

LAPORAN TUGAS KECIL 2
IF2211 STRATEGI ALGORITMA

Kompresi Gambar Dengan Metode Quadtree



Disusun oleh :

Ivan Wirawan 13523046

Carlo Angkisan 13523091

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
JL. GANESHA 10, BANDUNG 40132

2025

DAFTAR ISI

DAFTAR ISI.....	2
BAB I DESKRIPSI MASALAH DAN ALGORITMA.....	3
1.1 Quadtree.....	3
1.2 Algoritma Divide and Conquer.....	3
1.3 Penerapan Algoritma Divide and Conquer pada Kompresi Gambar dengan Metode Quadtree.....	4
BAB II IMPLEMENTASI PROGRAM DALAM BAHASA JAVA.....	8
2.1 Class QuadtreeNode.....	8
2.2 Class QuadtreeCompression.....	9
2.3 Class ErrorMetrics.....	11
2.4 Class ImageMatrix.....	12
2.5 Class IO.....	13
2.6 Class Main.....	14
2.7 Class UI.....	15
BAB III SOURCE CODE DAN STRUKTUR PROGRAM.....	17
3.1 Repozitori Github.....	17
3.2 Struktur Program.....	17
3.3 Source Code Program.....	18
3.3.1. Class Main.....	18
BAB IV HASIL PENGUJIAN DAN ANALISIS.....	42
4.1 Hasil Pengujian.....	42
4.2 Analisis Pengujian dan Algoritma.....	74
LAMPIRAN.....	79
Tabel Kelengkapan Spesifikasi.....	79
DAFTAR PUSTAKA.....	80

BAB I

DESKRIPSI MASALAH DAN ALGORITMA

1.1 Quadtree

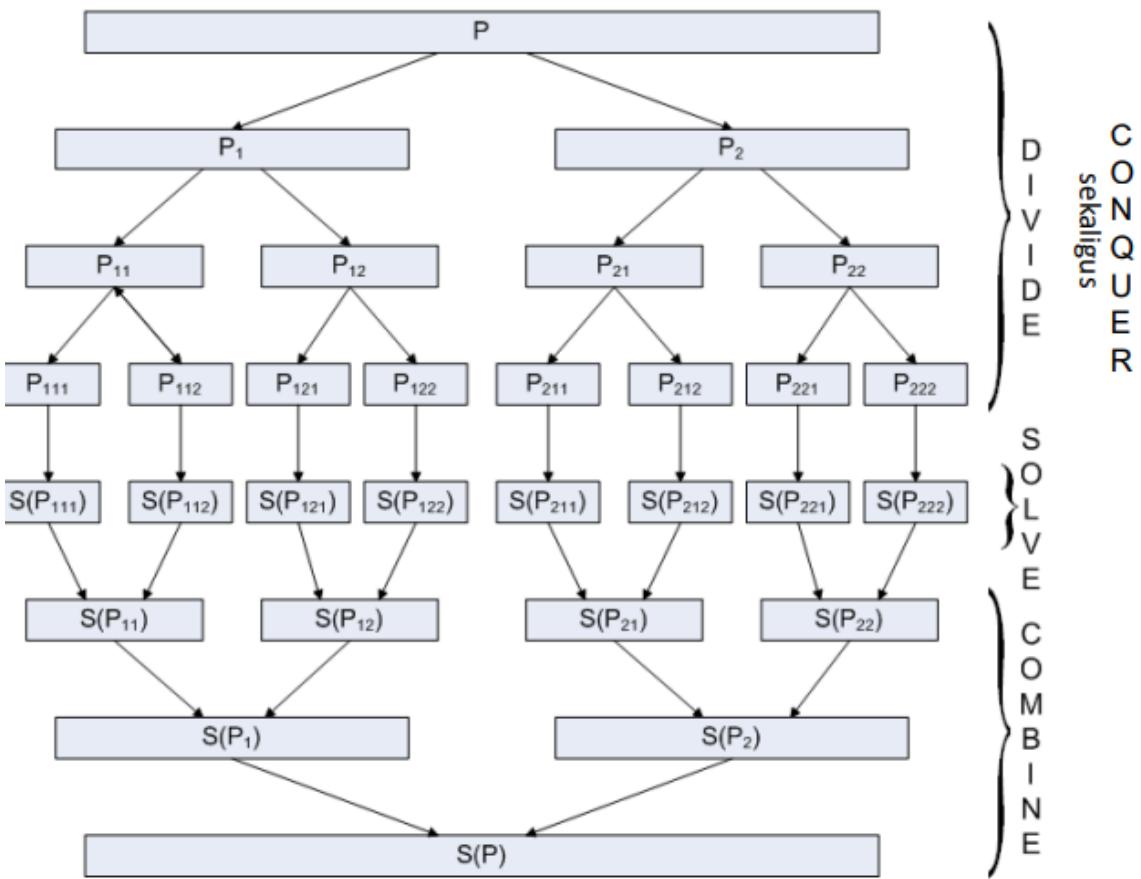
Quadtree adalah struktur data hierarkis yang digunakan untuk membagi ruang atau data menjadi bagian yang lebih kecil, yang sering digunakan dalam pengolahan gambar. Dalam konteks kompresi gambar, Quadtree membagi gambar menjadi blok-blok kecil berdasarkan keseragaman warna atau intensitas piksel. Prosesnya dimulai dengan membagi gambar menjadi empat bagian, lalu memeriksa apakah setiap bagian memiliki nilai yang seragam berdasarkan analisis sistem warna RGB, yaitu dengan membandingkan komposisi nilai merah (R), hijau (G), dan biru (B) pada piksel-piksel di dalamnya. Jika bagian tersebut tidak seragam, maka bagian tersebut akan terus dibagi hingga mencapai tingkat keseragaman tertentu atau ukuran minimum yang ditentukan.

Dalam implementasi teknis, sebuah Quadtree direpresentasikan sebagai simpul (node) dengan maksimal empat anak (children). Simpul daun (leaf) merepresentasikan area gambar yang seragam, sementara simpul internal menunjukkan area yang masih membutuhkan pembagian lebih lanjut. Setiap simpul menyimpan informasi seperti posisi (x, y), ukuran (width, height), dan nilai rata-rata warna atau intensitas piksel dalam area tersebut. Struktur ini memungkinkan pengkodean data gambar yang lebih efisien dengan menghilangkan redundansi pada area yang seragam. QuadTree sering digunakan dalam algoritma kompresi lossy karena mampu mengurangi ukuran file secara signifikan tanpa mengorbankan detail penting pada gambar.

1.2 Algoritma Divide and Conquer

Pada dasarnya, algoritma *Divide and Conquer* adalah sebuah konsep algoritma yang akan membagi sebuah persoalan besar menjadi beberapa persoalan kecil. Melalui pembagian tersebut, setiap upa-persoalan akan diselesaikan secara langsung maupun rekursif. Hasil dari penyelesaian tersebut akan digabungkan dengan solusi-solusi lainnya agar membentuk sebuah jawaban besar bagi permasalahan utamanya.

Tujuan utama dari algoritma *Divide and Conquer* adalah membagi sebuah persoalan besar menjadi beberapa upa-persoalan yang mirip dengan persoalan besar namun memiliki ukuran lebih kecil. Harapannya, dengan membagi sebuah persoalan besar menjadi kecil, maka persoalan kecil itu dapat diselesaikan langsung dengan mudah. Akan tetapi, apabila masih terlalu besar, maka dapat menggunakan penyelesaian secara rekursif. Setelah berhasil menyelesaikan seluruh upa-persoalan, masing-masing solusinya akan digabungkan hingga membentuk solusi utama.



Keterangan:

P = persoalan
 S = solusi

Syarat dari penggunaan algoritma *Divide and Conquer* pada sebuah persoalan adalah objek dari masalah tersebut dapat dibagi menjadi beberapa bagian dengan karakteristik yang sama. Contohnya adalah tabel, matriks, eksponen, dan polinom. Apabila memiliki upa-persoalan yang dengan karakteristik yang sama namun lebih kecil, waktu dan sumber daya yang digunakan dalam menyelesaikan masalah akan lebih kecil sehingga meningkatkan efisiensi mesin.

1.3 Penerapan Algoritma Divide and Conquer pada Kompresi Gambar dengan Metode Quadtree

Adapun implementasi Algoritma Divide and Conquer pada kompresi gambar dengan metode quadtree menggunakan metode rekursif. Setelah program berhasil meminta masukan dari pengguna mengenai variabel-variabel yang akan digunakan seperti alamat gambar masukan dan keluaran, metode perhitungan, ambang batas, dan ukuran blok minimum, program akan menjalankan algoritma *Divide and Conquer* pada gambar yang telah diterima.

Sebelumnya, program akan mengubah bentuk gambar menjadi matriks berwarna. Dengan pengubahan ini, algoritma *Divide and Conquer* dapat dilakukan dengan pemecahan matriks menjadi blok-blok yang lebih kecil dari utamanya. Blok besar yang masih berkondisi tidak homogen (nilai error yang masih lebih besar daripada ambang batas yang diinginkan) dan ukurannya memungkinkan untuk pembagian, akan dibagi menjadi empat kuadran yaitu atas-kiri, atas-kanan, bawah-kiri, dan bawah-kanan. Dalam struktur Quadtree, awalnya satu matriks utama adalah sebuah simpul akar yang mewakili seluruh gambar, setelah pembagian, maka hasil pembagian itulah yang akan menjadi anak-anak dari simpul akar tersebut. Setelah itu, algoritma pembentukan Quadtree akan dipanggil kembali kepada pecahan-pecahan blok utama tadi. Hal ini akan membagi sebuah gambar utuh menjadi bagian-bagian yang dengan kehomogenan dan ukuran yang sesuai. Selagi pemecahan, data Quadtree akan terus mengalami pembaharuan sehingga terbentuk struktur data pohon yang berisikan alur pemecahan matriks.

Setelah kondisi pemecahan blok yaitu variasi warna dan ukuran yang kecil, maka proses pembagian akan berhenti pada setiap blok. Hasil pembagian tersebut akan direpresentasikan dengan sebuah nilai rata-rata warna. Hal ini akan melangsungkan sebuah kompresi pada gambar. Tujuannya adalah mengurangi variasi dari warna yang ada pada gambar sehingga ukurannya dapat lebih kecil.

Setelah kompresi dilakukan terhadap blok-blok yang memenuhi syarat, gambar akan dibentuk kembali. Proses penggabungan pada algoritma *Divide and Conquer* yang diaplikasikan dalam program ini memanfaatkan struktur pohon *Quadtree* yang ada. Rekonstruksi gambar akan dilakukan melalui himpunan blok-blok homogen yang terdapat pada simpul daun atau simpul paling bawah yang tidak memiliki anak lagi.

Sebagai sebuah kesimpulan, algoritma *Divide and Conquer* pada kompresi gambar dengan metode *Quadtree* bekerja dalam beberapa tahapan yaitu :

1. Inisialisasi : Setelah menghimpun seluruh data yang dibutuhkan melalui masukan dari pengguna, gambar akan dibentuk sebagai sebuah matriks. Selain itu, gambar juga akan dibentuk sebagai sebuah simpul akar yang mencakup seluruh gambar.
2. Pembagian (*Divide*) : Dalam membagi gambar menjadi 4 pecahan lainnya, algoritma akan menghitung nilai homogenitas dari setiap blok sebelum memecahnya. Apabila nilai errornya melebihi dari keinginan pengguna serta ukurannya memenuhi, maka blok tersebut akan dipecah menjadi 4 bagian selayaknya sebuah kuadran 1-4. Jika tidak memenuhi, maka blok tersebut akan menjadi simpul daun.. Apabila nilai error yang terdapat pada sebuah blok besar melebihi keinginan pengguna dan ukuran bloknya masih cukup besar. Proses inilah yang akan membagi sebuah masalah besar yaitu gambar besar, menjadi 4 bagian gambar yang lebih kecil. Proses pembagian ini akan dipanggil kembali untuk setiap pecahan blok dari gambar utamanya.
3. Penyelesaian (*Conquer*) : Setelah berhasil memecah gambar menjadi banyak blok kecil yang sudah tidak bisa diganti atau memiliki nilai error yang rendah, maka

algoritma ini akan menghitung nilai rata-rata warna dari blok tersebut yang menjadi nilai representasi warna blok tersebut. Proses penggantian warna dari yang heterogen menjadi sebuah warna rata-rata inilah yang melangsungkan sebuah kompresi pada gambar sehingga mengurangi ukuran berkas gambar tersebut.

4. Peng gabungan (*Combine*) : Apabila proses pemecahan dan proses kompresi setiap blok kecil sudah selesai dilaksanakan, maka gambar tersebut akan direkonstruksi ulang. Dengan struktur pohon *Quadtree* yang telah dibentuk, maka gambar ini dapat dibentuk kembali dengan simpul-simpul daun yang dimilikinya. Hasil akhirnya adalah sebuah gambar dengan ukuran panjang dan lebar yang sama, namun nilai warna yang lebih homogen berdasarkan masukan yang diberikan pengguna.

Algoritma tersebut dapat dituliskan dalam Pseudocode sebagai berikut.

```

procedure QuadTreeCompression (input G : image, T : real, minSize : integer, width : integer, height : integer) → QuadTreeNode
{ Menyelesaikan kompresi citra G menggunakan metode Quadtree berbasis algoritma divide and conquer

Masukan :
    G           : gambar berukuran width × height
    T           : threshold error untuk menentukan homogenitas
    minSize     : ukuran minimum blok (dalam piksel) agar masih boleh dibagi

Luaran :
    root       : simpul akar Quadtree, mewakili struktur pohon hasil kompresi

Variabel :
    root        : QuadTreeNode
    errorBlock   : real
    halfWidth    : integer
    halfHeight   : integer
}

Algoritma:
    root ← CreateNode(G, x, y, width, height)
    errorBlock ← CalculateError(G)

    halfWidth ← width / 2
    halfHeight ← height / 2

    if ( (errorBlock ≤ T)
        OR (width(G) * height(G) < minSize) )
        OR width(G) <= 1 OR height(G) <= 1
        OR ((width(G)/2) * (height(G)/2)) < minSize) then
            root.isLeaf ← true
            root.averageColor ← CalculateAverageColor(G)

    else

```

```
root.children[1] ← QuadTreeCompression(G, x, y, halfWidth,  
halfHeight, T, minSize)  
root.children[2] ← QuadTreeCompression(G, x + halfWidth, y,  
width - halfWidth, halfHeight, T, minSize)  
root.children[3] ← QuadTreeCompression(G, x, y +  
halfHeight, halfWidth, height - halfHeight, T, minSize)  
root.children[4] ← QuadTreeCompression(G, x + halfWidth, y +  
halfHeight, width - halfWidth, height - halfHeight, T, minSize)  
  
root.isLeaf ← false  
endif  
  
→ root  
}
```

BAB II

IMPLEMENTASI PROGRAM DALAM BAHASA JAVA

Dalam implementasi program Kompresi Gambar dengan Metode Quadtree dalam bahasa Java, pendekatan paradigma berorientasi objek (OOP) digunakan karena Java sendiri berbasis OOP. Dengan pendekatan ini, setiap komponen diorganisir dalam bentuk kelas (*class*) yang memiliki tanggung jawab spesifik, sehingga meningkatkan modularitas dan kemudahan dalam pengelolaan kode. Berikut adalah kelas-kelas yang digunakan dalam program ini.

