

# KOMPRESI CITRA BERBASIS *MULTI-LEVEL THRESHOLDING* MENGGUNAKAN *DIFFERENTIAL EVOLUTION*

I Gusti Lanang Arda Narendra<sup>1</sup>, I Gusti Ngurah Putu Krisna Adi<sup>1</sup>, Yobel Christanto Nadeak<sup>1</sup>, I Made Oka Widyantara<sup>2</sup>, Dewa Made Wiharta<sup>2</sup>

<sup>1</sup>Mahasiswa Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana

<sup>2</sup>Dosen Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana  
Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana, Jalan Raya Kampus Unud Jimbaran, Bali

[ardanarendr@gmail.com](mailto:ardanarendr@gmail.com), [ngurah.krisna5@gmail.com](mailto:ngurah.krisna5@gmail.com), [yobelnadeak13@gmail.com](mailto:yobelnadeak13@gmail.com),  
[oka.widyantara@unud.ac.id](mailto:oka.widyantara@unud.ac.id), [wiharta@unud.ac.id](mailto:wiharta@unud.ac.id)

## ABSTRAK

Kompresi pada data citra perlu dilakukan untuk mengelola ruang penyimpanan agar dapat digunakan secara efisien. Pada penelitian ini kompresi citra dilakukan menggunakan algoritma *Differential Evolution* (DE) sebagai pendekatan untuk mencapai ambang batas yang optimal dalam mengelompokkan piksel pada citra. Unjuk kerja metode kompresi citra ini dievaluasi berdasarkan parameter *Peak Signal-to-Noise Ratio* (PSNR) dan *Structural Similarity Index Measure* (SSIM). PSNR kompresi citra DE adalah 27 dB hingga 40 dB dengan level *threshold* antara 10 sampai dengan 40. Sedangkan SSIM bernilai 0,83 hingga 0,97 pada level *threshold* 10 sampai dengan 40. PSNR dan SSIM pada kompresi citra DE bernilai baik serta menghasilkan visual citra terkompresi yang mendekati citra aslinya seiring dengan kenaikan level *threshold* yang ditetapkan.

**Kata kunci** : Kompresi Citra, *Thresholding*, *Differential Evolution*, PSNR, SSIM.

## ABSTRACT

*Compression in image data needs to be done to manage storage space efficiently. In this study, image compression is performed using the Differential Evolution (DE) algorithm as an approach to achieve an optimal threshold for pixel clustering in images. The performance of this image compression method is evaluated based on the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) parameters. The PSNR of DE image compression ranges from 27 dB to 40 dB with threshold levels between 10 to 40. Meanwhile, the SSIM ranges from 0.83 to 0.97 at threshold levels of 10 to 40. The PSNR and SSIM of DE image compression are both at a good level, resulting in compressed image visuals that closely resemble the original image as the threshold levels increase.*

**Key Words** : Image compression, *Thresholding*, *Differential Evolution*, PSNR, SSIM.

## 1. PENDAHULUAN

Kebutuhan media penyimpanan di era digital semakin meningkat akibat kemajuan teknologi yang memungkinkan pengguna untuk mengakses dan mengunduh data dengan ukuran *file* yang besar. Seperti pada sistem presensi wajah di perkantoran, penggunaan wajah sebagai validasi dan verifikasi presensi memerlukan ruang lebih besar untuk menyimpan data citra wajah tiap pegawai dalam kurun waktu tertentu. Data citra digital yang tidak dikelola dengan baik

menyebabkan ruang penyimpanan menjadi penuh dan tidak dapat digunakan secara efisien. Hal tersebut dapat diatasi dengan melakukan kompresi terhadap data citra. Kompresi citra dilakukan untuk mengurangi ukuran data citra sehingga mengefisienkan ruang penyimpanan dan waktu transfer menjadi lebih singkat [1].

Kompresi citra terdiri dari dua jenis, yaitu kompresi *lossless* dan kompresi *lossy*. Kompresi *lossless* menghasilkan citra hasil kompresi yang serupa dengan citra aslinya.

Sedangkan kompresi *lossy* menghasilkan citra terkompresi yang kurang serupa dengan citra aslinya karena sebagian informasinya dihapus ketika proses kompres-dekompresi [2]. Teknik kompresi citra yang selama ini digunakan umumnya berbasis spasial pada logika pengkodean berdasarkan pemrosesan citra dengan membaginya menjadi blok-blok yang lebih kecil [3].

