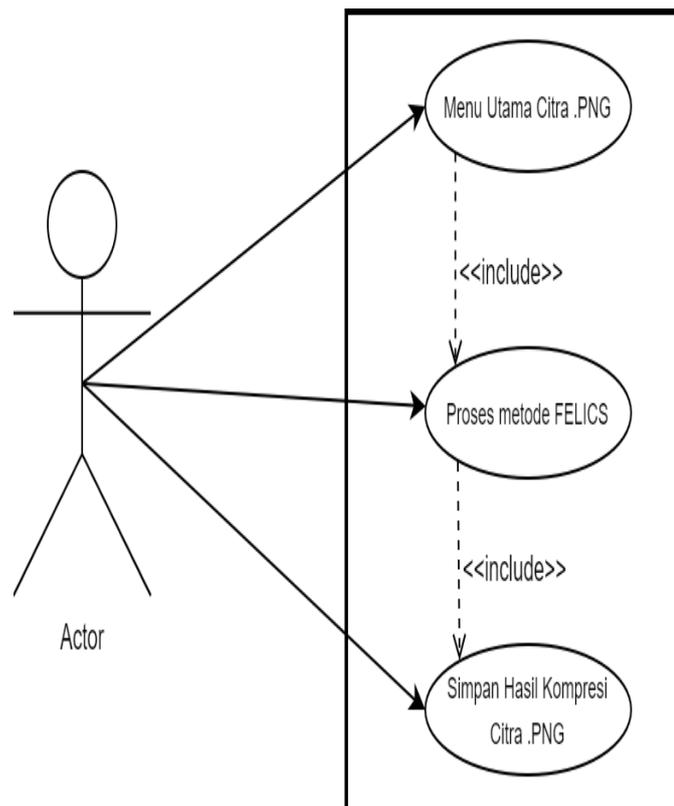
### 2.2 UML (Unified Modeling Language)

UML dapat digunakan untuk visualisasi, spesifikasi, konstruksi dan dokumentasi beberapa bagian sistem dalam perangkat lunak. Dengan kata lain, seperti seorang arsitek membuat dokumen cetak biru yang digunakan oleh perusahaan konstruksi untuk membangun sebuah bangunan, seorang arsitek perangkat lunak membuat diagram UML untuk membantu pemrogram/pengembang membangun perangkat lunak. [13]. *Unified Modeling Language* (UML) merupakan salah satu tools atau alat pemodelan yang paling dapat diandalkan dalam dunia pengembangan sistem berorientasi objek. UML adalah bahasa pemodelan terpadu yang dikembangkan oleh *Booch*, *Objects Modeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode ini menjadikan proses analisis dan desain dalam tahap yang berulang [14].

### 2.2.1 Use Case Diagram

Diagram ini sangat penting untuk mengatur dan memodelkan perilaku suatu sistem yang dibutuhkan dan diharapkan oleh pengguna. *Unified Modeling Language* adalah menentukan urutan dan langkah-langkah yang akan dilakukan oleh sistem. Setiap *use case* menyatakan spesifikasi perilaku (fungsionalitas) dari sistem yang dijelaskan yang diperlukan oleh aktor untuk memenuhi tujuannya [15].