2.1 Class QuadtreeNode

- Attribute

Nama	Tipe	Deskripsi
x	private int	Koordinat x (posisi horizontal) dari blok gambar yang diwakili oleh node.
y	private int	Koordinat y (posisi vertikal) dari blok gambar yang diwakili oleh node.
width	private int	Lebar blok gambar pada node ini.
height	private int	Tinggi blok gambar pada node ini.
imageMatrix	private ImageMatrix	Referensi ke objek ImageMatrix yang memuat data gambar; digunakan untuk menghitung rata-rata warna dari blok yang diberikan.
children	private QuadtreeNode[]	Array anak (4 node) yang merupakan hasil pembagian blok jika node tidak termasuk daun (leaf).
averageColor	private int	Nilai rata-rata warna (dalam format integer RGB) yang dihitung dari blok gambar pada node ini.
isLeaf	private boolean	Menandakan apakah node merupakan daun (tidak terbagi lebih lanjut) atau tidak.

- Method

Nama	Tipe	Deskripsi
QuadtreeNode	public	Konstruktor yang menginisialisasi node

	constructor	dengan blok gambar tertentu berdasarkan posisi (x,y), ukuran (width & height) dan menghitung nilai rata-rata warnanya.
calculateAverageColor	private int	Metode privat untuk menghitung rata-rata warna (RGB) pada blok gambar yang diwakili oleh node.
getChildren	public QuadtreeNode[]	Mengembalikan array anak (children) node; jika null, maka node tersebut merupakan daun.
getAverageColor	public int	Mengembalikan nilai rata-rata warna dari node.
getX	public int	Mengembalikan nilai koordinat x untuk blok gambar.
getY	public int	Mengembalikan nilai koordinat y untuk blok gambar.
getWidth	public int	Mengembalikan lebar blok gambar.
getHeight	public int	Mengembalikan tinggi blok gambar.
isLeaf	public boolean	Mengembalikan nilai boolean yang menunjukkan apakah node merupakan daun (tidak terbagi).
setChildren	public void	Menetapkan anak-anak node dan mengubah status node menjadi non-daun (isLeaf = false).

2.2 Class QuadtreeCompression

- Attribute

Nama	Tipe	Deskripsi
originalImage	private ImageMatrix	Objek ImageMatrix yang menyimpan data gambar asli.
threshold	private double	Nilai threshold awal atau optimal yang digunakan sebagai batas error saat kompresi.
minimumBlockSize	private int	Ukuran blok minimum yang harus dipenuhi

		agar blok gambar bisa dibagi lebih lanjut.
errorMethod	private int	Metode error yang dipilih (1: Variance, 2: Mean Absolute Deviation, 3: Max Pixel Difference, 4: Entropy, 5: SSIM).
targetCompressionPercentage	private double	Persentase target kompresi (nilai yang diharapkan) dalam satuan persentase.
inputPath	package-private String	Path file dari gambar input. (Tidak dideklarasikan sebagai private, bersifat default/package-private)

- Method

Nama	Tipe	Deskripsi
QuadtreeCompression	public constructor	Konstruktor untuk menginisialisasi objek QuadtreeCompression dengan parameter gambar asli, metode error, threshold, ukuran blok minimum, target kompresi, dan path input.
compress	public QuadtreeNode	Melakukan proses kompresi gambar dengan membangun Quadtree berdasarkan error yang dihitung; jika target kompresi diaktifkan, mencari threshold optimal terlebih dahulu.
buildQuadtree	private QuadtreeNode	Metode rekursif untuk membangun struktur Quadtree dari gambar dengan membagi blok gambar apabila error melebihi threshold (dan memenuhi ukuran minimum jika berlaku).
findOptimalThreshold	private double	Melakukan pencarian threshold optimal secara biner (binary search) agar kompresi gambar mendekati target compression percentage.
calculateTreeDepth	public int	Menghitung dan mengembalikan kedalaman maksimum Quadtree yang telah dibangun.
calculateNodeCount	public int	Menghitung jumlah node dalam struktur Quadtree secara keseluruhan.

getThreshold	public double	Mengembalikan nilai threshold yang digunakan (setelah optimasi jika berlaku).
--------------	---------------	---

2.3 Class ErrorMetrics

- Method

Nama	Tipe	Deskripsi
calculateVariance	private static double	Menghitung variance (ragam) total untuk setiap kanal warna (R, G, B) dalam sebuah blok gambar, dengan hasil rata-rata dari ketiga kanal.
calculateChannel Variance	private static double	Menghitung variance untuk kanal tertentu (0: R, 1: G, 2: B) dalam blok gambar.
calculateMean	private static double	Menghitung rata-rata nilai piksel untuk kanal tertentu dalam blok gambar.
calculateMeanAbs oluteDeviation	private static double	Menghitung nilai deviasi absolut rata-rata (Mean Absolute Deviation) untuk masing-masing kanal, dan mengembalikan rata-ratanya.
calculateMaxPixel Difference	private static double	Menghitung perbedaan maksimum piksel (selisih antara nilai maksimum dan minimum) untuk tiap kanal, kemudian mengembalikan rata-ratanya.
calculateEntropy	private static double	Menghitung entropi (pengukuran ketidakteraturan/distribusi) untuk masing-masing kanal dan mengembalikan rata-ratanya.
calculateSSIM	public static double	Menghitung indeks SSIM (Structural Similarity Index) antara gambar asli dan gambar terkompresi dengan bobot masing-masing kanal (R, G, B).
calculateChannelsS IM	private static double	Menghitung SSIM untuk sebuah kanal, menggunakan perhitungan rata-rata, variance, dan covariance antara gambar asli dan terkompresi.

calculateCovariance	private static double	Menghitung kovarians antara dua blok gambar untuk kanal tertentu, yang digunakan dalam perhitungan SSIM.
calculateError	public static double	Menghitung error berdasarkan metode yang dipilih (1: Variance, 2: Mean Absolute Deviation, 3: Max Pixel Difference, 4: Entropy, 5: SSIM).
calculateBlockSSIM	public static double	Menghitung nilai SSIM blok gambar dengan membuat blok kompresi sederhana berdasarkan rata-rata warna, lalu membandingkannya dengan gambar asli.

2.4 Class ImageMatrix

- Attribute

Nama	Tipe	Deskripsi
pixels	private int[][]	Array 3 dimensi yang menyimpan nilai piksel untuk masing-masing kanal (R, G, B) dengan indeks [kanal][x][y].
width	private int	Lebar gambar (jumlah kolom piksel).
height	private int	Tinggi gambar (jumlah baris piksel).

- Method

Nama	Tipe	Deskripsi
ImageMatrix(BufferedImage image)	public constructor	Konstruktor yang menginisialisasi objek ImageMatrix dengan mengambil data dari objek BufferedImage dan memecah nilai RGB tiap piksel ke array pixels.
ImageMatrix(int width, int height)	public constructor	Konstruktor yang membuat ImageMatrix dengan ukuran lebar dan tinggi tertentu, dan menginisialisasi array pixels.
toBufferedImage	public BufferedImage	Mengonversi ImageMatrix kembali menjadi objek BufferedImage dengan merakit nilai RGB dari array pixels.
getPixels	public	Mengembalikan array tiga dimensi pixels.

	int[][][]	
getWidth	public int	Mengembalikan lebar gambar.
getHeight	public int	Mengembalikan tinggi gambar.
getPixel	public int	Mengambil nilai piksel untuk kanal tertentu pada koordinat (x, y).
setPixel	public void	Menetapkan nilai piksel untuk kanal tertentu pada koordinat (x, y).
getAverageColor	public int	Menghitung rata-rata warna dalam blok gambar yang ditentukan dengan menghitung rata-rata nilai untuk tiap kanal, kemudian menggabungkannya menjadi satu nilai integer RGB.

2.5 Class IO

- Method

Nama	Tipe	Deskripsi
readImage	public static ImageMatrix	Membaca file gambar dari path yang diberikan dan mengembalikan objek ImageMatrix.
calcFileSize	public static long	Menghitung dan mengembalikan ukuran file dalam byte berdasarkan path file yang diberikan.
calcCompressionPercentage	public static double	Menghitung persentase kompresi dengan membandingkan ukuran file asli dan file terkompresi.
reconstructImageFromQuadtree	public static ImageMatrix	Mereconstruct (menyusun kembali) gambar dari struktur Quadtree yang dihasilkan dengan menggunakan informasi posisi dan ukuran dari tiap node.
writeCompressedImage	public static void	Menulis (menyimpan) ImageMatrix sebagai file gambar ke output path yang ditentukan.
reconstructGIFFromQuadtree	public static List<BufferedImage>	Membuat list frame BufferedImage dengan merekonstruksi gambar dari Quadtree untuk tiap tingkat (depth) sebagai persiapan

		pembuatan file GIF animasi.
reconstructNodeFor GIF	public static void	Versi modifikasi dari metode rekonstruksi node yang digunakan untuk pembuatan GIF, dengan mempertimbangkan batas maksimum.
createCompressionG if	public static void	Membuat file GIF yang berisi frame-frame kompresi dengan pengaturan delay time antar frame.
isValidImagePath	public static boolean	Memvalidasi apakah path file gambar valid (ekstensi yang sesuai dan file tersebut ada).
getExtension	private static String	Mengambil ekstensi file dari path (dipanggil secara internal oleh metode validasi).
hasSameExtension	public static boolean	Memeriksa apakah dua file memiliki ekstensi yang sama.
isValidThreshold	public static boolean	Memvalidasi apakah nilai threshold berada dalam rentang yang sesuai berdasarkan metode error yang dipilih.

2.6 Class Main

- Attribute

Nama	Tipe	Deskripsi
RESET	private static final String	Kode warna ANSI untuk mereset warna output di terminal.
RED	private static final String	Kode warna ANSI untuk warna merah.
GREEN	private static final String	Kode warna ANSI untuk warna hijau.
YELLOW	private static final String	Kode warna ANSI untuk warna kuning.

BLUE	private static final String	Kode warna ANSI untuk warna biru.
PURPLE	private static final String	Kode warna ANSI untuk warna ungu.
WHITE	private static final String	Kode warna ANSI untuk warna putih.

- Method

Nama	Tipe	Deskripsi
main	public static void	Method utama yang mengeksekusi aplikasi, menangani input pengguna, validasi file, pemilihan metode error, pengaturan threshold, kompresi gambar melalui Quadtree, dan output hasil serta GIF jika diminta.

2.7 Class UI

- Attribute

Nama	Tipe	Deskripsi
RESET	private static final String	Kode warna ANSI untuk mereset warna output di terminal.
RED	private static final String	Kode warna ANSI untuk warna merah.
GREEN	private static final String	Kode warna ANSI untuk warna hijau.
YELLOW	private static final String	Kode warna ANSI untuk warna kuning.

BLUE	private static final String	Kode warna ANSI untuk warna biru.
PURPLE	private static final String	Kode warna ANSI untuk warna ungu.
WHITE	private static final String	Kode warna ANSI untuk warna putih.

- Method

Nama	Tipe	Deskripsi
printLogo	public static	Menampilkan logo aplikasi dengan efek warna dan format ASCII art.
printHeader	public static	Menampilkan header aplikasi dengan judul tool Quadtree Image Compression.
printProgressBar	public static void	Menampilkan progress bar di terminal untuk menunjukkan perkembangan proses kompresi berdasarkan persentase yang telah dicapai.
printErrorMethodInfo	public static void	Menampilkan informasi dan pilihan terkait metode pengukuran error (Variance, MAD, Max Pixel Difference, Entropy, SSIM).
printResultMenu	public static void	Menampilkan hasil kompresi, termasuk waktu eksekusi, ukuran file asli dan kompresi, kedalaman pohon, total node, dan informasi file output (gambar & GIF jika ada).
getThresholdRange	public static String	Mengembalikan string yang menunjukkan range nilai threshold yang valid berdasarkan metode error yang dipilih.
getMethodNames	public static String	Mengembalikan nama metode error dalam bentuk string sesuai pilihan (misal: "Variance", "Mean Absolute Deviation", dll.).

BAB III

SOURCE CODE DAN STRUKTUR PROGRAM

3.1 Repositori Github

Berikut tautan repository github kelompok kami untuk Tugas Kecil 2 IF2211 Strategi Algoritma :

https://github.com/ivan-wirawan/Tucil2_13523046_13523091

3.2 Struktur Program

Program memiliki struktur folder sebagai berikut.

- **src** : berisi source code program dalam bentuk file **.java**.
- **bin** : berisi hasil kompilasi dalam bentuk file **.class** yang siap dijalankan.
- **test** : berisi hasil pengujian yang dicantumkan pada laporan.
- **doc** : berisi laporan tugas kecil dan dokumentasi program.

3.3 Source Code Program

3.3.1. Class Main

```
 1 public class Main {
 2     private static final String RESET = "\u001B[0m";
 3     private static final String RED = "\u001B[31m";
 4     private static final String GREEN = "\u001B[32m";
 5     private static final String YELLOW = "\u001B[33m";
 6     private static final String BLUE = "\u001B[34m";
 7     private static final String PURPLE = "\u001B[35m";
 8     private static final String WHITE = "\u001B[1;37m";
 9
10     public static void main(String[] args) {
11         Scanner scanner = new Scanner(System.in);
12
13         try {
14             UI.printLogo();
15             UI.printHeader();
16             String inputPath;
17             while (true) {
18                 System.out.print(WHITE + "\n █ Absolute path of input image: " + RESET);
19                 inputPath = scanner.nextLine().trim();
20                 if (IO.isValidImagePath(inputPath)) {
21                     System.out.println(GREEN + " ✅ Valid image found!" + RESET);
22                     break;
23                 }
24                 System.out.println(RED + " ❌ Invalid image path or format." + RESET);
25             }
26
27             int errorMethod = 0;
28             while (true) {
29                 UI.printErrorMethodInfo();
30
31                 System.out.print(WHITE + "\n 🖊 Choose method (1-5): " + RESET);
32
33                 try {
34                     errorMethod = Integer.parseInt(scanner.nextLine());
35                     if (errorMethod >= 1 && errorMethod <= 5) {
36                         System.out.println(GREEN + " ✅ Method selected: " +
37                         UI.getMethodName(errorMethod) + RESET);
38                         break;
39                     }
40                 } catch (Exception ignored) {
41                 }
42                 System.out.println(RED + " ❌ Please select a valid number (1-4)." +
43                 RESET);
43             }
44         }
45     }
46 }
```

```

1 double threshold;
2         while (true) {
3             System.out.print(
4                 WHITE + "\n 🔍 Enter threshold " + YELLOW +
5                 UI.getThresholdRange(errorMethod) + RESET + ": ");
6             try {
7                 threshold = Double.parseDouble(scanner.nextLine());
8                 if (IO.isValidThreshold(threshold, errorMethod)) {
9                     System.out.println(GREEN + " ✅ Threshold set!" + RESET);
10                break;
11            }
12        } catch (Exception ignored) {
13    }
14    System.out.println(RED + " ❌ Threshold out of range for selected
method." + RESET);
15}
16
17 int minBlock;
18 while (true) {
19     System.out.print(WHITE + "\n 🔍 Minimum block size: " + RESET);
20     try {
21         minBlock = Integer.parseInt(scanner.nextLine());
22         if (minBlock > 0) {
23             System.out.println(GREEN + " ✅ Minimum block size set to " +
minBlock + RESET);
24             break;
25         }
26     } catch (Exception ignored) {
27     }
28     System.out.println(RED + " ❌ Must be a positive integer." + RESET);
29}
30
31 double compressionTarget;
32 while (true) {
33     System.out.print(WHITE + "\n 🔍 Target compression percentage " + YELLOW
+ "(0.0-1.0)" + RESET + ": ");
34     try {
35         compressionTarget = Double.parseDouble(scanner.nextLine());
36         if (compressionTarget >= 0.0 && compressionTarget <= 1.0) {
37             if (compressionTarget == 0.0) {
38                 System.out.println(BLUE + " 🚧 Target compression disabled" +
RESET);
39             } else {
40                 System.out.println(GREEN + " ✅ Target set to " +
(compressionTarget * 100) + "%" + RESET);
41             }
42             break;
43         }
44     } catch (Exception ignored) {
45     }
46     System.out.println(RED + " ❌ Must be between 0.0 and 1.0." + RESET);
47 }