Seiring dengan kemajuan teknologi di bidang pengolahan citra, beberapa metode kompresi citra telah dikembangkan dan dikombinasikan untuk mencapai hasil terbaik. Penelitian [3], yaitu penelitian mengenai kompresi citra berbasis histogram yang menggunakan metode *multi-level thresholding* [3]. Metode *thresholding* umumnya digunakan pada segmentasi citra yang merupakan proses penentuan ambang batas nilai piksel dengan membagi citra menjadi beberapa bagian berdasarkan tingkat intensitas skala keabuan [4]. Ketika nilai *threshold* meningkat, perkiraan frekuensi penyebaran nilai intensitas piksel pada citra menjadi lebih akurat dan hampir sesuai dengan nilai sebenarnya. Pada saat yang sama, variasi nilai piksel semakin berkurang dan dengan ini kompresi citra dapat dicapai [3].

Teknik optimasi seperti *Differential Evolution* merupakan algoritma komputasi evolusioner berbasis populasi yang meningkatkan kandidat solusi secara iteratif berdasarkan proses evolusi. Dalam hal ini, *Differential Evolution* digunakan sebagai pendekatan untuk mencari ambang batas yang optimal dalam mengelompokkan piksel citra pada metode *multi-level thresholding*.

Penelitian ini mengusulkan metode *multi-level thresholding* menggunakan algoritma *Differential Evolution* sebagai pendekatan dalam melakukan kompresi citra. Hasil kompresi citra kemudian dianalisis berdasarkan parameter *Peak Signal-to-Noise Ratio* (PSNR) dan *Structural Similarity Index Measure* (SSIM)

untuk mengevaluasi kinerja dari algoritma serta kualitas citra terkompresi.

## 2. LANDASAN TEORI

### 2.1 Thresholding

Metode *thresholding* digunakan untuk melakukan segmentasi pada citra dengan tujuan membedakan antara objek dengan *background* dalam citra berdasarkan perbedaan intensitas atau tingkat kecerahan pada citra tersebut. *Thresholding* bekerja dengan cara menentukan nilai *threshold* (*th*) atau nilai ambang batas sehingga piksel yang nilai intensitasnya lebih tinggi dari (*th*) diberi label sebagai satu kelas [5]. *Thresholding* dibedakan menjadi *bi-level thresholding* dan *multi-level thresholding*. *Bi-level thresholding* membagi citra menjadi dua kelas hanya berdasarkan satu nilai *threshold*, sedangkan pada *multi-level thresholding* memerlukan lebih dari dua nilai *threshold* dan membagi piksel citra menjadi beberapa kelas [4].

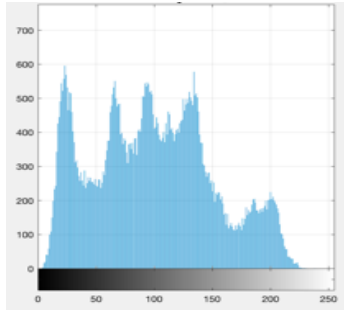
### 2.2 Histogram Citra

Histogram adalah representasi grafis dari persebaran nilai-nilai intensitas piksel dari suatu citra [6]. Histogram citra dapat menunjukkan frekuensi kemunculan dari intensitas, kecerahan, dan kontras dari sebuah citra. Secara matematis, histogram citra dapat dihitung menggunakan persamaan (1) [7].

$$p_r(r_k) = \frac{n_k}{n}; 0 \leq r_k \leq 1; k = 0, 1, \dots, L-1 \quad (1)$$

dengan  $L$  merupakan jumlah level,  $p_r(r_k)$  merupakan probabilitas dari kemunculan level ke- $k$ ,  $n_k$  merupakan jumlah kemunculan level  $k$  pada citra, dan  $n$  adalah total jumlah piksel pada citra [7].

Gambar 1 adalah contoh dari histogram citra, dengan sumbu-x pada histogram tersebut merepresentasikan rentang dari intensitas yang dimiliki oleh piksel pada citra, sedangkan sumbu-y mewakili jumlah piksel pada citra dengan nilai yang sama.