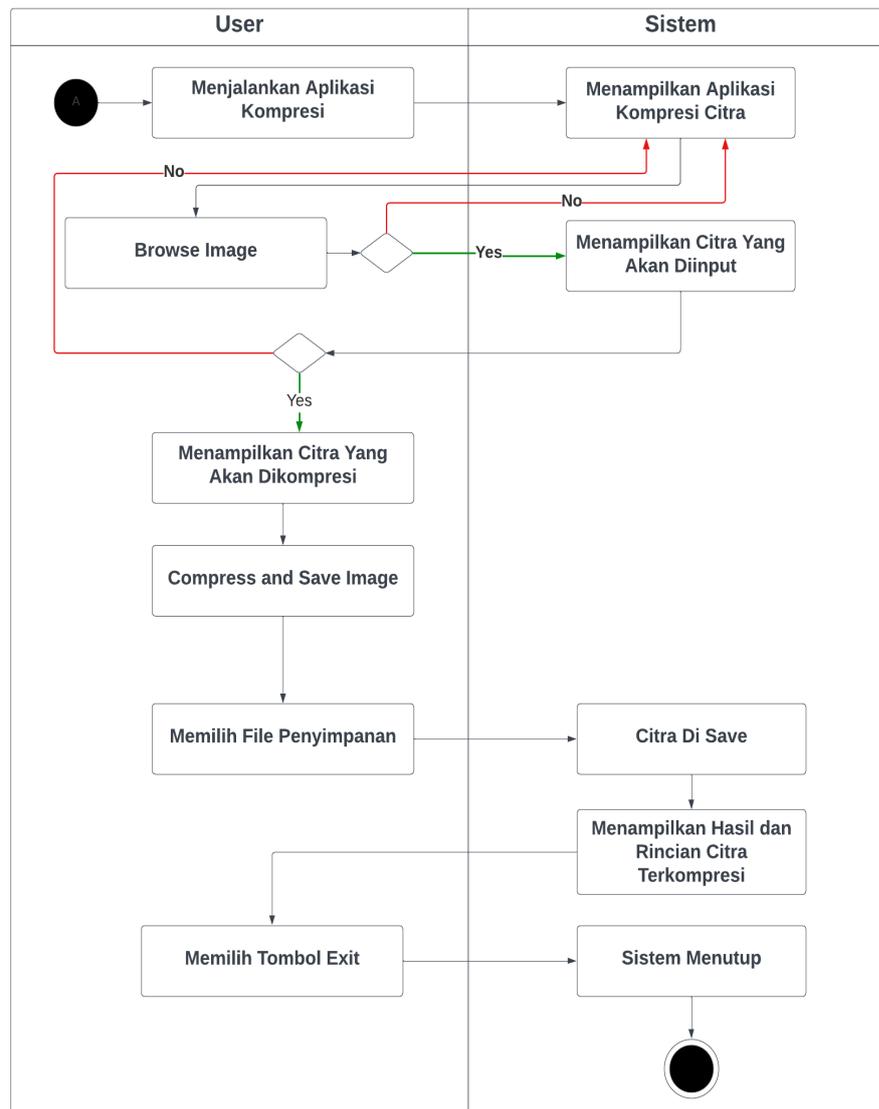
*Use case* merupakan gambaran fungsi suatu sistem dari sudut pandang atau sudut pandang pengguna sistem. Kasus penggunaan menentukan apa yang akan diproses oleh sistem dan komponennya. *Use case* bekerja dengan menggunakan skenario yang merupakan deskripsi urutan atau langkah-langkah yang menjelaskan apa yang dilakukan pengguna terhadap sistem dan sebaliknya [16].



**Gambar 1.** Use Case Diagram

### 2.2.2 Activity Diagram

Diagram Aktivitas menunjukkan aliran dari satu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram Aktivitas Kompresi merupakan gambaran umum mengenai aktivitas yang dilakukan oleh pengguna pada aplikasi kompresi citra digital yang dirancang pada penelitian ini ketika melakukan proses kompresi citra digital. Alur atau aktivitas dapat berupa rangkaian menu atau proses bisnis yang terdapat dalam sistem [18]. Setelah membuat model *Use Case*, setiap skenario pada *Use Case* akan dijelaskan lebih jelas pada diagram aktivitas. Activity diagram merupakan pemodelan yang menggambarkan sistem kerja suatu objek atau sistem, diagram aktivitas digambarkan dengan alur terstruktur dari proses kerja use case yang sedang diproses dari titik awal hingga titik akhir [20].



**Gambar 2.** Activity Diagram

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisis Permasalahan Sistem

*File* dokumen dengan *format .png* merupakan salah satu dari beberapa jenis *format file* yang sering digunakan di kehidupan sehari-hari. PNG merupakan *file* yang memiliki ukuran yang besar bila dibandingkan dengan format *file* lain pada umumnya, seperti pada *format .jpg* atau *format .gif*. Permasalahan yang dihadapi oleh pengguna teknologi informasi salah satunya adalah ukuran file yang besar membutuhkan suatu media penyimpanan yang besar pula. Serta memakan waktu saat dilakukan *transmisi* data. Ukuran data menjadi sesuatu yang sangat diperhatikan di era digital. Solusi untuk masalah di atas adalah dengan melakukan pemampatan/ kompresi data.