```

```

1 String outputPath;
2         while (true) {
3             System.out.print(WHITE + "\n 📁 Absolute path for compressed image: " +
RESET);
4             outputPath = scanner.nextLine().trim();
5
6             if (outputPath.equals(inputPath)) {
7                 System.out.println(RED + " ✗ Output path cannot be the same as input
path." + RESET);
8                 continue;
9             }
10
11             File outputFile = new File(outputPath);
12             File outputDir = outputFile.getParentFile();
13             if (outputDir != null && !outputDir.exists()) {
14                 System.out
15                     .println(RED + " ✗ Output directory doesn't exist: " +
outputDir.getAbsolutePath() + RESET);
16                 continue;
17             }
18
19             if (outputFile.exists()) {
20                 System.out.println(RED + " ✗ Output file already exists: " +
outputPath + RESET);
21                 continue;
22             }
23
24             if (IO.hasSameExtension(inputPath, outputPath)) {
25                 System.out.println(GREEN + " ✓ Output path set!" + RESET);
26                 break;
27             }
28             System.out.println(RED + " ✗ Output file must have the same extension as
input." + RESET);
29         }
30
31         boolean saveGif = false;
32         String gifPath = null;
33         while (true) {
34             System.out.print(WHITE + "\n 🎨 Do you want to save compression GIF? " +
YELLOW + "(y/n)" + RESET + ": ");
35             String gifChoice = scanner.nextLine().trim().toLowerCase();
36             if (gifChoice.equals("y")) {
37                 saveGif = true;
38                 while (true) {
39                     System.out.print(WHITE + "\n 📂 Enter absolute path to save GIF:
" + RESET);
40                     gifPath = scanner.nextLine().trim();
41
42                     if (gifPath.equals(inputPath) || gifPath.equals(outputPath)) {
43                         System.out.println(RED + " ✗ GIF path cannot be the same as
input or output path." + RESET);
44                         continue;
45                     }

```

```
● ● ●

1             File gifFile = new File(gifPath);
2             File gifDir = gifFile.getParentFile();
3             if (gifDir != null && !gifDir.exists()) {
4                 System.out.println(
5                     RED + " ❌ GIF directory doesn't exist: " +
6                     gifDir.getAbsolutePath() + RESET);
7                 continue;
8             }
9             if (gifFile.exists()) {
10                 System.out.println(RED + " ❌ GIF file already exists: " +
11                     gifPath + RESET);
12                 continue;
13             }
14             if (gifPath.toLowerCase().endsWith(".gif")) {
15                 System.out.println(GREEN + " ✅ GIF path set!" + RESET);
16                 break;
17             }
18             System.out.println(RED + " ❌ Output file must have a .gif
19                 extension." + RESET);
20             }
21             break;
22         } else if (gifChoice.equals("n")) {
23             saveGif = false;
24             break;
25         }
26         System.out.println(RED + " ❌ Please type 'y' or 'n'." + RESET);
27     }
28 }
```

```

1 System.out.println("\n" + PURPLE + "\n ⚙ Processing image compression..." + RESET);
2     UI.printProgressBar(0);
3
4     long startTime = System.nanoTime();
5
6     ImageMatrix original = IO.readImage(inputPath);
7     int width = original.getWidth();
8     int height = original.getHeight();
9     UI.printProgressBar(20);
10
11    QuadtreeCompression compressor = new QuadtreeCompression(
12        original, errorMethod, threshold, minBlock, compressionTarget*100,
13        outputPath);
14    UI.printProgressBar(40);
15
16    QuadtreeNode root = compressor.compress();
17    UI.printProgressBar(70);
18
19    int treeDepth = compressor.calculateTreeDepth(root);
20    int nodeCount = compressor.calculateNodeCount(root);
21
22    UI.printProgressBar(80);
23
24    ImageMatrix compressed = IO.reconstructImageFromQuadtree(root, width, height);
25
26    IO.writeCompressedImage(compressed, outputPath);
27    UI.printProgressBar(90);
28
29    if (saveGif && gifPath != null) {
30        System.out.println(BLUE + "\n\n 🎨 Generating compression GIF..." +
31        RESET);
32        List<BufferedImage> frames = IO.reconstructGIFFromQuadtree(root, width,
33        height,
34        treeDepth);
35        IO.createCompressionGif(frames, gifPath, 500);
36    }
37    UI.printProgressBar(100);
38
39    long endTime = System.nanoTime();
40    double executionTime = (endTime - startTime) / 1_000_000.0;
41    double compressionPercentage = IO.calcCompressionPercentage(inputPath,
42        outputPath);
43
44    threshold = compressor.getThreshold();
45    UI.printResultMenu(treeDepth, nodeCount, executionTime, compressionPercentage,
46        outputPath, outputPath, gifPath, saveGif, compressionTarget, threshold);
47
48    System.out.println("\n" + PURPLE + "Thank you for using Quadtree Image
49    Compression!" + RESET);
50
51    } catch (IOException e) {
52        System.err.println(RED + " ❌ Error processing image: " + e.getMessage() +
53        RESET);
54    } finally {
55        scanner.close();
56    }
57}

```

3.3.2. Class QuadtreeNode

```
 1  public class QuadtreeNode {
 2      private int x, y;
 3      private int width, height;
 4      private ImageMatrix imageMatrix;
 5      private QuadtreeNode[] children;
 6      private int averageColor;
 7      private boolean isLeaf;
 8
 9      public QuadtreeNode(ImageMatrix imageMatrix, int x, int y, int width, int height) {
10          this.imageMatrix = imageMatrix;
11          this.x = x;
12          this.y = y;
13          this.width = width;
14          this.height = height;
15          this.children = new QuadtreeNode[4];
16          this.isLeaf = true;
17          this.averageColor = calculateAverageColor();
18      }
19      private int calculateAverageColor() {
20          return imageMatrix.getAverageColor(x, y, width, height);
21      }
22
23      public QuadtreeNode[] getChildren() {
24          return children;
25      }
26      public int getAverageColor() {
27          return averageColor;
28      }
29      public int getX() {
30          return x;
31      }
32      public int getY() {
33          return y;
34      }
35      public int getWidth() {
36          return width;
37      }
38      public int getHeight() {
39          return height;
40      }
41      public boolean isLeaf() {
42          return isLeaf;
43      }
44      public void setChildren(QuadtreeNode[] children) {
45          this.children = children;
46          this.isLeaf = false;
47      }
48  }
```

3.3.4. Class QuadtreeCompression

```
 1 public class QuadtreeCompression {
 2     private ImageMatrix originalImage;
 3     private double threshold;
 4     private int minimumBlockSize;
 5     private int errorMethod;
 6     private double targetCompressionPercentage;
 7     String inputPath;
 8
 9     public QuadtreeCompression(ImageMatrix originalImage, int errorMethod,
10         double threshold, int minimumBlockSize, double targetCompressionPercentage,
11         String inputPath) {
12         this.originalImage = originalImage;
13         this.errorMethod = errorMethod;
14         this.threshold = threshold;
15         this.minimumBlockSize = minimumBlockSize;
16         this.targetCompressionPercentage = targetCompressionPercentage;
17         this.inputPath = inputPath;
18     }
19
20     public QuadtreeNode compress() {
21         // Bonus: target Compression Percentage
22         boolean useMinimumBlockSize = true;
23         if (targetCompressionPercentage > 0) {
24             this.threshold = findOptimalThreshold();
25             useMinimumBlockSize = false;
26         }
27
28         return buildQuadtree(originalImage, 0, 0, originalImage.getWidth(),
29             originalImage.getHeight(),
30             0, this.threshold, useMinimumBlockSize);
31     }
32
33     private QuadtreeNode buildQuadtree(ImageMatrix originalImage, int x, int y, int width,int height, int currentDepth, double threshold, boolean useMinimumBlockSize) {
34         QuadtreeNode node = new QuadtreeNode(originalImage, x, y, width, height);
35
36         if (width <= 1 || height <= 1) {
37             return node;
38         }
39
40         double error = ErrorMetrics.calculateError(originalImage, x, y, width, height,
41             errorMethod);
42
43         int halfWidth = width / 2;
44         int halfHeight = height / 2;
45
46         boolean shouldDivide;
47
48         if (useMinimumBlockSize) {
49             shouldDivide = (error > threshold) && (width * height >= minimumBlockSize)
50                 && (halfWidth * halfHeight >= minimumBlockSize);
51         } else {
52             shouldDivide = (error > threshold);
53         }
54     }
55 }
```

```

1 if (shouldDivide) {
2     QuadtreeNode[] children = new QuadtreeNode[4];
3
4         // Top-left
5         children[0] = buildQuadtree(originalImage, x, y, halfWidth, halfHeight,
6             currentDepth + 1,
7                     threshold, useMinimumBlockSize);
8
9         // Top-right
10        children[1] = buildQuadtree(originalImage, x + halfWidth, y, width -
11            halfWidth, halfHeight,
12                    currentDepth + 1, threshold, useMinimumBlockSize);
13
14         // Bottom-left
15         children[2] = buildQuadtree(originalImage, x, y + halfHeight, halfWidth,
16             height - halfHeight, currentDepth + 1, threshold,
17                     useMinimumBlockSize);
18
19         // Bottom-right
20         children[3] = buildQuadtree(originalImage, x + halfWidth, y + halfHeight,
21             width - halfWidth, height - halfHeight, currentDepth + 1, threshold,
22                     useMinimumBlockSize);
23
24         node.setChildren(children);
25     }
26
27     return node;
28 }
29
30 private double findOptimalThreshold() {
31     double minThreshold = 0.0;
32     double maxThreshold;
33
34     if (errorMethod == 1) { // Variance
35         maxThreshold = 16256.25;
36     } else if (errorMethod == 2) { // Mean Absolute Deviation
37         maxThreshold = 127.5;
38     } else if (errorMethod == 3) { // Max Pixel Difference
39         maxThreshold = 255.0;
40     } else if (errorMethod == 4) { // Entropy
41         maxThreshold = 8.0;
42     } else if (errorMethod == 5) { // SSIM
43         maxThreshold = 1.0;
44     } else {
45         maxThreshold = 255.0;
46     }
47
48     double currentThreshold;
49     double currentCompression;
50     double epsilon = 0.001;
51     int maxIterations = 50;
52     int iterations = 0;

```

```
 1 String originalExtension = inputPath.substring(inputPath.lastIndexOf('.'));  
 2     String tempDirPath = "test/temp";  
 3     String tempPath = tempDirPath + "/temp" + originalExtension;  
 4  
 5     File tempDir = new File(tempDirPath);  
 6     if (!tempDir.exists()) {  
 7         if (!tempDir.mkdirs()) {  
 8             System.err.println("Gagal membuat direktori: " + tempDirPath);  
 9             return maxThreshold / 2;  
10        }  
11    }  
12  
13    File tempFile = new File(tempPath);  
14  
15    try {  
16        while (minThreshold < maxThreshold && iterations < maxIterations) {  
17            iterations++;  
18            currentThreshold = (minThreshold + maxThreshold) / 2;  
19  
20            QuadtreeNode root = buildQuadtree(originalImage, 0, 0,  
21                originalImage.getWidth(), originalImage.getHeight(), 0,  
currentThreshold, false);  
22  
23            ImageMatrix reconstructedImage = IO.reconstructImageFromQuadtree(root,  
24                originalImage.getWidth(), originalImage.getHeight());  
25  
26            IO.writeCompressedImage(reconstructedImage, tempPath);  
27  
28            currentCompression = IO.calcCompressionPercentage(inputPath, tempPath);  
29  
30            if (Math.abs(currentCompression - targetCompressionPercentage) < epsilon)  
{  
31                if (tempFile.exists()) {  
32                    tempFile.delete();  
33                }  
34                return currentThreshold;  
35            }  
36  
37            if (currentCompression > targetCompressionPercentage) {  
38                maxThreshold = currentThreshold;  
39            } else {  
40                minThreshold = currentThreshold;  
41            }  
42        }  
43  
44        double finalThreshold = (minThreshold + maxThreshold) / 2;  
45  
46        if (tempFile.exists()) {  
47            tempFile.delete();  
48        }  
49  
50        return finalThreshold;  
51    }  
52}
```

```

1 } catch (Exception e) {
2     System.err.println("Error during threshold optimization: " + e.getMessage());
3     e.printStackTrace();
4
5     if (tempFile.exists()) {
6         tempFile.delete();
7     }
8
9     return (minThreshold + maxThreshold) / 2;
10 } finally {
11     if (tempFile.exists()) {
12         tempFile.delete();
13     }
14 }
15 }
16
17 public int calculateTreeDepth(QuadtreeNode root) {
18     if (root == null) {
19         return -1;
20     }
21
22     if (root.getChildren() == null) {
23         return 0;
24     }
25
26     int topLeftDepth = calculateTreeDepth(root.getChildren()[0]);
27     int topRightDepth = calculateTreeDepth(root.getChildren()[1]);
28     int bottomLeftDepth = calculateTreeDepth(root.getChildren()[2]);
29     int bottomRightDepth = calculateTreeDepth(root.getChildren()[3]);
30
31     int maxDepth = topLeftDepth;
32     if (topRightDepth > maxDepth)
33         maxDepth = topRightDepth;
34     if (bottomLeftDepth > maxDepth)
35         maxDepth = bottomLeftDepth;
36     if (bottomRightDepth > maxDepth)
37         maxDepth = bottomRightDepth;
38
39     return maxDepth + 1;
40 }
41
42 public int calculateNodeCount(QuadtreeNode root) {
43     if (root == null) {
44         return 0;
45     }
46
47     int count = 1;
48
49     if (root.getChildren() != null) {
50         int topLeftCount = calculateNodeCount(root.getChildren()[0]);
51         int topRightCount = calculateNodeCount(root.getChildren()[1]);
52         int bottomLeftCount = calculateNodeCount(root.getChildren()[2]);
53         int bottomRightCount = calculateNodeCount(root.getChildren()[3]);
54
55         count += topLeftCount + topRightCount + bottomLeftCount + bottomRightCount;
56     }
57
58     return count;
59 }
60
61 public double getThreshold() {
62     return threshold;
63 }
64 }
```