Gambar 1. Contoh histogram citra

### 2.3 Differential Evolution (DE)

*Differential Evolution* merupakan algoritma pengoptimalan berdasarkan populasi. Pada algoritma *DE*, vektor parameter atau individu ( $i$ ) dari populasi pada generasi waktu ( $t$ ) adalah vektor dimensi- $D$  yang memuat sekumpulan parameter pengoptimalan  $D$ . Adapun persamaannya dirumuskan dengan persamaan (2) [8].

$$\vec{Z}_i(t) = [Z_{i,1}(t), Z_{i,2}(t), \dots, Z_{i,D}(t)] \quad (2)$$

Anggota populasi diubah dari  $\vec{Z}_i(t)$  menjadi vektor pembagi  $\vec{Y}_i(t)$  pada setiap generasi. Pembuatan vektor pembagi inilah yang membedakan masing-masing skema dari *DE*. Salah satu varian *DE* dinamakan skema *DE/rand/1* yang menghasilkan vektor pembagi pada setiap individu ( $i$ ) dengan parameter lain (misalkan vektor  $r_1, r_2, r_3$  sehingga  $r_1, r_2, r_3 \in [1, NP]$  dan  $r_1 \neq r_2 \neq r_3$ ) dipilih *random* dari populasi saat ini. Vektor pembagi  $\vec{Y}_i(t)$  kemudian didapat dari mengalikan perbedaan antara dua dari tiga bilangan dengan bilangan skalar  $F$ . Komponen  $j^{th}$  dan vektor  $i^{th}$  dapat dinyatakan pada persamaan (3) [8].

$$\vec{Y}_{i,j}(t) = Z_{r_1,j}(t) + F Z_{r_2,j}(t) - Z_{r_3,j}(t) \quad (3)$$

Proses persilangan binomial atau *crossover* bertujuan untuk mengoptimalkan keragaman populasi. *Crossover* binomial dilakukan di tiap variabel  $D$ . Angka yang dipilih secara *random* berkisar 0 dan 1 yang berada pada nilai  $C_r$ . Setiap vektor target  $\vec{Z}_i(t)$  dan vektor percobaan  $\vec{R}_i(t)$  dirumuskan dengan persamaan (4) [8].

$$R_{i,j}(t) = Y_{i,j}(t) \text{ jika } r \text{ dan } j (0,1) \leq C_r \text{ atau } j=rn(i) \quad (4)$$

$= Z_{i,1}(t) \text{ adalah sebaliknya}$

Dengan  $j = 1, 2, \dots, D$  dan  $rand_j(0,1) \in [0,1]$  merupakan evaluasi *random* namun seragam ke- $j$ ,  $rn(i) \in [1, 2, \dots, D]$  merupakan indeks yang dipilih secara *random* untuk meyakinkan bahwa  $\vec{R}_i(t)$  memperoleh minimal satu komponen dari  $\vec{Z}_i(t)$ . Proses selanjutnya yaitu melakukan proses pemilihan akhir untuk menentukan mana yang akan tetap berada pada generasi selanjutnya. Apabila vektor percobaan memperoleh nilai *fitness function* atau nilai kesesuaian yang lebih baik, maka akan menggeser vektor targetnya pada generasi selanjutnya. Jika nilai *fitness function* yang diperoleh tidak lebih baik, maka vektor induk pada generasi sebelumnya akan tetap berada dalam populasi yang dirumuskan pada persamaan (5) [8].

$$\vec{Z}_i(t+1) = \begin{cases} \vec{R}_i(t) & \text{jika } f(\vec{R}_i(t)) < f(\vec{Z}_i(t)) \\ \vec{Z}_i(t) & \text{jika } f(\vec{R}_i(t)) \geq f(\vec{Z}_i(t)) \end{cases} \quad (5)$$

dengan  $f$  adalah fungsi yang harus diminimalkan. Tahap di atas dilanjutkan sampai kriteria penghentian atau jumlah iterasi terpenuhi.

### 2.4 Peak Signal-to-Noise Ratio (PSNR)

*Peak Signal-to-Noise Ratio* (PSNR) merupakan perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau (*noise*) yang berpengaruh pada sinyal tersebut [2]. Adapun rumus untuk menghitung PSNR ditunjukkan pada persamaan (6) [2].

$$PSNR = 20 \log_{10} \frac{Max_i[P_i]}{\sigma^2} \quad (6)$$

dengan  $Max_i[P_i]$  adalah nilai maksimum yang diambil dari elemen atau piksel suatu citra, dan  $\sigma^2$  adalah perbedaan kesalahan kuadrat rerata antara nilai piksel asli dan nilai piksel yang dikompresi [9].

Umumnya, nilai PSNR untuk kompresi citra dan video *lossy* berada pada rentang 30 dB hingga 50 dB dan jika lebih tinggi maka akan lebih baik. Nilai di atas 40 dB

dianggap sangat baik dan di bawah 20 dB biasanya tidak dapat diterima [10].