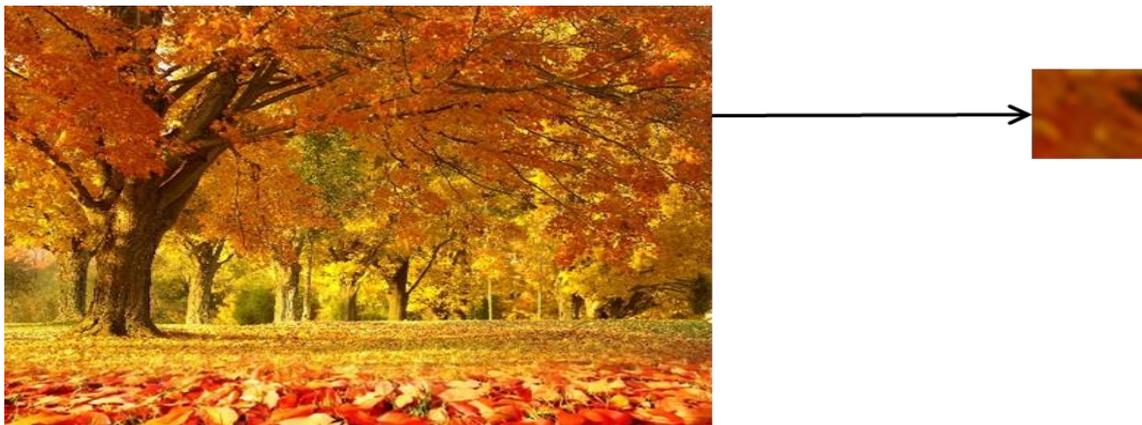
#### 3.2 Analisa FELICS

Untuk kepentingan hitungan manual, maka diambil satu objek, dimana objek tersebut akan diperkecil untuk dijadikan sebagai *sample*.

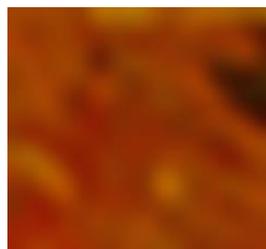


**Gambar 3.** Objek Citra File *.png*

Untuk memudahkan proses penerapan metode *FELICS*, selanjutnya diperkecil bagian citra *.png* yang akan dijadikan sebagai *sample*. Berikut tampilan citra *sample*, dan penulis menggunakan aplikasi Matlab untuk mengambil bagian kecil citra.



**Gambar 4.** Pengambilan Citra *Sample* Dengan Resolusi 16 x 16 Piksel



**Gambar 5.** Citra *sample* dengan resolusi 16 x 16 piksel

Dari citra *sample* dengan piksel 16 x 16 dilakukan proses pengambilan nilai citra *grayscale*. Tabel 1 adalah hasil citra *grayscale*.

**Tabel 1.** Nilai *Grayscale* 16 x 16

176	176	176	175	175	175	176	175	175	175	176	178	179	179	177	175
177	176	179	178	176	178	178	176	173	177	179	179	178	177	176	175
176	177	177	177	176	178	179	175	174	176	179	177	175	176	178	179
175	176	176	175	174	176	176	175	173	175	176	176	174	175	178	178
175	175	175	175	175	176	176	175	175	175	175	175	175	175	176	175

174	174	176	178	177	175	174	177	177	176	174	174	174	175	175	174
174	174	176	178	178	175	174	177	178	176	174	174	174	175	175	174
175	175	175	176	176	176	174	175	176	176	175	176	177	176	174	174
176	176	175	174	175	176	174	174	175	175	175	176	179	177	175	175
178	176	175	174	175	175	176	177	175	175	174	176	177	177	174	175
178	176	176	176	176	176	178	179	175	174	173	174	175	176	175	174
176	176	176	179	178	177	177	177	175	174	173	174	175	177	176	175
174	174	176	178	177	177	175	175	175	176	174	174	176	178	178	175
174	175	176	175	174	175	176	175	177	178	176	174	175	178	178	176
175	175	175	174	174	175	175	175	177	180	177	176	176	175	176	174
176	175	174	176	177	179	179	175	176	178	178	179	178	175	174	171

Setelah diperoleh nilai grayscale, selanjutnya melakukan perhitungan dengan menggunakan metode FELICS, yaitu:

- a. Buat tabel kemungkinan frekuensi kemunculan nilai piksel.