3.3.5. Class ErrorMetrics

```
 1 public class ErrorMetrics {
 2     private static double calculateVariance(ImageMatrix image, int x, int y, int width,
 3     int height) {
 4         double totalVariance = 0.0;
 5         for (int channel = 0; channel < 3; channel++) {
 6             totalVariance += calculateChannelVariance(image, x, y, width, height,
 7             channel);
 8         }
 9     }
10
11     private static double calculateChannelVariance(ImageMatrix image, int x, int y, int
12     width, int height, int channel) {
13         double mean = calculateMean(image, x, y, width, height, channel);
14         double channelVariance = 0.0;
15
16         for (int i = 0; i < width; i++) {
17             for (int j = 0; j < height; j++) {
18                 int pixelValue = image.getPixel(channel, x + i, y + j);
19                 channelVariance += Math.pow(pixelValue - mean, 2);
20             }
21         }
22
23         return channelVariance / (width * height);
24     }
25
26     private static double calculateMean(ImageMatrix image, int x, int y, int width, int
27     height, int channel) {
28         double sum = 0.0;
29         int pixelCount = width * height;
30
31         for (int i = 0; i < width; i++) {
32             for (int j = 0; j < height; j++) {
33                 sum += image.getPixel(channel, x + i, y + j);
34             }
35         }
36
37         return sum / pixelCount;
38     }
39
40     private static double calculateMeanAbsoluteDeviation(ImageMatrix image, int x, int y,
41     int width, int height) {
42         double totalMAD = 0.0;
43         for (int channel = 0; channel < 3; channel++) {
44             double mean = calculateMean(image, x, y, width, height, channel);
45             double channelMAD = 0.0;
46
47             for (int i = 0; i < width; i++) {
48                 for (int j = 0; j < height; j++) {
49                     int pixelValue = image.getPixel(channel, x + i, y + j);
50                     channelMAD += Math.abs(pixelValue - mean);
51                 }
52             }
53
54             totalMAD += channelMAD / (width * height);
55         }
56
57         return totalMAD / 3.0;
58     }
59 }
```

```
 1     private static double calculateMaxPixelDifference(ImageMatrix image, int x, int y, int
 2         width, int height) {
 3         double totalMaxDiff = 0.0;
 4         for (int channel = 0; channel < 3; channel++) {
 5             int maxValue = 0;
 6             int minValue = 255;
 7
 8             for (int i = 0; i < width; i++) {
 9                 for (int j = 0; j < height; j++) {
10                     int pixelValue = image.getPixel(channel, x + i, y + j);
11
12                     if (pixelValue > maxValue) {
13                         maxValue = pixelValue;
14                     }
15                     if (pixelValue < minValue) {
16                         minValue = pixelValue;
17                     }
18                 }
19             }
20
21             totalMaxDiff += (maxValue - minValue);
22         }
23
24         return totalMaxDiff / 3.0;
25     }
26
27     private static double calculateEntropy(ImageMatrix image, int x, int y, int width, int
28         height) {
29         double totalEntropy = 0.0;
30
31         for (int channel = 0; channel < 3; channel++) {
32             int[] histogram = new int[256];
33
34             for (int i = 0; i < width; i++) {
35                 for (int j = 0; j < height; j++) {
36                     int pixelValue = image.getPixel(channel, x + i, y + j);
37                     histogram[pixelValue]++;
38                 }
39             }
40
41             double channelEntropy = 0.0;
42             int totalPixels = width * height;
43
44             for (int count : histogram) {
45                 if (count > 0) {
46                     double probability = (double) count / totalPixels;
47                     channelEntropy -= probability * (Math.log(probability) /
48                         Math.log(2));
49                 }
50             }
51             totalEntropy += channelEntropy;
52         }
53
54         return totalEntropy / 3.0;
55     }

```

```
 1     public static double calculateSSIM(ImageMatrix original, ImageMatrix compressed, int
 2         x, int y, int width, int height) {
 3         double weightR = 0.299;
 4         double weightG = 0.587;
 5         double weightB = 0.114;
 6         double ssimR = calculateChannelSSIM(original, compressed, x, y, width, height, 0);
 7
 8         double ssimG = calculateChannelSSIM(original, compressed, x, y, width, height, 1);
 9
10        double ssimB = calculateChannelSSIM(original, compressed, x, y, width, height, 2);
11
12        return weightR * ssimR + weightG * ssimG + weightB * ssimB;
13    }
14
15    private static double calculateChannelSSIM(ImageMatrix original, ImageMatrix
16        compressed, int x, int y, int width, int height, int channel) {
17        double meanOriginal = calculateMean(original, x, y, width, height, channel);
18        double meanCompressed = calculateMean(compressed, x, y, width, height, channel);
19        double varianceOriginal = calculateChannelVariance(original, x, y, width, height,
20            channel);
21        double varianceCompressed = calculateChannelVariance(compressed, x, y, width,
22            height, channel);
23
24        double C1 = 6.5025; // (0.01 * 255)^2
25        double C2 = 58.5225; // (0.03 * 255)^2
26
27        double ssimC = ((2 * meanOriginal * meanCompressed + C1)
28                         * (2 * calculateCovariance(original, compressed, x, y, width, height,
29                           channel) + C2))
30                         / ((meanOriginal * meanOriginal + meanCompressed * meanCompressed + C1)
31                             * (varianceOriginal + varianceCompressed + C2));
32
33        return ssimC;
34    }
35
36    private static double calculateCovariance(ImageMatrix img1, ImageMatrix img2, int x,
37        int y, int width, int height, int channel) {
38        double meanImg1 = calculateMean(img1, x, y, width, height, channel);
39        double meanImg2 = calculateMean(img2, x, y, width, height, channel);
40
41        double covariance = 0.0;
42        int pixelCount = width * height;
43
44        for (int i = 0; i < width; i++) {
45            for (int j = 0; j < height; j++) {
46                double pixel1 = img1.getPixel(channel, x + i, y + j);
47                double pixel2 = img2.getPixel(channel, x + i, y + j);
48
49                covariance += (pixel1 - meanImg1) * (pixel2 - meanImg2);
50            }
51        }
52
53        return covariance / pixelCount;
54    }
55}
```

```
 1  public static double calculateError(ImageMatrix original, int x, int y, int width, int
 2      height, int errorMethod) {
 3          // 1: Variance
 4          // 2: Mean Absolute Deviation
 5          // 3: Max Pixel Difference
 6          // 4: Entropy
 7          // 5: SSIM (Not Fixed)
 8
 9          double error = 0.0;
10         if (errorMethod == 1){
11             error = calculateVariance(original, x, y, width, height);
12         } else if (errorMethod == 2){
13             error = calculateMeanAbsoluteDeviation(original, x, y, width, height);
14         } else if (errorMethod == 3){
15             error = calculateMaxPixelDifference(original, x, y, width, height);
16         } else if (errorMethod == 4){
17             error = calculateEntropy(original, x, y, width, height);
18         } else if (errorMethod == 5){
19             error = 1.0 - calculateBlockSSIM(original, x, y, width, height);
20         } else {
21             throw new IllegalArgumentException("Invalid error method");
22         }
23
24         return error;
25     }
26
27     public static double calculateBlockSSIM(ImageMatrix original, int x, int y, int width,
28         int height) {
29         ImageMatrix compressedBlock = new ImageMatrix(width, height);
30         int avgColor = original.getAverageColor(x, y, width, height);
31         int avgR = (avgColor >> 16) & 0xFF;
32         int avgG = (avgColor >> 8) & 0xFF;
33         int avgB = avgColor & 0xFF;
34
35         for (int i = 0; i < width; i++) {
36             for (int j = 0; j < height; j++) {
37                 compressedBlock.setPixel(0, i, j, avgR);
38                 compressedBlock.setPixel(1, i, j, avgG);
39                 compressedBlock.setPixel(2, i, j, avgB);
40             }
41         }
42         return calculateSSIM(original, compressedBlock, 0, 0, width, height);
43     }
44 }
```

3.3.6. Class ImageMatrix

```
 1  public class ImageMatrix {
 2      private int[][][] pixels;
 3      private int width;
 4      private int height;
 5
 6      public ImageMatrix(BufferedImage image) {
 7          this.width = image.getWidth();
 8          this.height = image.getHeight();
 9          this.pixels = new int[3][width][height];
10
11         for (int x = 0; x < width; x++) {
12             for (int y = 0; y < height; y++) {
13                 int rgb = image.getRGB(x, y);
14                 pixels[0][x][y] = (rgb >> 16) & 0xFF;
15                 pixels[1][x][y] = (rgb >> 8) & 0xFF;
16                 pixels[2][x][y] = rgb & 0xFF;
17             }
18         }
19     }
20
21     public ImageMatrix(int width, int height) {
22         this.width = width;
23         this.height = height;
24         this.pixels = new int[3][width][height];
25     }
26
27     public BufferedImage toBufferedImage() {
28         BufferedImage image = new BufferedImage(width, height,
29             BufferedImage.TYPE_INT_RGB);
30
31         for (int x = 0; x < width; x++) {
32             for (int y = 0; y < height; y++) {
33                 int r = pixels[0][x][y];
34                 int g = pixels[1][x][y];
35                 int b = pixels[2][x][y];
36                 int rgb = (r << 16) | (g << 8) | b;
37                 image.setRGB(x, y, rgb);
38             }
39         }
40
41         return image;
42     }
}
```

```
● ● ●

1  public int[][][] getPixels() {
2      return pixels;
3  }
4
5  public int getWidth() {
6      return width;
7  }
8
9  public int getHeight() {
10     return height;
11 }
12
13 public int getPixel(int channel, int x, int y) {
14     return pixels[channel][x][y];
15 }
16
17 public void setPixel(int channel, int x, int y, int value) {
18     pixels[channel][x][y] = value;
19 }
20
21 public int getAverageColor(int x, int y, int blockWidth, int blockHeight) {
22     long sumR = 0, sumG = 0, sumB = 0;
23     int pixelCount = blockWidth * blockHeight;
24
25     for (int i = 0; i < blockWidth; i++) {
26         for (int j = 0; j < blockHeight; j++) {
27             sumR += pixels[0][x + i][y + j];
28             sumG += pixels[1][x + i][y + j];
29             sumB += pixels[2][x + i][y + j];
30         }
31     }
32
33     int avgR = (int) (sumR / pixelCount);
34     int avgG = (int) (sumG / pixelCount);
35     int avgB = (int) (sumB / pixelCount);
36
37     return (avgR << 16) | (avgG << 8) | avgB;
38 }
39 }
```

3.3.7. Class IO

```
1 public class IO {
2     public static ImageMatrix readImage(String imagePath) throws IOException {
3         BufferedImage bufferedImage = ImageIO.read(new File(imagePath));
4         return new ImageMatrix(bufferedImage);
5     }
6
7     public static long calcFileSize(String imagePath) {
8         File file = new File(imagePath);
9         return file.length();
10    }
11
12    public static double calcCompressionPercentage(String originalPath, String
13 compressedPath) {
14        long originalSize = calcFileSize(originalPath);
15        long compressedSize = calcFileSize(compressedPath);
16
17        if (originalSize == 0) {
18            throw new IllegalArgumentException("[ERROR] : Original file size cannot be
19 zero.");
20        }
21
22        double compressionRatio = 1.0 - ((double) compressedSize / originalSize);
23        return compressionRatio * 100;
24    }
25
26    // Output Image
27    public static ImageMatrix reconstructImageFromQuadtree(QuadtreeNode root, int width,
28 int height) {
29        ImageMatrix reconstructedImage = new ImageMatrix(width, height);
30        reconstructNode(root, reconstructedImage);
31        return reconstructedImage;
32    }
33
34    private static void reconstructNode(QuadtreeNode node, ImageMatrix image) {
35        if (node.isLeaf()) {
36            int avgColor = node.getAverageColor();
37            for (int x = node.getX(); x < node.getX() + node.getWidth(); x++) {
38                for (int y = node.getY(); y < node.getY() + node.getHeight(); y++) {
39                    int r = (avgColor >> 16) & 0xFF;
40                    int g = (avgColor >> 8) & 0xFF;
41                    int b = avgColor & 0xFF;
42
43                    image.setPixel(0, x, y, r);
44                    image.setPixel(1, x, y, g);
45                    image.setPixel(2, x, y, b);
46                }
47            }
48        } else {
49            // Top-left
50            reconstructNode(node.getChildren()[0], image);
51
52            // Top-right
53            reconstructNode(node.getChildren()[1], image);
54
55            // Bottom-left
56            reconstructNode(node.getChildren()[2], image);
57
58        }
59    }
60}
```

```

1      public static void writeCompressedImage(ImageMatrix image, String outputPath) throws
2          IOException {
3              BufferedImage bufferedImage = image.toBufferedImage();
4              File outputFile = new File(outputPath);
5              String format = outputPath.substring(outputPath.lastIndexOf('.') + 1);
6              ImageIO.write(bufferedImage, format, outputFile);
7          }
8
9      // Output GIF
10     public static List<BufferedImage> reconstructGIFFromQuadtree(QuadtreeNode root, int
11         width, int height, int maxDepth) {
12         List<BufferedImage> frames = new ArrayList<>();
13
14         for (int i = 0; i <= maxDepth; i++) {
15             ImageMatrix frameImage = new ImageMatrix(width, height);
16             IO.reconstructNodeForGIF(root, frameImage, 0, i);
17             frames.add(frameImage.toBufferedImage());
18         }
19
20         return frames;
21     }
22
23     public static void reconstructNodeForGIF(QuadtreeNode node, ImageMatrix image, int
24         currentDepth, int maxDepth) {
25         if (node.isLeaf() || currentDepth >= maxDepth) {
26             int avgColor = node.getAverageColor();
27             for (int x = node.getX(); x < node.getX() + node.getWidth(); x++) {
28                 for (int y = node.getY(); y < node.getY() + node.getHeight(); y++) {
29                     int r = (avgColor >> 16) & 0xFF;
30                     int g = (avgColor >> 8) & 0xFF;
31                     int b = avgColor & 0xFF;
32
33                     image.setPixel(0, x, y, r);
34                     image.setPixel(1, x, y, g);
35                     image.setPixel(2, x, y, b);
36                 }
37             }
38         } else {
39             // Top-left
40             reconstructNodeForGIF(node.getChildren()[0], image, currentDepth + 1,
41             maxDepth);
42
43             // Top-right
44             reconstructNodeForGIF(node.getChildren()[1], image, currentDepth + 1,
45             maxDepth);
46
47             // Bottom-left
48             reconstructNodeForGIF(node.getChildren()[2], image, currentDepth + 1,
49             maxDepth);
50         }
51     }