**2.5 Structure Similarity Index Measure (SSIM)**

Parameter *Structure Similarity Index Measure* (SSIM) adalah indeks penilaian kualitas citra untuk mengukur kesamaan struktural dari citra asli dengan citra hasil kompresi. Rentang nilai SSIM adalah 0 sampai dengan 1 [11]. Nilai SSIM yang mendekati 1 menandakan citra yang dibandingkan memiliki keselarasan dan memiliki kualitas citra yang baik, begitupun untuk sebaliknya. Adapun rumus untuk menghitung SSIM ditunjukkan pada persamaan (7) [7].

$$SSIM(x, y) = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (7)$$

dengan  $\mu_x$  dan  $\mu_y$  adalah intensitas rata-rata citra *input* dan *output*,  $\sigma_x^2$  dan  $\sigma_y^2$  adalah standar deviasi,  $\sigma_{xy}$  didefinisikan sebagai kovarian, sedangkan  $c_1$  dan  $c_2$  adalah konstanta.

**3. METODOLOGI PENELITIAN**

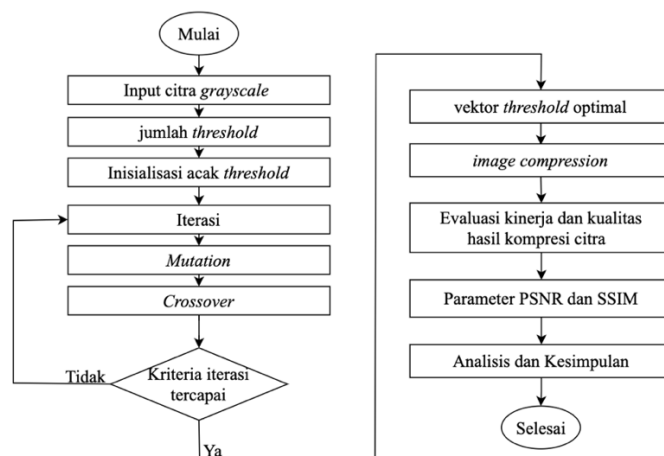
Penelitian ini menggunakan kerangka kerja kompresi citra berbasis metode *multi-level thresholding* menggunakan algoritma *Differential Evolution*. Proses kompresi citra dilakukan menggunakan perangkat lunak Matlab. Adapun parameter yang digunakan pada algoritma *Differential Evolution* ditampilkan pada Tabel 1.

Tabel 1. Parameter Algoritma DE

Parameter	Nilai
Faktor Pembobotan (F)	0,5
Probabilitas Persilangan (Cr)	0,9
Populasi DE (NP)	10
Maksimum generasi	100
Batas minimum - maksimum	0-255

Gambar 2 adalah diagram alir tahapan kompresi citra berbasis *multi-level thresholding* dengan algoritma *Differential Evolution* dijelaskan sebagai berikut.

1. Masukkan citra wajah *grayscale* dan nilai *threshold* pada program Matlab.
2. Lakukan inialisasi parameter pada algoritma *Differential Evolution*.
3. Lakukan proses *mutation* dan *crossover* pada algoritma *Differential Evolution* untuk menghasilkan kandidat solusi terbaik dalam populasi hingga mencapai kriteria iterasi yang ditentukan.
4. Lakukan proses kompresi citra menggunakan algoritma *Differential Evolution* dengan vektor *threshold image* berdasarkan histogram citra.
5. Lakukan evaluasi dan analisis terhadap hasil dari kualitas citra terkompresi beserta kinerja algoritma kompresi citra dengan parameter PSNR dan SSIM.
6. Kesimpulan dibuat berdasarkan hasil evaluasi kinerja algoritma *Differential Evolution* pada kompresi citra berbasis *multi-level thresholding*.

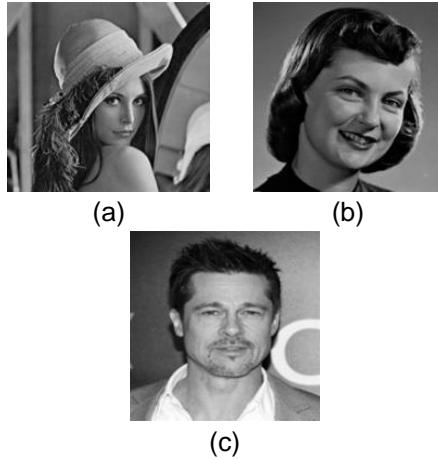


Gambar 2. Diagram alir penelitian

#### 4. HASIL DAN PEMBAHASAN

##### 4.1 Citra Input

Citra yang digunakan sebagai input pada penelitian ini merupakan citra wajah *greyscale* dengan ukuran 8 bit per piksel (256 x 256). Citra tersebut dikompresi dengan empat level *threshold* yang berbeda yaitu level 10, 20, 30, dan 40. Gambar 3 adalah citra input yang digunakan.