**Tabel 2.** Frekuensi Kemunculan Nilai Piksel

NO	Piksel L s/d H	Frekuensi	Nilai Biner	Jlh Biner	Jumlah Bit * Frek
1.	171	1	10101011	8	8
2.	173	4	10101101	8	32
3.	174	43	10101110	8	344
4.	175	77	10101111	8	616
5.	176	63	10110000	8	504
6.	177	27	10110001	8	216
7.	178	26	10110010	8	208
8.	179	14	10110011	8	112
9.	180	1	10110100	8	8
Jumlah Total Bit * Frekuensi					2048

Frekuensi kemunculan nilai piksel pada tabel 2 mempunyai nilai total jumlah bit dikalikan frekuensi kemunculan nilai piksel senilai 2048 bit.

- b. Buat Tabel Kompresi FELICS

Membuat tabel kompresi FELICS mengacu pada keputusan tata letak 3 piksel dengan ketentuan sebagai berikut:

**Tabel 3.** Penentuan Posisi Piksel yang akan di *Encoding*

Posisi Nilai Piksel	Biner Code Awal (Biner +)	Encode
$L \leq P \leq H$	00	$N = P - L$
$P < L$	10	$N = L - P - 1$
$P > H$	11	$N = P - H - 1$

Selanjutnya tentukan nilai N1 dan N2 sebagai nilai piksel terdekat dengan nilai piksel (P), kemudian tentukan nilai Tinggi (H) dan Rendah (L) dari "N1; N2" telah ditentukan. Berikut tabel codewordnya.

**Tabel 4.** Hasil Kompresi Metode *FELICS*

No	Piksel (P)	Frekuensi	N1 ; N2	H	L	Posisi Piksel	Rumus Encode	Hasil (Des)
1	171	1	174 ; 174	174	174	P<L	$N = L - P - 1$	2
2	173	4	176 ; 175	176	175	P<L	$N = L - P - 1$	1
3	174	43	176 ; 175	176	175	P<L	$N = L - P - 1$	0

4	175	77	174 ; 175	174	175	P>H	$N = P - H - 1$	0
5	176	63	175 ; 174	175	174	P>H	$N = P - H - 1$	0
6	177	27	178 ; 175	178	175	$L \leq P \leq H$	$N = P - L$	2
7	178	26	176 ; 178	176	178	$L \leq P \leq H$	$N = P - L$	0
8	179	14	178 ; 176	178	176	P>H	$N = P - H - 1$	0
9	180	1	177 ; 178	177	178	P>H	$N = P - H - 1$	2

**Tabel 3.** Penentuan Posisi Piksel yang akan di *Encoding*

Posisi Nilai Piksel	Biner Code Awal	
	(Biner +)	Encode
$L \leq P \leq H$	00	$N = P - L$
$P < L$	10	$N = L - P - 1$
$P > H$	11	$N = P - H - 1$

Selanjutnya menentukan nilai N1 dan N2 sebagai nilai piksel terdekat terhadap nilai piksel (P), kemudian menentukan nilai *High* (H) dan *Low* (L) dari nilai "N1; N2" yang telah ditentukan. Berikut tabel *codeword*.

**Tabel 4.** Hasil Kompresi Metode *FELICS*

No	Piksel (P)	Frekuensi	N1 ; N2	H	L	Posisi Piksel	Rumus Encode	Hasil (Des)
1	171	1	174 ; 174	174	174	P<L	$N = L - P - 1$	2
2	173	4	176 ; 175	176	175	P<L	$N = L - P - 1$	1
3	174	43	176 ; 175	176	175	P<L	$N = L - P - 1$	0
4	175	77	174 ; 175	174	175	P>H	$N = P - H - 1$	0
5	176	63	175 ; 174	175	174	P>H	$N = P - H - 1$	0
6	177	27	178 ; 175	178	175	$L \leq P \leq H$	$N = P - L$	2
7	178	26	176 ; 178	176	178	$L \leq P \leq H$	$N = P - L$	0
8	179	14	178 ; 176	178	176	P>H	$N = P - H - 1$	0
9	180	1	177 ; 178	177	178	P>H	$N = P - H - 1$	2

Berikut tabel lanjutan proses kompresi serta hasil penerapan metode *FELICS*, ditampilkan pada tabel 5.