```

```
 1  public static void createCompressionGif(List<BufferedImage> frames, String outputPath,
 2      int delayTime) throws IOException {
 3      ImageWriter gifWriter = ImageIO.getImageWritersBySuffix("gif").next();
 4      ImageWriteParam imageWriteParam = gifWriter.getDefaultWriteParam();
 5      ImageOutputStream output = ImageIO.createImageOutputStream(new File(outputPath));
 6      gifWriter.setOutput(output);
 7
 8      gifWriter.prepareWriteSequence(null);
 9
10     for (int i = 0; i < frames.size(); i++) {
11         BufferedImage frame = frames.get(i);
12         IIOMetadata metadata = gifWriter.getDefaultImageMetadata(new
13             ImageTypeSpecifier(frame), imageWriteParam);
14         String metaFormat = metadata.getNativeMetadataFormatName();
15         IIOMetadataNode root = (IIOMetadataNode) metadata.getAsTree(metaFormat);
16         IIOMetadataNode graphicsControlExtensionNode = new
17             IIOMetadataNode("GraphicControlExtension");
18         graphicsControlExtensionNode.setAttribute("disposalMethod", "none");
19         graphicsControlExtensionNode.setAttribute("userInputFlag", "FALSE");
20         graphicsControlExtensionNode.setAttribute("transparentColorFlag", "FALSE");
21         graphicsControlExtensionNode.setAttribute("delayTime",
22             String.valueOf(delayTime / 10));
23         graphicsControlExtensionNode.setAttribute("transparentColorIndex", "0");
24         root.appendChild(graphicsControlExtensionNode);
25
26         // Loop forever
27         if (i == 0) {
28             IIOMetadataNode appExtensions = new
29                 IIOMetadataNode("ApplicationExtensions");
30             IIOMetadataNode appExtension = new
31                 IIOMetadataNode("ApplicationExtension");
32             appExtension.setAttribute("applicationID", "NETSCAPE");
33             appExtension.setAttribute("authenticationCode", "2.0");
34
35             byte[] loopForever = new byte[] { 0x1, 0x0, 0x0 };
36             appExtension.setUserObject(loopForever);
37             appExtensions.appendChild(appExtension);
38             root.appendChild(appExtensions);
39         }
40
41         gifWriter.endWriteSequence();
42         output.close();
43     }
44 }
```

```
● ● ●

1 // Validation IO
2 public static boolean isValidImagePath(String path) {
3     String ext = getExtension(path).toLowerCase();
4     if (!ext.equals("jpg") && !ext.equals("jpeg") && !ext.equals("png")) {
5         System.out.println(" ✗ Invalid image format! Allowed: .jpg, .jpeg, .png");
6         return false;
7     }
8
9     File file = new File(path);
10    if (!file.exists() || !file.isFile()) {
11        System.out.println(" ✗ File does not exist.");
12        return false;
13    }
14
15    return true;
16 }
17
18
19 private static String getExtension(String path) {
20     int i = path.lastIndexOf('.');
21     return (i > 0) ? path.substring(i + 1).toLowerCase() : "";
22 }
23
24 public static boolean hasSameExtension(String path1, String path2) {
25     return getExtension(path1).equals(getExtension(path2));
26 }
27
28 public static boolean isValidThreshold(double value, int method) {
29     if (method == 1) {
30         return value >= 0 && value <= 16256.25;
31     } else if (method == 2) {
32         return value >= 0 && value <= 127.5;
33     } else if (method == 3) {
34         return value >= 0 && value <= 255;
35     } else if (method == 4) {
36         return value >= 0 && value <= 8;
37     } else if (method == 5) {
38         return value >= 0 && value <= 1;
39     } else {
40         return false;
41     }
42 }
43 }
```

3.3.8. Class UI

```
1     public static void printProgressBar(int percent) {
2         final int width = 40;
3         int completed = width * percent / 100;
4         String[] spinChars = new String[] { ":", ":", ":", ":", ".:", ".:", ".:", ".:" ,
5             ":", ":" };
6         String spinChar = spinChars[(percent / 5) % spinChars.length];
7
8         StringBuilder bar = new StringBuilder();
9         bar.append("\r").append(BLUE).append(spinChar).append(" ").append(WHITE).append([
10
11         for (int i = 0; i < width; i++) {
12             if (i < completed) {
13                 bar.append(GREEN).append("█").append(RESET);
14             } else {
15                 bar.append(" ");
16             }
17
18             bar.append(WHITE).append("] ").append(CYAN).append(String.format("%3d%", percent)).append(RESET);
19             System.out.print(bar);
20             System.out.flush();
21     }
```

```
1  public static void printErrorMethodInfo() {
2      System.out.println("\n" + CYAN+ "_____"
3          System.out.println(CYAN + "||" + WHITE + "■" + CYAN + " ERROR MEASUREMENT METRICS")
4          System.out.println(CYAN + "||" + WHITE + "■" + CYAN + "_____"
5              System.out.println(CYAN + "||" + BLUE + "1. " + RESET + "Variance" + CYAN + "||" + RESET);
6              System.out.println(CYAN + "||" + BLUE + "2. " + RESET + "Mean Absolute Deviation" + CYAN + "||" + RESET);
7              System.out.println(CYAN + "||" + BLUE + "3. " + RESET + "Max Pixel Difference" + CYAN + "||" + RESET);
8              System.out.println(CYAN + "||" + BLUE + "4. " + RESET + "Entropy" + CYAN + "||" + RESET);
9              System.out.println(CYAN + "||" + BLUE + "5. " + RESET + "Structural Similarity Index" + CYAN + "||" + RESET);
10             System.out.println(CYAN + "||" + CYAN + "_____"
11         }
12
13     public static void printResultMenu(int treeDepth, int nodeCount, double executionTime, double compressionTarget, String inputPath, String outputPath, String gifPath, boolean saveGif, double compressionPercentage) {
14         String compressionColor = compressionPercentage > 50 ? GREEN : (compressionPercentage <= 50) ? PURPLE : YELLOW;
15
16         System.out.println("\n\n" + PURPLE + "_____"
17             System.out.println(PURPLE + "||" + WHITE + " COMPRESSION RESULTS"
18             System.out.println(PURPLE + "||" + WHITE + "_____"
19                 System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + CYAN + " %-18s" + PURPLE + "||" + RESET, "Execution Time", String.format("%.4f ms", executionTime));
20                 System.out.println(PURPLE + "||" + WHITE + "_____"
21                     System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + YELLOW + " %-18s" + PURPLE + "||" + RESET, "Original Size", String.format("%d bytes", IO.calcFileSize(inputPath)));
22                     System.out.println(PURPLE + "||" + WHITE + "_____"
23                         System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + GREEN + " %-18s" + PURPLE + "||" + RESET, "Compressed Size", String.format("%d bytes", IO.calcFileSize(outputPath)));
24                         System.out.println(PURPLE + "||" + WHITE + "_____"
25                             System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + compressionColor + " %-18s" + PURPLE + "||" + RESET, "Compression Rate", String.format("%.4f%", compressionPercentage));
26                             System.out.println(PURPLE + "||" + WHITE + "_____"
27                                 System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + BLUE + " %-18s" + PURPLE + "||" + RESET, "Tree Depth", String.format("%d", treeDepth));
28                                 System.out.println(PURPLE + "||" + WHITE + "_____"
29                                     System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + BLUE + " %-18s" + PURPLE + "||" + RESET, "Total Nodes", String.format("%d", nodeCount));
30
31         if (compressionTarget != 0){
32             System.out.println(PURPLE + "||" + WHITE + "_____"
33                 System.out.printf(PURPLE + "||" + RESET + " %-22s" + PURPLE + "||" + BLUE + " %-18s" + PURPLE + "||" + RESET, "Optimal Threshold", String.format("%.4f", threshold));
34             }
35
36             System.out.println(PURPLE + "||" + WHITE + "_____"
37                 System.out.println(WHITE + "\n ─ Output Files:" + RESET);
38                 System.out.println(GREEN + " " + checkedIcon + " Compressed Image: " + RESET + outputPath);
39                 if (saveGif && gifPath != null) {
40                     System.out.println(GREEN + " " + checkedIcon + " Compression GIF: " + RESET + gifPath);
41                 }
42             }
43
44 }
```

```
● ● ●

1  public static String getThresholdRange(int method) {
2      switch (method) {
3          case 1:
4              return "(0-16256.25)";
5          case 2:
6              return "(0-127.5)";
7          case 3:
8              return "(0-255)";
9          case 4:
10             return "(0.0-8.0)";
11         case 5:
12             return "(0.0-1.0)";
13         default:
14             return "";
15     }
16 }
17
18 public static String getMethodName(int method) {
19     switch (method) {
20         case 1:
21             return "Variance";
22         case 2:
23             return "Mean Absolute Deviation";
24         case 3:
25             return "Max Pixel Difference";
26         case 4:
27             return "Entropy";
28         case 5:
29             return "SSIM";
30         default:
31             return "Unknown";
32     }
33 }
34 }
```

BAB IV

HASIL PENGUJIAN DAN ANALISIS

4.1 Hasil Pengujian

4.1.1 Kasus Eksperimen 1

Gambar Asli :

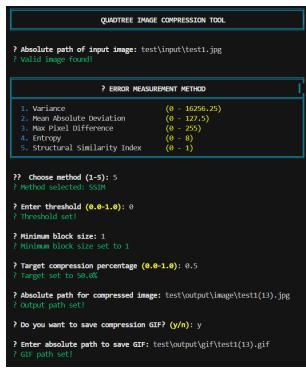
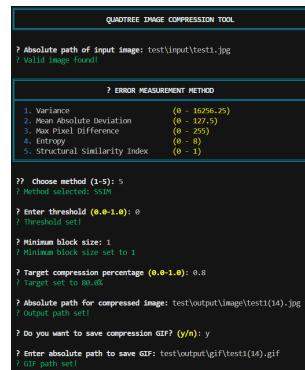
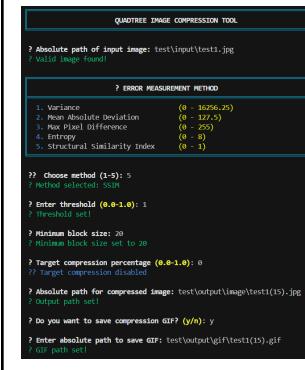
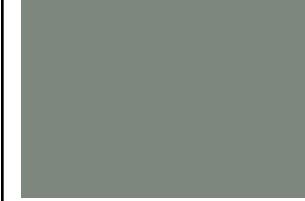


Hasil Kompresi :

Metode	Ukuran Blok Minimum : 1	Ukuran Blok Minimum : 1	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <pre>? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</pre> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(1).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(1).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <pre>? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</pre> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(2).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(2).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <pre>? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</pre> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 3000 ? Threshold set!</pre> <pre>? Minimum block size: 20 ? Minimum block size set to 20</pre> <pre>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled</pre> <pre>? Absolute path for compressed image: test\output\image\test1(3).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): test\output\gif\test1(3).gif ? Please type 'y' or 'n': y ? Enter absolute path to save GIF: test\output\gif\test1(3).gif ? GIF path set!</pre>
Variansi			

Nilai Ambang Batas :	0	0	1000																																																
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>1981,2501 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>97369 bytes</td></tr> <tr> <td>Compression Rate</td><td>49,9992%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>49525</td></tr> <tr> <td>Optimal Threshold</td><td>2445,7798</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(1).jpg ? Compression GIF: test\output\gif\test1(1).gif</pre>	COMPRESSION RESULTS		Execution Time	1981,2501 ms	Original Size	194735 bytes	Compressed Size	97369 bytes	Compression Rate	49,9992%	Tree Depth	10	Total Nodes	49525	Optimal Threshold	2445,7798	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>3322,7386 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>39033 bytes</td></tr> <tr> <td>Compression Rate</td><td>79,9558%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>4713</td></tr> <tr> <td>Optimal Threshold</td><td>3675,4486</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(2).jpg ? Compression GIF: test\output\gif\test1(2).gif</pre>	COMPRESSION RESULTS		Execution Time	3322,7386 ms	Original Size	194735 bytes	Compressed Size	39033 bytes	Compression Rate	79,9558%	Tree Depth	10	Total Nodes	4713	Optimal Threshold	3675,4486	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>1085,6454 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>88356 bytes</td></tr> <tr> <td>Compression Rate</td><td>54,6276%</td></tr> <tr> <td>Tree Depth</td><td>7</td></tr> <tr> <td>Total Nodes</td><td>12525</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(3).jpg ? Compression GIF: test\output\gif\test1(3).gif</pre>	COMPRESSION RESULTS		Execution Time	1085,6454 ms	Original Size	194735 bytes	Compressed Size	88356 bytes	Compression Rate	54,6276%	Tree Depth	7	Total Nodes	12525		
COMPRESSION RESULTS																																																			
Execution Time	1981,2501 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	97369 bytes																																																		
Compression Rate	49,9992%																																																		
Tree Depth	10																																																		
Total Nodes	49525																																																		
Optimal Threshold	2445,7798																																																		
COMPRESSION RESULTS																																																			
Execution Time	3322,7386 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	39033 bytes																																																		
Compression Rate	79,9558%																																																		
Tree Depth	10																																																		
Total Nodes	4713																																																		
Optimal Threshold	3675,4486																																																		
COMPRESSION RESULTS																																																			
Execution Time	1085,6454 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	88356 bytes																																																		
Compression Rate	54,6276%																																																		
Tree Depth	7																																																		
Total Nodes	12525																																																		
Input :	<table border="1"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image:</td><td>test\input\test1.jpg</td></tr> <tr> <td>? Valid image found!</td><td></td></tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td></tr> <tr> <td>1. Variance</td><td>(0 - 16256,25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127,5): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0% ? Absolute path for compressed image: test\output\image\test1(4).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(4).gif ? GIF path set!</pre>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image:	test\input\test1.jpg	? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<table border="1"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image:</td><td>test\input\test1.jpg</td></tr> <tr> <td>? Valid image found!</td><td></td></tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td></tr> <tr> <td>1. Variance</td><td>(0 - 16256,25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127,5): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0% ? Absolute path for compressed image: test\output\image\test1(5).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(5).gif ? GIF path set!</pre>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image:	test\input\test1.jpg	? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td><td>(0 - 16256,25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127,5): 120 ? Threshold set! ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ?? Target compression disabled ? Absolute path for compressed image: test\output\image\test1(6).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(6).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)
QUADTREE IMAGE COMPRESSION TOOL																																																			
? Absolute path of input image:	test\input\test1.jpg																																																		
? Valid image found!																																																			
? ERROR MEASUREMENT METHOD																																																			
1. Variance	(0 - 16256,25)																																																		
2. Mean Absolute Deviation	(0 - 127,5)																																																		
3. Max Pixel Difference	(0 - 255)																																																		
4. Entropy	(0 - 8)																																																		
5. Structural Similarity Index	(0 - 1)																																																		
QUADTREE IMAGE COMPRESSION TOOL																																																			
? Absolute path of input image:	test\input\test1.jpg																																																		
? Valid image found!																																																			
? ERROR MEASUREMENT METHOD																																																			
1. Variance	(0 - 16256,25)																																																		
2. Mean Absolute Deviation	(0 - 127,5)																																																		
3. Max Pixel Difference	(0 - 255)																																																		
4. Entropy	(0 - 8)																																																		
5. Structural Similarity Index	(0 - 1)																																																		
? ERROR MEASUREMENT METHOD																																																			
1. Variance	(0 - 16256,25)																																																		
2. Mean Absolute Deviation	(0 - 127,5)																																																		
3. Max Pixel Difference	(0 - 255)																																																		
4. Entropy	(0 - 8)																																																		
5. Structural Similarity Index	(0 - 1)																																																		
Mean Absolute Deviation																																																			
Nilai Ambang Batas :	0	0	120																																																
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>3937,4134 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>97532 bytes</td></tr> <tr> <td>Compression Rate</td><td>49,9155%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>53809</td></tr> <tr> <td>Optimal Threshold</td><td>39,2669</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(4).jpg ? Compression GIF: test\output\gif\test1(4).gif</pre>	COMPRESSION RESULTS		Execution Time	3937,4134 ms	Original Size	194735 bytes	Compressed Size	97532 bytes	Compression Rate	49,9155%	Tree Depth	10	Total Nodes	53809	Optimal Threshold	39,2669	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>3421,1890 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>38500 bytes</td></tr> <tr> <td>Compression Rate</td><td>80,2295%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>4681</td></tr> <tr> <td>Optimal Threshold</td><td>49,8182</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(5).jpg ? Compression GIF: test\output\gif\test1(5).gif</pre>	COMPRESSION RESULTS		Execution Time	3421,1890 ms	Original Size	194735 bytes	Compressed Size	38500 bytes	Compression Rate	80,2295%	Tree Depth	10	Total Nodes	4681	Optimal Threshold	49,8182	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>338,7278 ms</td></tr> <tr> <td>Original Size</td><td>194735 bytes</td></tr> <tr> <td>Compressed Size</td><td>11635 bytes</td></tr> <tr> <td>Compression Rate</td><td>94,0252%</td></tr> <tr> <td>Tree Depth</td><td>0</td></tr> <tr> <td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(6).jpg ? Compression GIF: test\output\gif\test1(6).gif</pre>	COMPRESSION RESULTS		Execution Time	338,7278 ms	Original Size	194735 bytes	Compressed Size	11635 bytes	Compression Rate	94,0252%	Tree Depth	0	Total Nodes	1		
COMPRESSION RESULTS																																																			
Execution Time	3937,4134 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	97532 bytes																																																		
Compression Rate	49,9155%																																																		
Tree Depth	10																																																		
Total Nodes	53809																																																		
Optimal Threshold	39,2669																																																		
COMPRESSION RESULTS																																																			
Execution Time	3421,1890 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	38500 bytes																																																		
Compression Rate	80,2295%																																																		
Tree Depth	10																																																		
Total Nodes	4681																																																		
Optimal Threshold	49,8182																																																		
COMPRESSION RESULTS																																																			
Execution Time	338,7278 ms																																																		
Original Size	194735 bytes																																																		
Compressed Size	11635 bytes																																																		
Compression Rate	94,0252%																																																		
Tree Depth	0																																																		
Total Nodes	1																																																		