Gambar 3. (a) Citra wajah 1, (b) Citra wajah 2, (c) Citra wajah 3.

##### 4.2 Evaluasi Kompresi Citra

Hasil citra terkompresi kemudian dievaluasi menggunakan parameter PSNR dan SSIM untuk meninjau kualitas visual dan kemiripan citra terkompresi dengan citra aslinya. Hasil dari kompresi citra berbasis metode *multi-level thresholding* menggunakan algoritma *Differential Evolution* ditunjukkan pada Tabel 2.

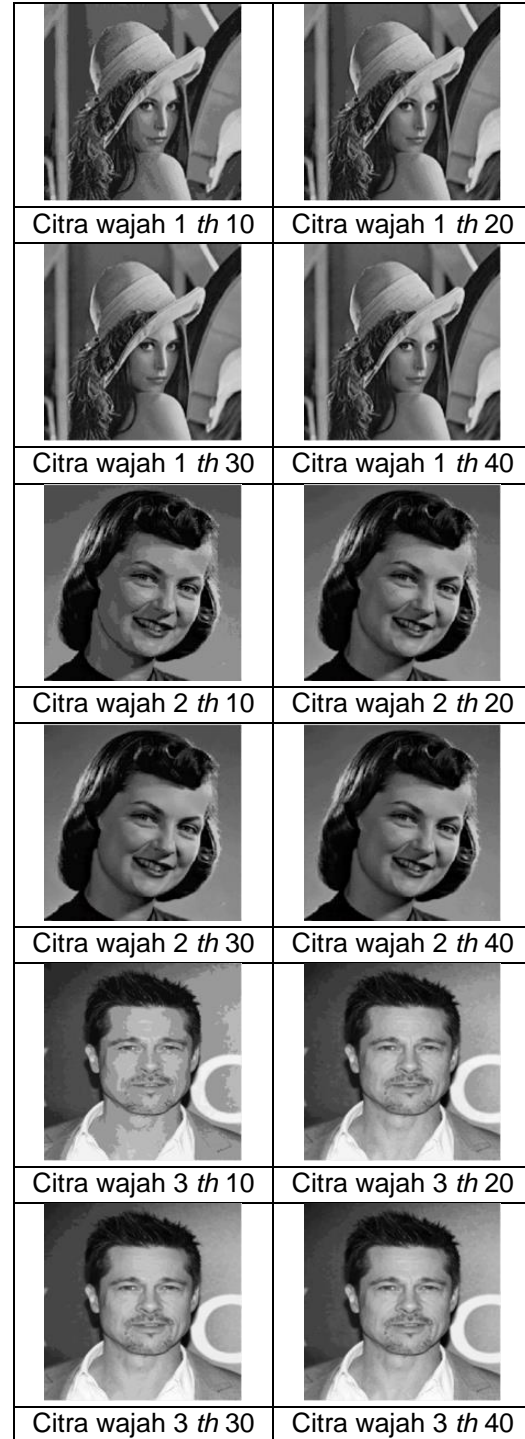
Tabel 2. Hasil Kompresi Citra DE

Citra Input	Level <i>th</i>	PSNR	SSIM
Citra Wajah 1	10	27,9013	0,8380
	20	34,4413	0,9332
	30	38,4006	0,9666
	40	39,9306	0,9763
Citra Wajah 2	10	29,2085	0,8635
	20	34,8871	0,9343
	30	38,3753	0,9589
	40	40,7358	0,9744
Citra Wajah 3	10	29,5993	0,8420
	20	35,7298	0,9281
	30	37,3026	0,9448
	40	38,2655	0,9591

##### 4.3 Evaluasi Hasil Citra Terkompresi

Hasil citra terkompresi dari kompresi citra menggunakan algoritma *Differential Evolution* dengan input citra wajah ditunjukkan pada Tabel 3.