**Tabel 5.** Hasil Kompresi Dalam Bentuk Biner

No	Piksel	Frekuensi	Hasil (Des)	Biner (B)	Biner+ (B+)	Biner+ : Biner	Jlh Biner	Frekuensi * Jlh Bit
1	171	1	2	10	10	1010	4	4
2	173	4	1	01	10	1001	4	16
3	174	43	0	00	10	1000	4	172
4	175	77	0	00	11	1100	4	308
5	176	63	0	00	11	1100	4	252
6	177	27	2	10	00	0010	4	108
7	178	26	0	00	00	0000	4	104
8	179	14	0	00	11	1100	4	56
9	180	1	0	00	11	1100	4	4
Total Jumlah Bit								1024

Dari hasil penerapan metode *FELICS* di atas memiliki total jumlah bit senilai 1024 bit.

### c. Menghitung Kinerja Kompresi

Ukuran citra .png sebelum dikompresi adalah 2048 piksel. Setelah dilakukan proses pemampatan menggunakan metode *FELICS* maka dihasilkan nilai piksel sebesar 1024. Analisa rasio kompresinya dapat dihitung sebagai berikut:

1. *Ratio of Compression (Rc)*, merupakan nilai perbandingan antara ukuran bit data sebelum di kompresi dengan ukuran bit data yang telah dikompresi.

$$Rc = \frac{\text{Ukuran File Asli}}{\text{Ukuran File Terkompresi}}$$

$$\begin{aligned} &= \frac{2048}{1024} \\ &= 2 \end{aligned}$$

2. *Compression Ratio (CR)*, merupakan persentase perbandingan antara data yang sudah dikompresi dengan data yang belum dikompresi.

$$\begin{aligned} CR &= \frac{\text{Ukuran File Terkompresi}}{\text{Ukuran Asli}} \times 100\% \\ &= \frac{1024}{2048} \times 100\% \\ &= 0,5 \times 100\% \\ &= 50\% \end{aligned}$$

3. *Space Saving (SS)*, adalah jumlah penyimpanan yang dihemat setelah dilakukan kompresi, persentase antara data yang tidak terkompresi dengan data terkompresi

$$\begin{aligned} SS &= 100\% - CR \\ &= 100\% - 50\% \\ &= 50\% \end{aligned}$$

#### 4. KESIMPULAN

Berdasarkan pengujian kompresi data menggunakan FELICS terhadap citra PNG yang telah dilakukan, maka dapat disimpulkan bahwa prosedur kompresi pada citra format .png menggunakan metode *Fast Efficient & Lossless Image Compression System (FELICS)* telah berhasil dilakukan dan proses kompresi berjalan sesuai dengan dengan teknik kompresi. Berdasarkan hasil uji coba aplikasi yang telah dibangun terhadap lima citra dengan format .png yang berukuran 5.249.494 *byte* berhasil dikompresi menjadi 2.041.208 *byte* dengan CR 28,00%, citra kedua 3.124.201 *byte* menjadi 1.060.781 *byte* dengan CR 33,95%, citra ketiga 10.473.459 *byte* menjadi 3.065.010 *byte* dengan CR 29,26%, citra keempat 804.728 *byte* menjadi 200.215 *byte* dengan CR 24,87% dan citra kelima 503.111 *byte* menjadi 158.711 *byte* dengan CR 31,55%.