Input :	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(7).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(7).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(8).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(8).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 255 ? Threshold set!</pre> <pre>? Minimum block size: 20 ? Minimum block size set to 20</pre> <pre>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled!</pre> <pre>? Absolute path for compressed image: test\output\image\test1(9).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(9).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)										
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
Max Pixel Difference																																																	
Nilai Ambang Batas :	0	0	255																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>2824.8899 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>39193 bytes</td></tr> <tr><td>Compression Rate</td><td>79.8737%</td></tr> <tr><td>Tree Depth</td><td>9</td></tr> <tr><td>Total Nodes</td><td>1249</td></tr> <tr><td>Optimal Threshold</td><td>253.6667</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(7).jpg ? Compression GIF: test\output\gif\test1(7).gif</pre>	COMPRESSION RESULTS		Execution Time	2824.8899 ms	Original Size	194735 bytes	Compressed Size	39193 bytes	Compression Rate	79.8737%	Tree Depth	9	Total Nodes	1249	Optimal Threshold	253.6667	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>2815.7150 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>39193 bytes</td></tr> <tr><td>Compression Rate</td><td>79.8737%</td></tr> <tr><td>Tree Depth</td><td>9</td></tr> <tr><td>Total Nodes</td><td>1249</td></tr> <tr><td>Optimal Threshold</td><td>253.6667</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(8).jpg ? Compression GIF: test\output\gif\test1(8).gif</pre>	COMPRESSION RESULTS		Execution Time	2815.7150 ms	Original Size	194735 bytes	Compressed Size	39193 bytes	Compression Rate	79.8737%	Tree Depth	9	Total Nodes	1249	Optimal Threshold	253.6667	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>321.1228 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>11635 bytes</td></tr> <tr><td>Compression Rate</td><td>94.0252%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(9).jpg ? Compression GIF: test\output\gif\test1(9).gif</pre>	COMPRESSION RESULTS		Execution Time	321.1228 ms	Original Size	194735 bytes	Compressed Size	11635 bytes	Compression Rate	94.0252%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	2824.8899 ms																																																
Original Size	194735 bytes																																																
Compressed Size	39193 bytes																																																
Compression Rate	79.8737%																																																
Tree Depth	9																																																
Total Nodes	1249																																																
Optimal Threshold	253.6667																																																
COMPRESSION RESULTS																																																	
Execution Time	2815.7150 ms																																																
Original Size	194735 bytes																																																
Compressed Size	39193 bytes																																																
Compression Rate	79.8737%																																																
Tree Depth	9																																																
Total Nodes	1249																																																
Optimal Threshold	253.6667																																																
COMPRESSION RESULTS																																																	
Execution Time	321.1228 ms																																																
Original Size	194735 bytes																																																
Compressed Size	11635 bytes																																																
Compression Rate	94.0252%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Input :	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(10).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Please type 'y' or 'n'. ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(10).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test1(11).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): test\output\gif\test1(11).gif ? Please type 'y' or 'n'. ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(11).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test1.jpg ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 1626..25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 137..5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 8 ? Threshold set!</pre> <pre>? Minimum block size: 20 ? Minimum block size set to 20</pre> <pre>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled!</pre> <pre>? Absolute path for compressed image: test\output\image\test1(12).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test1(12).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626..25)	2. Mean Absolute Deviation	(0 - 137..5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)										
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 1626..25)																																																
2. Mean Absolute Deviation	(0 - 137..5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																

Entropy																																																	
Nilai Ambang Batas :	0	0	8																																														
Output :	<table border="1" data-bbox="470 599 747 825"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>5677.6536 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>97357 bytes</td></tr> <tr><td>Compression Rate</td><td>50.0054%</td></tr> <tr><td>Tree Depth</td><td>8</td></tr> <tr><td>Total Nodes</td><td>17757</td></tr> <tr><td>Optimal Threshold</td><td>5.2881</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(10).jpg ? Compression GIF: test\output\gif\test1(10).gif</pre>	COMPRESSION RESULTS		Execution Time	5677.6536 ms	Original Size	194735 bytes	Compressed Size	97357 bytes	Compression Rate	50.0054%	Tree Depth	8	Total Nodes	17757	Optimal Threshold	5.2881	<table border="1" data-bbox="801 599 1078 825"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>3026.4432 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>38978 bytes</td></tr> <tr><td>Compression Rate</td><td>79.9841%</td></tr> <tr><td>Tree Depth</td><td>6</td></tr> <tr><td>Total Nodes</td><td>989</td></tr> <tr><td>Optimal Threshold</td><td>7.1731</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(11).jpg ? Compression GIF: test\output\gif\test1(11).gif</pre>	COMPRESSION RESULTS		Execution Time	3026.4432 ms	Original Size	194735 bytes	Compressed Size	38978 bytes	Compression Rate	79.9841%	Tree Depth	6	Total Nodes	989	Optimal Threshold	7.1731	<table border="1" data-bbox="1127 599 1405 825"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>325.4680 ms</td></tr> <tr><td>Original Size</td><td>194735 bytes</td></tr> <tr><td>Compressed Size</td><td>11635 bytes</td></tr> <tr><td>Compression Rate</td><td>94.0252%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test1(12).jpg ? Compression GIF: test\output\gif\test1(12).gif</pre>	COMPRESSION RESULTS		Execution Time	325.4680 ms	Original Size	194735 bytes	Compressed Size	11635 bytes	Compression Rate	94.0252%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	5677.6536 ms																																																
Original Size	194735 bytes																																																
Compressed Size	97357 bytes																																																
Compression Rate	50.0054%																																																
Tree Depth	8																																																
Total Nodes	17757																																																
Optimal Threshold	5.2881																																																
COMPRESSION RESULTS																																																	
Execution Time	3026.4432 ms																																																
Original Size	194735 bytes																																																
Compressed Size	38978 bytes																																																
Compression Rate	79.9841%																																																
Tree Depth	6																																																
Total Nodes	989																																																
Optimal Threshold	7.1731																																																
COMPRESSION RESULTS																																																	
Execution Time	325.4680 ms																																																
Original Size	194735 bytes																																																
Compressed Size	11635 bytes																																																
Compression Rate	94.0252%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Input :																																																	
Structural Similarity Index																																																	
Nilai Ambang Batas :	0	0	1																																														

Output :	COMPRESSION RESULTS	COMPRESSION RESULTS	COMPRESSION RESULTS
	Execution Time 10889,1044 ms Original Size 194735 bytes Compressed Size 97409 bytes Compression Rate 49,9787% Tree Depth 10 Total Nodes 43789 Optimal Threshold 0.7032	Execution time 6797,0638 ms Original Size 194735 bytes Compressed Size 38004 bytes Compression Rate 80,4842% Tree Depth 6 Total Nodes 1081 Optimal Threshold 0.9873	Execution Time 372,5254 ms Original Size 194735 bytes Compressed Size 11635 bytes Compression Rate 94,8252% Tree Depth 0 Total Nodes 1
	? Output Files: ? Compressed Image: test\output\image\test1(13).jpg ? Compression GIF: test\output\gif\test1(13).gif	? Output Files: ? Compressed Image: test\output\image\test1(14).jpg ? Compression GIF: test\output\gif\test1(14).gif	? Output Files: ? Compressed Image: test\output\image\test1(15).jpg ? Compression GIF: test\output\gif\test1(15).gif

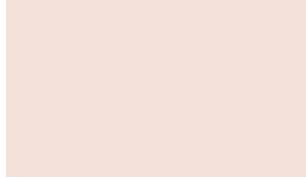
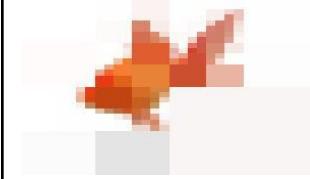
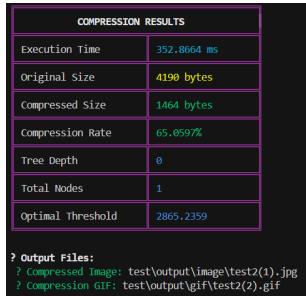
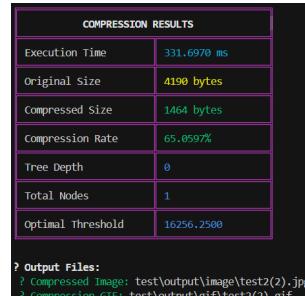
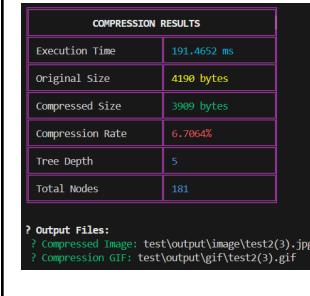
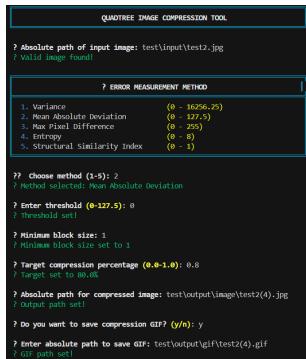
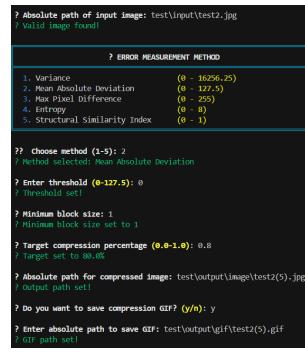
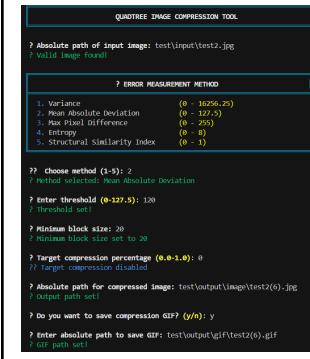
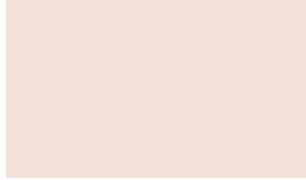
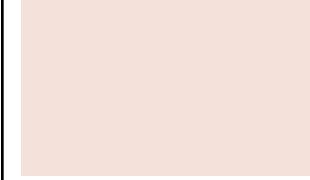
4.1.2 Kasus Eksperimen 2

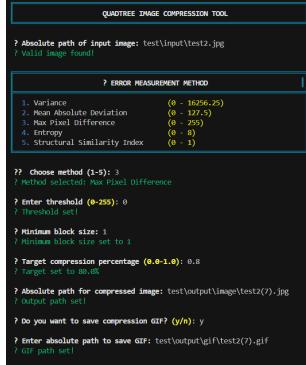
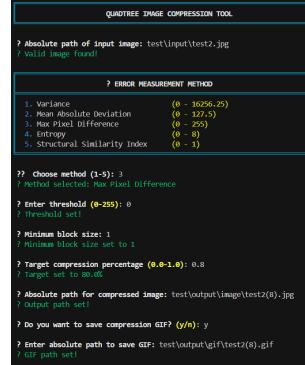
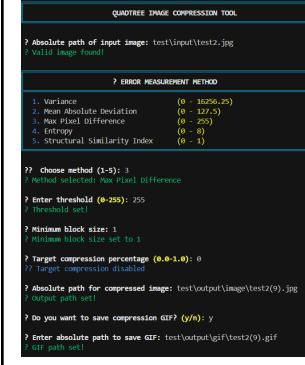
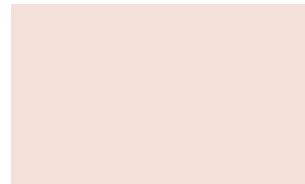
Gambar Asli :