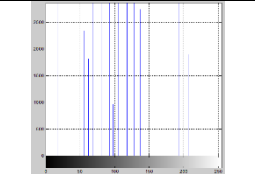
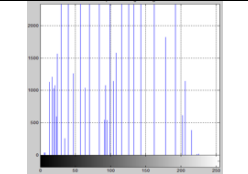
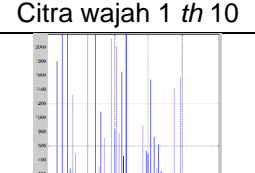
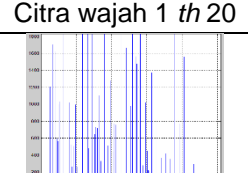

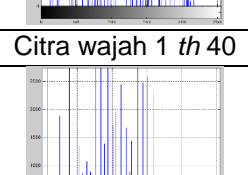
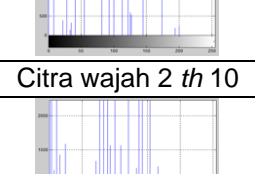
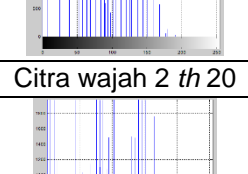


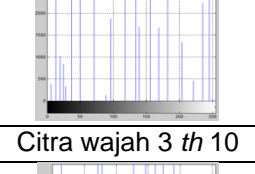
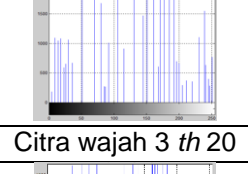
Tabel 3. Hasil Citra Terkompresi



#### 4.4 Evaluasi Histogram

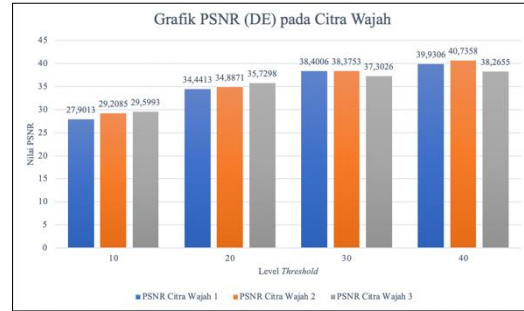
Histogram hasil kompresi citra menggunakan algoritma *Differential Evolution* dengan input citra wajah ditunjukkan pada Tabel 4.

Tabel 4. Histogram Citra Terkompresi

	
Citra wajah 1 th 10	Citra wajah 1 th 20
	
Citra wajah 1 th 30	Citra wajah 1 th 40
	
Citra wajah 2 th 10	Citra wajah 2 th 20
	
Citra wajah 2 th 30	Citra wajah 2 th 40
	
Citra wajah 3 th 10	Citra wajah 3 th 20
	
Citra wajah 3 th 30	Citra wajah 3 th 40

#### 4.5 Analisis PSNR

Grafik PSNR berdasarkan hasil penelitian pada Tabel 2 dari ketiga input citra wajah yang digunakan pada kompresi citra DE ditunjukkan pada Gambar 4.



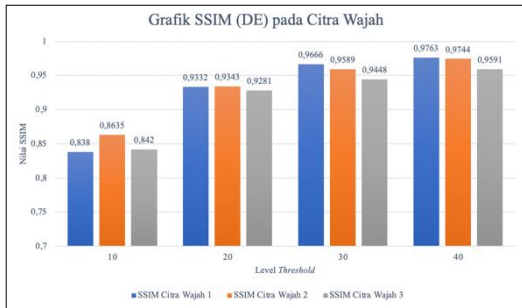
Gambar 4. Grafik PSNR DE

Berdasarkan grafik PSNR pada Gambar 4, nilai PSNR di setiap citra mengalami peningkatan ketika level *threshold* yang ditetapkan semakin tinggi. Citra wajah 1 menunjukkan bahwa nilai PSNR yang diperoleh meningkat dari 27,90 dB pada level *threshold* 10 mencapai 39,93 dB pada level *threshold* 40. Nilai PSNR pada Citra wajah 2 meningkat dari 29,20 dB pada level *threshold* 10 dan mencapai 40,73 dB pada level *threshold* 40. Nilai PSNR pada Citra wajah 3 adalah 29,59 dB pada level *threshold* 10 dan meningkat menjadi 38,26 dB pada level *threshold* 40. Hasil ini menunjukkan bahwa nilai PSNR meningkat seiring dengan kenaikan level *threshold* yang ditetapkan. Begitupun sebaliknya, jika level *threshold* yang digunakan semakin rendah, nilai PSNR akan rendah.

Nilai PSNR untuk kompresi citra *lossy* menunjukkan hasil yang baik apabila memiliki rentang nilai berkisar antara 30 dB hingga 50 dB. Nilai PSNR di atas 40 dB dianggap sangat baik dan di bawah 20 dB biasanya tidak dapat diterima [10]. Nilai PSNR yang diperoleh dari kompresi citra dengan inputan citra wajah berkisar dari 27 dB hingga 40 dB, bergantung pada level *threshold* yang digunakan. Merujuk pada standar PSNR tersebut, hasil kompresi citra menggunakan algoritma *Differential Evolution* dengan inputan citra wajah dinilai telah memperoleh nilai PSNR yang baik.