#### REFERENCES

- [1] S. Ratna, "Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm," *Technol. J. Ilm.*, vol. 11, no. 3, p. 181, 2020, doi: 10.31602/tji.v11i3.3294.
- [2] E. Prayoga and K. M. Suryaningrum, "Implementasi Algoritma Huffman Dan Run Length Encoding Pada Aplikasi Kompresi Berbasis Web," *J. Ilm. Teknol. Infomasi Terap.*, vol. 4, no. 2, pp. 92–101, 2018, doi: 10.33197/jitter.vol4.iss2.2018.154.
- [3] H. Wijayanto, I. A. Prabowo, and P. Harsadi, "Optimalisasi Penyusutan Exif Metadata Dengan Teknik Substitusi Null Value Pada Kasus Keamanan Citra Digital," *J. Ilm. SINUS*, vol. 16, no. 1, p. 1, 2018, doi: 10.30646/sinus.v16i1.327.
- [4] S. F. Florentin, M. A. Rony, and S. Kom, "Penerapan Kompresi Citra Digital Dengan Metode Kuantisasi Warna Menggunakan Algoritma K-Means Pada Website Peduli Sekitar," *Skanika*, vol. 1, no. 2, pp. 431–438, 2018, [Online]. Available: <https://jom.fti.budiluhur.ac.id/index.php/SKANIKA/article/view/237>
- [5] Preddy Marpaung, *Peningkatan Kualitas Citra Menggunakan Algoritma Spatial Median Filter Dan Adaptive Fuzzy Contrast Enhancement*. Pasuruan: CV. Penerbit Qiara Media, 2020.
- [6] N. Wijaya and A. Ridwan, "Klasifikasi Jenis Buah Apel Dengan," *Sisfokom*, vol. 08, no. 1, pp. 74–78, 2019.
- [7] G. Hasibuan, "Analisa Kombinasi Algoritma Burrows Wheeler Transform dan Adaptive Huffman Coding untuk Kompresi Citra," *Bull. Multi-Disciplinary Sci. Appl. Technol.*, vol. 1, no. 2, pp. 34–40, 2022.
- [8] R. D. B. Sitepu, R. Fadhil, and I. S. Nasution, "Klasifikasi Karakteristik Fisik Biji Pinang Belah Kering (Areca catechu) Menggunakan Pengolahan Citra Digital," *J. Ilm. Mhs. Pertan.*, vol. 7, no. 2, pp. 505–513, 2022, doi: 10.17969/jimfp.v7i2.19860.
- [9] R. Panjaitan, M. Andriana, T. Sumarlin, S. Rakasiswi, and Y. Fitrianto, "Literasi Metode Penelitian Dan Bisnis Teknologi Kreatif Di Era New Normal," *J. Komunitas J. Pengabd. Kpd. Masy.*, vol. 3, no. 2, pp. 109–116, 2021, doi: 10.31334/jks.v3i2.1272.
- [10] G. H. Irawan and N. Ramdhani, "Steganografi Foto dalam Citra dengan Metode LSB dan Kompresi Metode Huffman," 2019.

- 
- [11] Eni, “Seminar Nasional Multidisiplin Ilmu ‘Inovasi Teknologi dan Produk Penelitian Pengabdian Masyarakat Berbasis Revolusi Industri 4.0 di Era New Normal,’” *Angew. Chemie Int. Ed.* 6(11), 951–952., no. Mi, pp. 5–24, 2021.
- [12] M. Zalukhu, “Analisis Efisiensi Dan Efektivitas Metode FELICS Pada Proses Kompresi Citra JPEG,” *KOMIK (Konferensi Nas. Teknol. ...)*, vol. 4, pp. 99–108, 2020, doi: 10.30865/komik.v4i1.2648.
- [13] M. Purnasari and Y. Hartiwi, “Juli 2022 Hal 258-264 Fakultas Ilmu Komputer, Sistem informasi,” *Media Online*, vol. 2, no. 6, pp. 258–264, 2022, [Online]. Available: <https://djournals.com/resolusi>
- [14] A. Voutama, “Sistem Antrian Cucian Mobil Berbasis Website Menggunakan Konsep CRM dan Penerapan UML,” *Komputika J. Sist. Komput.*, vol. 11, no. 1, pp. 102–111, 2022, doi: 10.34010/komputika.v11i1.4677.
- [15] E. Affandi and T. Syahputra, “Pemodelan UML Manajemen Sistem Inventory,” *J. Teknol. Sist. Inf. dan Sist. Komput. TGD*, vol. 1, no. 2, pp. 14–25, 2018.
- [16] A. Mubarak, “Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 2, no. 1, pp. 19–25, 2019, doi: 10.33387/jiko.v2i1.1052.
- [17] N. I. Yusman, “Perancangan Sistem Informasi Berbasis Orientasi Objek Menggunakan Star Uml Di Cv Niasa Bandung,” *J. Account. Inf. Syst.*, vol. 1, no. 2, pp. 101–109, 2018, doi: 10.32627/aims.v1i2.358.
- [18] A. F. Prasetya, Sintia, and U. L. D. Putri, “Perancangan Aplikasi Rental Mobil Menggunakan Diagram UML (Unified Modelling Language),” *J. Ilm. Komput. Terap. dan Inf.*, vol. 1, no. 1, pp. 14–18, 2022.
- [19] N. SALSABILLA, “Peranan Perangkat Keras ( Hardware ) Dalam Sis- Tem Informasi Manajemen,” *Inf. Manaj.*, no. 0702212214, 2022.