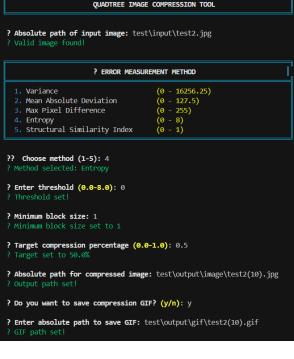
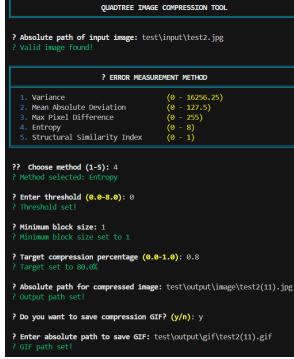
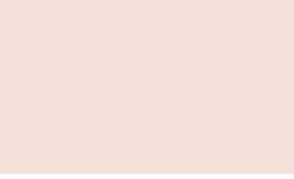
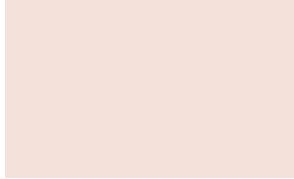
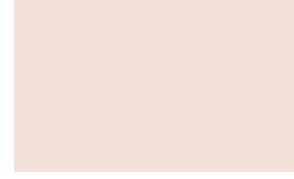
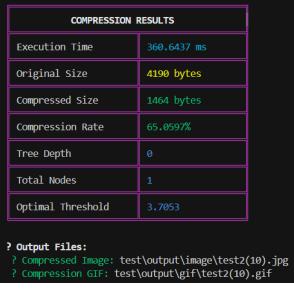
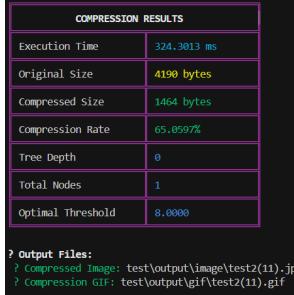
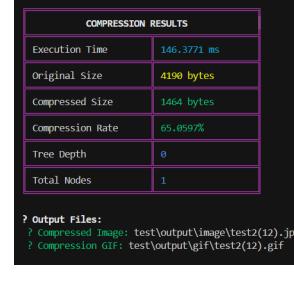
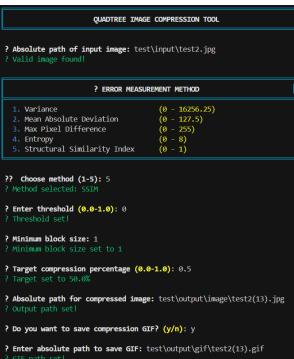
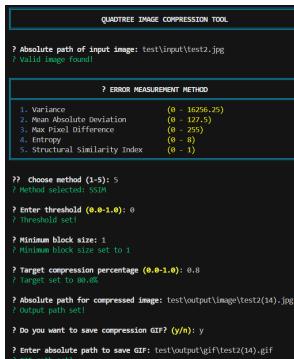
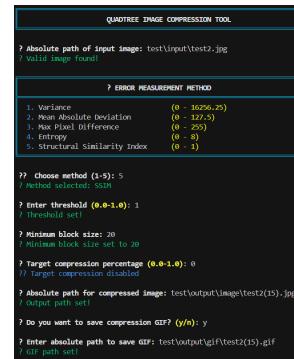


Hasil Kompresi :

Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test2.jpg ? Valid image found! ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256,25) 2. Mean Absolute Deviation (0 - 327,5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256,25): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.00 ? Absolute path for compressed image: test\output\image\test2(1).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test2(1).gif ? GIF path cannot be the same as input or output path. ? Enter absolute path to save GIF: test\output\gif\test2(1).gif ? GIF path set!</pre> <pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test2.jpg ? Valid image found! ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256,25) 2. Mean Absolute Deviation (0 - 327,5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256,25): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.00 ? Absolute path for compressed image: test\output\image\test2(2).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test2(2).gif ? GIF file already exists: test\output\gif\test2(2).gif ? Enter absolute path to save GIF: test\output\gif\test2(2).gif ? GIF path set!</pre> <pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test2.jpg ? Valid image found! ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256,25) 2. Mean Absolute Deviation (0 - 327,5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256,25): 1000 ? Threshold set! ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ?? Target compression disabled ? Absolute path for compressed image: test\output\image\test2(3).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test2(3).gif ? GIF path set!</pre>		

Variansi			
Nilai Ambang Batas :	0	0	1000
Output :			
Input :			
Mean Absolute Deviation			
Nilai Ambang Batas :	0	0	120

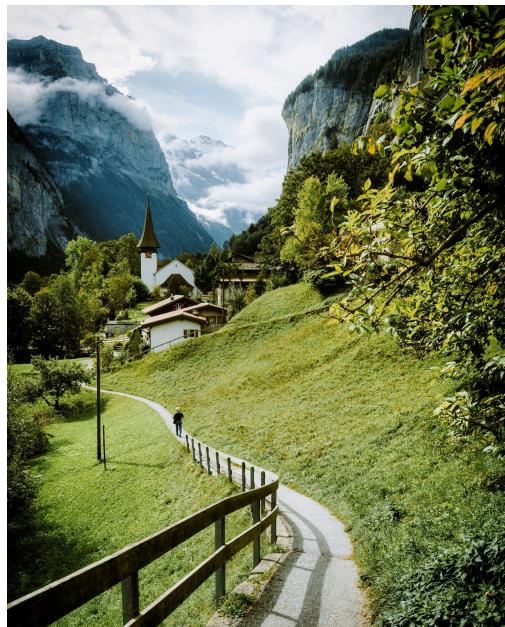
Output :	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>320.0018 ms</td> </tr> <tr> <td>Original size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>127.5000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(4).jpg ? Compression GIF: test\output\gif\test2(4).gif</pre>	COMPRESSION RESULTS		Execution Time	320.0018 ms	Original size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1	Optimal Threshold	127.5000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>330.1255 ms</td> </tr> <tr> <td>Original Size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>127.5000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(5).jpg ? Compression GIF: test\output\gif\test2(5).gif</pre>	COMPRESSION RESULTS		Execution Time	330.1255 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1	Optimal Threshold	127.5000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>124.4097 ms</td> </tr> <tr> <td>Original Size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(6).jpg ? Compression GIF: test\output\gif\test2(6).gif</pre>	COMPRESSION RESULTS		Execution Time	124.4097 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	320.0018 ms																																																
Original size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	127.5000																																																
COMPRESSION RESULTS																																																	
Execution Time	330.1255 ms																																																
Original Size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	127.5000																																																
COMPRESSION RESULTS																																																	
Execution Time	124.4097 ms																																																
Original Size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Input :																																																	
Max Pixel Difference																																																	
Nilai Ambang Batas :	0	0	255																																														
Output :	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>307.8876 ms</td> </tr> <tr> <td>Original Size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>255.0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(7).jpg ? Compression GIF: test\output\gif\test2(7).gif</pre>	COMPRESSION RESULTS		Execution Time	307.8876 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1	Optimal Threshold	255.0000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>312.7615 ms</td> </tr> <tr> <td>Original Size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>255.0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(8).jpg ? Compression GIF: test\output\gif\test2(8).gif</pre>	COMPRESSION RESULTS		Execution Time	312.7615 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1	Optimal Threshold	255.0000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>148.7714 ms</td> </tr> <tr> <td>Original Size</td> <td>4190 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1464 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>65.0597%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(9).jpg ? Compression GIF: test\output\gif\test2(9).gif</pre>	COMPRESSION RESULTS		Execution Time	148.7714 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	307.8876 ms																																																
Original Size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	255.0000																																																
COMPRESSION RESULTS																																																	
Execution Time	312.7615 ms																																																
Original Size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	255.0000																																																
COMPRESSION RESULTS																																																	
Execution Time	148.7714 ms																																																
Original Size	4190 bytes																																																
Compressed Size	1464 bytes																																																
Compression Rate	65.0597%																																																
Tree Depth	0																																																
Total Nodes	1																																																

Input :			
Entropy			
Nilai Ambang Batas :	0	0	8
Output :			
Input :			

Structural Similarity Index																																											
Nilai Ambang Batas :	0	0	1																																								
Output :	<p>COMPRESSION RESULTS</p> <table border="1"> <tbody> <tr><td>Execution Time</td><td>503.7501 ms</td></tr> <tr><td>Original Size</td><td>4190 bytes</td></tr> <tr><td>Compressed Size</td><td>1847 bytes</td></tr> <tr><td>Compression Rate</td><td>55.9189%</td></tr> <tr><td>Tree Depth</td><td>2</td></tr> <tr><td>Total Nodes</td><td>9</td></tr> <tr><td>Optimal Threshold</td><td>0.9292</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(13).jpg ? Compression GIF: test\output\gif\test2(13).gif</pre>	Execution Time	503.7501 ms	Original Size	4190 bytes	Compressed Size	1847 bytes	Compression Rate	55.9189%	Tree Depth	2	Total Nodes	9	Optimal Threshold	0.9292	<p>COMPRESSION RESULTS</p> <table border="1"> <tbody> <tr><td>Execution Time</td><td>393.4729 ms</td></tr> <tr><td>Original Size</td><td>4190 bytes</td></tr> <tr><td>Compressed Size</td><td>1464 bytes</td></tr> <tr><td>Compression Rate</td><td>65.0597%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> <tr><td>Optimal Threshold</td><td>1.0000</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(14).jpg ? Compression GIF: test\output\gif\test2(14).gif</pre>	Execution Time	393.4729 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1	Optimal Threshold	1.0000	<p>COMPRESSION RESULTS</p> <table border="1"> <tbody> <tr><td>Execution Time</td><td>143.2767 ms</td></tr> <tr><td>Original Size</td><td>4190 bytes</td></tr> <tr><td>Compressed Size</td><td>1464 bytes</td></tr> <tr><td>Compression Rate</td><td>65.0597%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test2(15).jpg ? Compression GIF: test\output\gif\test2(15).gif</pre>	Execution Time	143.2767 ms	Original Size	4190 bytes	Compressed Size	1464 bytes	Compression Rate	65.0597%	Tree Depth	0	Total Nodes	1
Execution Time	503.7501 ms																																										
Original Size	4190 bytes																																										
Compressed Size	1847 bytes																																										
Compression Rate	55.9189%																																										
Tree Depth	2																																										
Total Nodes	9																																										
Optimal Threshold	0.9292																																										
Execution Time	393.4729 ms																																										
Original Size	4190 bytes																																										
Compressed Size	1464 bytes																																										
Compression Rate	65.0597%																																										
Tree Depth	0																																										
Total Nodes	1																																										
Optimal Threshold	1.0000																																										
Execution Time	143.2767 ms																																										
Original Size	4190 bytes																																										
Compressed Size	1464 bytes																																										
Compression Rate	65.0597%																																										
Tree Depth	0																																										
Total Nodes	1																																										

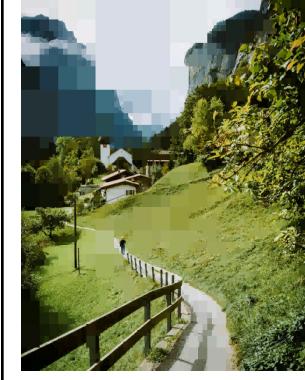
4.1.3 Kasus Eksperimen 3

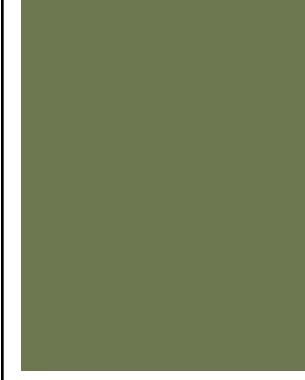
Gambar Asli :



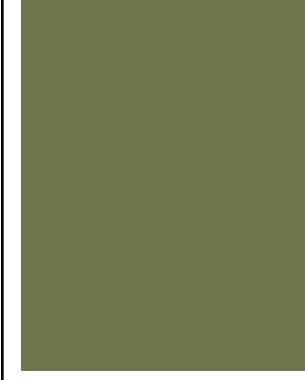
Hasil Kompresi :

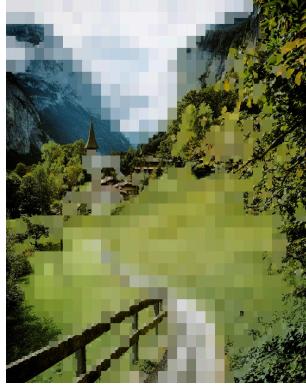
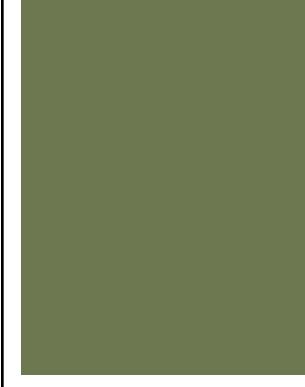
Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
---------------	-------------------------	-------------------------	--------------------------

	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%																																														
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Root Pixel Difference (0 - 255) 4. Entropy (0 - 1) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0 ? Absolute path for compressed image: test\output\image\test3(1).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(1).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Root Pixel Difference (0 - 255) 4. Entropy (0 - 1) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0 ? Absolute path for compressed image: test\output\image\test3(2).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(2).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Root Pixel Difference (0 - 255) 4. Entropy (0 - 1) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 1000 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test3(3).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(3).gif ? GIF path set!</pre>																																														
Variansi																																																	
Nilai Ambang Batas :	0	0	1000																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>38452.9487 ms</td></tr> <tr> <td>Original Size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>1323719 bytes</td></tr> <tr> <td>Compression Rate</td><td>49.9994%</td></tr> <tr> <td>Tree Depth</td><td>12</td></tr> <tr> <td>Total Nodes</td><td>437965</td></tr> <tr> <td>Optimal Threshold</td><td>1378.6657</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(1).jpg ? Compression GIF: test\output\gif\test3(1).gif</p>	COMPRESSION RESULTS		Execution Time	38452.9487 ms	Original Size	2647405 bytes	Compressed Size	1323719 bytes	Compression Rate	49.9994%	Tree Depth	12	Total Nodes	437965	Optimal Threshold	1378.6657	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>64645.0706 ms</td></tr> <tr> <td>Original Size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>529422 bytes</td></tr> <tr> <td>Compression Rate</td><td>80.0022%</td></tr> <tr> <td>Tree Depth</td><td>12</td></tr> <tr> <td>Total Nodes</td><td>44997</td></tr> <tr> <td>Optimal Threshold</td><td>2768.5294</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(2).jpg ? Compression GIF: test\output\gif\test3(2).gif</p>	COMPRESSION RESULTS		Execution Time	64645.0706 ms	Original Size	2647405 bytes	Compressed Size	529422 bytes	Compression Rate	80.0022%	Tree Depth	12	Total Nodes	44997	Optimal Threshold	2768.5294	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>20213.4724 ms</td></tr> <tr> <td>Original Size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>1298513 bytes</td></tr> <tr> <td>Compression Rate</td><td>50.9515%</td></tr> <tr> <td>Tree Depth</td><td>9</td></tr> <tr> <td>Total Nodes</td><td>124113</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(3).jpg ? Compression GIF: test\output\gif\test3(3).gif</p>	COMPRESSION RESULTS		Execution Time	20213.4724 ms	Original Size	2647405 bytes	Compressed Size	1298513 bytes	Compression Rate	50.9515%	Tree Depth	9	Total Nodes	124113
COMPRESSION RESULTS																																																	
Execution Time	38452.9487 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	1323719 bytes																																																
Compression Rate	49.9994%																																																
Tree Depth	12																																																
Total Nodes	437965																																																
Optimal Threshold	1378.6657																																																
COMPRESSION RESULTS																																																	
Execution Time	64645.0706 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	529422 bytes																																																
Compression Rate	80.0022%																																																
Tree Depth	12																																																
Total Nodes	44997																																																
Optimal Threshold	2768.5294																																																
COMPRESSION RESULTS																																																	
Execution Time	20213.4724 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	1298513 bytes																																																
Compression Rate	50.9515%																																																
Tree Depth	9																																																
Total Nodes	124113																																																