#### 4.6 Analisis SSIM

Grafik SSIM berdasarkan hasil penelitian pada Tabel 2 dari ketiga input citra wajah yang digunakan pada kompresi citra DE ditunjukkan pada Gambar 5.



Gambar 5. Grafik SSIM DE

Grafik yang ditampilkan pada Gambar 5 menunjukkan bahwa ketiga citra wajah yang diujikan mengalami peningkatan nilai SSIM yang signifikan terhadap kenaikan level *threshold* yang ditetapkan. Pada citra wajah 1, nilai SSIM yang diperoleh meningkat dari 0,83 pada level *threshold* 10 mencapai 0,97 pada level *threshold* 40. Nilai SSIM pada Citra wajah 2 adalah 0,86 pada level *threshold* 10 dan meningkat menjadi 0,97 pada level *threshold* 40. Nilai SSIM pada Citra wajah 3 adalah 0,84 pada level *threshold* 10 dan meningkat sampai 0,95 pada level *threshold* 40. Hasil kompresi citra dengan level *threshold* yang rendah menghasilkan nilai SSIM yang lebih rendah. Sedangkan pada level *threshold* yang lebih tinggi, kompresi citra menghasilkan nilai SSIM yang lebih tinggi hingga mendekati nilai 1.

Merujuk pada teori, nilai SSIM yang mendekati nilai 1 menandakan citra yang dibandingkan memiliki korelasi atau kemiripan struktur dan kualitas citra yang baik. Sehingga dapat dikatakan bahwa hasil kompresi citra menggunakan algoritma *Differential Evolution* dengan inputan citra wajah memiliki korelasi atau kemiripan struktur dan kualitas citra yang baik seiring dengan peningkatan level *threshold* yang digunakan. Sebaliknya, ketika level *threshold* yang digunakan lebih rendah, hasil kompresinya kurang memiliki korelasi atau kemiripan struktur seperti citra aslinya.

#### 4.7 Analisis Citra Terkompresi

Berdasarkan hasil kompresi citra pada Tabel 3, citra terkompresi yang dihasilkan menunjukkan visual yang semakin baik seiring dengan peningkatan level *threshold* yang ditetapkan. Hasil citra yang dikompresi dengan *threshold* level 10 memiliki kualitas

visual yang kurang baik jika dibandingkan dengan hasil citra yang dikompresi dengan *threshold* level 20, 30, dan 40. Hal ini juga sejalan dengan nilai PSNR dan SSIM yang diperoleh. Citra yang dikompresi dengan *threshold* level 10 memiliki nilai PSNR dan SSIM yang lebih rendah dibandingkan dengan citra yang dikompresi dengan *threshold* level 20, 30, dan 40. Dari hasil tersebut dapat dikatakan bahwa citra terkompresi menunjukkan hasil yang semakin baik atau mendekati dengan citra inputannya ketika level *threshold* yang digunakan meningkat.

#### 4.8 Analisis Histogram

Berdasarkan hasil histogram citra terkompresi yang ditunjukkan pada Tabel 4, citra yang dikompresi dengan level *threshold* 10 memiliki jumlah persebaran intensitas piksel yang lebih sedikit. Hal ini dikarenakan piksel di bawah nilai ambang batas yang ditentukan diubah menjadi nilai bit 0 (hitam), sedangkan piksel di atas nilai ambang batas diubah menjadi nilai bit 1 (putih). Oleh sebab itu, persebaran intensitas piksel pada level *threshold* 10 menjadi lebih sedikit dan tidak dapat merepresentasikan citra aslinya dengan maksimal. Ketika citra dikompresi dengan level *threshold* 40 atau level *threshold* yang ditetapkan meningkat, frekuensi persebaran nilai intensitas piksel pada citra menjadi lebih beragam dan dapat merepresentasikan citra aslinya dengan lebih baik.

Hal ini sejalan dengan teori yaitu metode *thresholding* membuat perkiraan histogram citra dengan memilih set *threshold* dengan tepat. Ketika *threshold* atau ambang batas meningkat, maka perkiraan nilai frekuensi histogram menjadi lebih akurat dan lebih sesuai dengan nilai sebenarnya [3]. Ketika level *threshold* meningkat, nilai PSNR dan SSIM juga meningkat karena persebaran intensitas piksel pada citra semakin mendekati aslinya dan menghasilkan citra terkompresi dengan visual yang juga semakin baik dan menyerupai citra aslinya. Berdasarkan hal tersebut, dapat dikatakan bahwa hasil

kompresi citra algoritma *Differential Evolution* dengan inputan citra wajah menghasilkan grafik histogram dengan persebaran nilai intensitas piksel yang lebih baik seiring dengan peningkatan level *threshold* yang ditetapkan.

## 5. KESIMPULAN

Analisis penelitian ini menunjukkan bahwa metode *multi-level thresholding* menggunakan algoritma *Differential Evolution* (DE) dilakukan sebagai pendekatan dalam mempartisi citra ke dalam beberapa level ambang batas untuk mencapai kompresi citra dan tetap mempertahankan kualitas visual citra terkompresi. Kompresi citra DE menghasilkan nilai PSNR mencapai rentang antara 27 dB hingga 40 dB. Nilai PSNR yang diperoleh meningkat seiring dengan kenaikan level *threshold* yang ditetapkan dan telah mencapai nilai standar untuk kompresi citra *lossy*.

Kompresi citra DE menghasilkan nilai SSIM dengan rentang 0,83 hingga 0,97. Nilai SSIM yang diperoleh mendekati nilai 1 seiring dengan kenaikan level *threshold* yang menunjukkan kemiripan struktur antara citra input dengan citra terkompresi. Grafik histogram citra terkompresi dari penelitian ini menunjukkan persebaran nilai intensitas piksel yang semakin luas ketika level *threshold* meningkat dan berpengaruh terhadap kualitas visual yang dihasilkan pada citra terkompresi semakin baik.

## 6. KESIMPULAN

- [1] V. Lusiana, I. H. Al Amin, and F. A. Sutanto. 2022. Pengaruh Peningkatan Kualitas Citra Menggunakan Modifikasi Kontras Pada Kompresi Data RLE. *Building of Informatics, Technology and Science (BITS)*, 4(1), 270-276.
- [2] S. I. Murpratiwi and I. M. O. Widyantara. 2018. Pemilihan Algoritma Kompresi Optimal untuk Citra Digital Bitmap. *Majalah Ilmiah Teknologi Elektro*, vol. 17, no. 1, pp. 94–101.
- [3] S. Paul and B. Bandyopadhyay. 2014. A novel approach for image compression based on multi-level image thresholding using Shannon Entropy and Differential Evolution. *IEEE TechSym 2014 - 2014 IEEE Students' Technol. Symp.*, no.

February 2014, pp. 56–61.

- [4] S. A. Rather and P. S. Bala,. 2021. Constriction coefficient based particle swarm optimization and gravitational search algorithm for multilevel image thresholding. *Expert Syst.*, vol. 38, no. 7, pp. 1–36.
- [5] S. Kotte, P. Rajesh Kumar, and S. K. Injeti. 2018. An efficient approach for optimal multilevel thresholding selection for gray scale images based on improved differential search algorithm. *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 1043–1067.
- [6] E. Maria, Y. Yulianto, Y. P. Arinda, J. Jumiaty, and P. Nobel. 2018. Segmentasi Citra Digital Bentuk Daun Pada Tanaman Di Politani Samarinda Menggunakan Metode Thresholding. *J. Rekayasa Teknol. Inf.*, vol. 2, no. 1, p. 37.
- [7] I. K. A. B. Adnyana, I. M. O. Widyantara, and N. D. Wirastuti. 2021. Analisa Metode Shannon Entropy Dan Differential Evolution Untuk Kompresi Gambar. *Jurnal SPEKTRUM* vol. 8, no. 2, pp. 221–228.
- [8] S. Sarkar, S. Paul, R. Burman, S. Das, and S. S. Chaudhuri. 2015. A fuzzy entropy based multi-level image thresholding using differential evolution. *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8947, no. August, pp. 386–395.
- [9] U. Sara, M. Akter, and M. S. Uddin. 2019. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *J. Comput. Commun.*, vol. 07, no. 03, pp. 8–18.
- [10] F. Z. David R. Bull. 2021. *Intelligent Image and Video Compression*.
- [11] L. Li, A. A. Abd El-Latif, X. Yan, S. Wang, and X. Niu. 2012. A lossless secret image sharing scheme based on steganography. *Proc. 2012 2nd Int. Conf. Instrum. Meas. Comput. Commun. Control. IMCCC 2012*, no. May 2014, pp. 1247–1250.