<p>Input :</p> <pre>? Absolute path of input image: test\input\test3.jpg ? Valid image found! ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1026.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127.5): 0 ? threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0% ? Absolute path for compressed image: test\output\image\test3(4).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(4).gif ? GIF path set!</pre>	<pre>? Absolute path of input image: test\input\test3.jpg ? Valid image found! ? QUADTREE IMAGE COMPRESSION TOOL ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1026.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127.5): 0 ? threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0% ? Absolute path for compressed image: test\output\image\test3(5).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(5).gif ? GIF path set!</pre>	<pre>? Absolute path of input image: test\input\test3.jpg ? Valid image found! ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1026.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation ? Enter threshold (0-127.5): 120 ? threshold set! ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test3(6).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(6).gif ? GIF path set!</pre>																																														
<p>Mean Absolute Deviation</p>																																																
<p>Nilai Ambang Batas :</p>	<p>0</p>	<p>0</p>	<p>120</p>																																													
<p>Output :</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td style="width: 15%;">Execution Time</td><td style="width: 85%;">78450.3774 ms</td></tr> <tr> <td>Original size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>1323235 bytes</td></tr> <tr> <td>Compression Rate</td><td>50.0177%</td></tr> <tr> <td>Tree Depth</td><td>12</td></tr> <tr> <td>Total Nodes</td><td>489421</td></tr> <tr> <td>Optimal Threshold</td><td>29.0253</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(4).jpg ? Compression GIF: test\output\gif\test3(4).gif</p>	COMPRESSION RESULTS		Execution Time	78450.3774 ms	Original size	2647405 bytes	Compressed Size	1323235 bytes	Compression Rate	50.0177%	Tree Depth	12	Total Nodes	489421	Optimal Threshold	29.0253	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td style="width: 15%;">Execution Time</td><td style="width: 85%;">65651.6541 ms</td></tr> <tr> <td>Original Size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>530027 bytes</td></tr> <tr> <td>Compression Rate</td><td>79.9794%</td></tr> <tr> <td>Tree Depth</td><td>12</td></tr> <tr> <td>Total Nodes</td><td>47153</td></tr> <tr> <td>Optimal Threshold</td><td>43.1331</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(5).jpg ? Compression GIF: test\output\gif\test3(5).gif</p>	COMPRESSION RESULTS		Execution Time	65651.6541 ms	Original Size	2647405 bytes	Compressed Size	530027 bytes	Compression Rate	79.9794%	Tree Depth	12	Total Nodes	47153	Optimal Threshold	43.1331	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td style="width: 15%;">Execution Time</td><td style="width: 85%;">4958.9017 ms</td></tr> <tr> <td>Original size</td><td>2647405 bytes</td></tr> <tr> <td>Compressed Size</td><td>208870 bytes</td></tr> <tr> <td>Compression Rate</td><td>92.1165%</td></tr> <tr> <td>Tree Depth</td><td>0</td></tr> <tr> <td>Total Nodes</td><td>1</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(6).jpg ? Compression GIF: test\output\gif\test3(6).gif</p>	COMPRESSION RESULTS		Execution Time	4958.9017 ms	Original size	2647405 bytes	Compressed Size	208870 bytes	Compression Rate	92.1165%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																
Execution Time	78450.3774 ms																																															
Original size	2647405 bytes																																															
Compressed Size	1323235 bytes																																															
Compression Rate	50.0177%																																															
Tree Depth	12																																															
Total Nodes	489421																																															
Optimal Threshold	29.0253																																															
COMPRESSION RESULTS																																																
Execution Time	65651.6541 ms																																															
Original Size	2647405 bytes																																															
Compressed Size	530027 bytes																																															
Compression Rate	79.9794%																																															
Tree Depth	12																																															
Total Nodes	47153																																															
Optimal Threshold	43.1331																																															
COMPRESSION RESULTS																																																
Execution Time	4958.9017 ms																																															
Original size	2647405 bytes																																															
Compressed Size	208870 bytes																																															
Compression Rate	92.1165%																																															
Tree Depth	0																																															
Total Nodes	1																																															

<p>Input :</p>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test1.jpg ? Valid image found. ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1625.25) 2. Mean Absolute Deviation (0 - 121.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set to 0 ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.00 ? Absolute path for compressed image: test\output\image\test3(7).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(7).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found. ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1625.25) 2. Mean Absolute Deviation (0 - 121.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 0 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set to 0 ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to no.08 ? Absolute path for compressed image: test\output\image\test3(8).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(8).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found. ? ERROR MEASUREMENT METHOD 1. Variance (0 - 1625.25) 2. Mean Absolute Deviation (0 - 121.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 255 ? Threshold set to 255 ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test3(9).jpg ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(9).gif ? GIF path set!</pre>																																														
<p>Max Pixel Difference</p>																																																	
<p>Nilai Ambang Batas :</p>	0	0	255																																														
<p>Output :</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td style="text-align: center;">70132.3106 ms</td> </tr> <tr> <td>Original Size</td> <td style="text-align: center;">2647405 bytes</td> </tr> <tr> <td>Compressed Size</td> <td style="text-align: center;">1327110 bytes</td> </tr> <tr> <td>Compression Rate</td> <td style="text-align: center;">49.8713%</td> </tr> <tr> <td>Tree Depth</td> <td style="text-align: center;">12</td> </tr> <tr> <td>Total Nodes</td> <td style="text-align: center;">178973</td> </tr> <tr> <td>Optimal Threshold</td> <td style="text-align: center;">175.3333</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(7).jpg ? Compression GIF: test\output\gif\test3(7).gif</p>	COMPRESSION RESULTS		Execution Time	70132.3106 ms	Original Size	2647405 bytes	Compressed Size	1327110 bytes	Compression Rate	49.8713%	Tree Depth	12	Total Nodes	178973	Optimal Threshold	175.3333	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td style="text-align: center;">64933.0733 ms</td> </tr> <tr> <td>Original Size</td> <td style="text-align: center;">2647405 bytes</td> </tr> <tr> <td>Compressed Size</td> <td style="text-align: center;">308424 bytes</td> </tr> <tr> <td>Compression Rate</td> <td style="text-align: center;">88.4255%</td> </tr> <tr> <td>Tree Depth</td> <td style="text-align: center;">11</td> </tr> <tr> <td>Total Nodes</td> <td style="text-align: center;">1141</td> </tr> <tr> <td>Optimal Threshold</td> <td style="text-align: center;">248.5254</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(8).jpg ? Compression GIF: test\output\gif\test3(8).gif</p>	COMPRESSION RESULTS		Execution Time	64933.0733 ms	Original Size	2647405 bytes	Compressed Size	308424 bytes	Compression Rate	88.4255%	Tree Depth	11	Total Nodes	1141	Optimal Threshold	248.5254	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td style="text-align: center;">5179.1910 ms</td> </tr> <tr> <td>Original Size</td> <td style="text-align: center;">2647405 bytes</td> </tr> <tr> <td>Compressed Size</td> <td style="text-align: center;">208707 bytes</td> </tr> <tr> <td>Compression Rate</td> <td style="text-align: center;">92.1165%</td> </tr> <tr> <td>Tree Depth</td> <td style="text-align: center;">0</td> </tr> <tr> <td>Total Nodes</td> <td style="text-align: center;">1</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test3(9).jpg ? Compression GIF: test\output\gif\test3(9).gif</p>	COMPRESSION RESULTS		Execution Time	5179.1910 ms	Original Size	2647405 bytes	Compressed Size	208707 bytes	Compression Rate	92.1165%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	70132.3106 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	1327110 bytes																																																
Compression Rate	49.8713%																																																
Tree Depth	12																																																
Total Nodes	178973																																																
Optimal Threshold	175.3333																																																
COMPRESSION RESULTS																																																	
Execution Time	64933.0733 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	308424 bytes																																																
Compression Rate	88.4255%																																																
Tree Depth	11																																																
Total Nodes	1141																																																
Optimal Threshold	248.5254																																																
COMPRESSION RESULTS																																																	
Execution Time	5179.1910 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	208707 bytes																																																
Compression Rate	92.1165%																																																
Tree Depth	0																																																
Total Nodes	1																																																

<p>Input :</p>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found!</pre> <table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td> <td>(0 - 16256, 25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 137.5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 8)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> </tbody> </table> <pre>?? choose method (1-5): 4 ? method selected: Entropy ? Enter threshold (0.0-0.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test3(10).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(10).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256, 25)	2. Mean Absolute Deviation	(0 - 137.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found!</pre> <table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td> <td>(0 - 16256, 25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 137.5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 8)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> </tbody> </table> <pre>?? choose method (1-5): 4 ? method selected: Entropy ? Enter threshold (0.0-0.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test3(11).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(11).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256, 25)	2. Mean Absolute Deviation	(0 - 137.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found!</pre> <table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td> <td>(0 - 16256, 25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 137.5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 8)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> </tbody> </table> <pre>?? choose method (1-5): 4 ? method selected: Entropy ? Enter threshold (0.0-0.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compressive disabled!</pre> <pre>? Absolute path for compressed image: test\output\image\test3(12).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(12).gif ? GIF path set!</pre>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256, 25)	2. Mean Absolute Deviation	(0 - 137.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)										
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256, 25)																																																
2. Mean Absolute Deviation	(0 - 137.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256, 25)																																																
2. Mean Absolute Deviation	(0 - 137.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256, 25)																																																
2. Mean Absolute Deviation	(0 - 137.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
<p>Entropy</p>																																																	
<p>Nilai Ambang Batas :</p>	0	0	8																																														
<p>Output :</p>	<table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>45414.2429 ms</td> </tr> <tr> <td>Original Size</td> <td>2647405 bytes</td> </tr> <tr> <td>compressed size</td> <td>1323706 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>49.999%</td> </tr> <tr> <td>Tree Depth</td> <td>9</td> </tr> <tr> <td>Total Nodes</td> <td>141417</td> </tr> <tr> <td>Optimal Threshold</td> <td>5.9818</td> </tr> </tbody> </table> <pre>? output Files: ? Compressed Image: test\output\image\test3(10).jpg ? Compression GIF: test\output\gif\test3(10).gif</pre>	COMPRESSION RESULTS		Execution Time	45414.2429 ms	Original Size	2647405 bytes	compressed size	1323706 bytes	Compression Rate	49.999%	Tree Depth	9	Total Nodes	141417	Optimal Threshold	5.9818	<table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>62401.5356 ms</td> </tr> <tr> <td>Original Size</td> <td>2647405 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>530034 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>79.9791%</td> </tr> <tr> <td>Tree Depth</td> <td>9</td> </tr> <tr> <td>Total Nodes</td> <td>15577</td> </tr> <tr> <td>Optimal Threshold</td> <td>6.8491</td> </tr> </tbody> </table> <pre>? output Files: ? Compressed Image: test\output\image\test3(11).jpg ? Compression GIF: test\output\gif\test3(11).gif</pre>	COMPRESSION RESULTS		Execution Time	62401.5356 ms	Original Size	2647405 bytes	Compressed Size	530034 bytes	Compression Rate	79.9791%	Tree Depth	9	Total Nodes	15577	Optimal Threshold	6.8491	<table border="1" style="margin-top: 10px; border-collapse: collapse; width: 100%;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>5485.3369 ms</td> </tr> <tr> <td>Original Size</td> <td>2647405 bytes</td> </tr> <tr> <td>compressed size</td> <td>288707 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>92.1165%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? output Files: ? Compressed Image: test\output\image\test3(12).jpg ? Compression GIF: test\output\gif\test3(12).gif</pre>	COMPRESSION RESULTS		Execution Time	5485.3369 ms	Original Size	2647405 bytes	compressed size	288707 bytes	Compression Rate	92.1165%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	45414.2429 ms																																																
Original Size	2647405 bytes																																																
compressed size	1323706 bytes																																																
Compression Rate	49.999%																																																
Tree Depth	9																																																
Total Nodes	141417																																																
Optimal Threshold	5.9818																																																
COMPRESSION RESULTS																																																	
Execution Time	62401.5356 ms																																																
Original Size	2647405 bytes																																																
Compressed Size	530034 bytes																																																
Compression Rate	79.9791%																																																
Tree Depth	9																																																
Total Nodes	15577																																																
Optimal Threshold	6.8491																																																
COMPRESSION RESULTS																																																	
Execution Time	5485.3369 ms																																																
Original Size	2647405 bytes																																																
compressed size	288707 bytes																																																
Compression Rate	92.1165%																																																
Tree Depth	0																																																
Total Nodes	1																																																

<p>Input :</p> <div style="display: flex; justify-content: space-around;"> <div style="width: 45%;"> <pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found!</pre> <table border="1" style="margin-top: 10px; border-collapse: collapse;"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 16256.25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 129.5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 295)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 5 ? Method selected: SSIM ? Enter threshold (0.0-1.0): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test3(13).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(13).gif ? GIF path set!</pre> </div> <div style="width: 45%;"> <pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test3.jpg ? Valid image found!</pre> <table border="1" style="margin-top: 10px; border-collapse: collapse;"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr><td>1. Variance</td><td>(0 - 16256.25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 127.5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 295)</td></tr> <tr><td>4. Entropy</td><td>(0 - 8)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <pre>?? Choose method (1-5): 5 ? Method selected: SSIM ? Enter threshold (0.0-1.0): 1 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test3(14).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test3(14).gif ? GIF path set!</pre> </div> </div>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256.25)	2. Mean Absolute Deviation	(0 - 129.5)	3. Max Pixel Difference	(0 - 295)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256.25)	2. Mean Absolute Deviation	(0 - 127.5)	3. Max Pixel Difference	(0 - 295)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<p>Structural Similarity Index</p>   																					
? ERROR MEASUREMENT METHOD																																														
1. Variance	(0 - 16256.25)																																													
2. Mean Absolute Deviation	(0 - 129.5)																																													
3. Max Pixel Difference	(0 - 295)																																													
4. Entropy	(0 - 8)																																													
5. Structural Similarity Index	(0 - 1)																																													
? ERROR MEASUREMENT METHOD																																														
1. Variance	(0 - 16256.25)																																													
2. Mean Absolute Deviation	(0 - 127.5)																																													
3. Max Pixel Difference	(0 - 295)																																													
4. Entropy	(0 - 8)																																													
5. Structural Similarity Index	(0 - 1)																																													
<p>Nilai Ambang Batas :</p>	0	0	1																																											
<p>Output :</p> <div style="display: flex; justify-content: space-around;"> <div style="width: 33%;"> <table border="1" style="margin-top: 10px; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>111969.0313 ms</td></tr> <tr><td>Original Size</td><td>2647405 bytes</td></tr> <tr><td>Compressed Size</td><td>1323720 bytes</td></tr> <tr><td>Compression Rate</td><td>49.9993%</td></tr> <tr><td>Tree Depth</td><td>12</td></tr> <tr><td>Total Nodes</td><td>5101369</td></tr> <tr><td>Optimal Threshold</td><td>0.2623</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test3(13).jpg ? Compression GIF: test\output\gif\test3(13).gif</pre> </div> <div style="width: 33%;"> <table border="1" style="margin-top: 10px; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>188852.1584 ms</td></tr> <tr><td>Original Size</td><td>2647405 bytes</td></tr> <tr><td>Compressed Size</td><td>529335 bytes</td></tr> <tr><td>Compression Rate</td><td>80.005%</td></tr> <tr><td>Tree Depth</td><td>12</td></tr> <tr><td>Total Nodes</td><td>1470245</td></tr> <tr><td>Optimal Threshold</td><td>0.5939</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test3(14).jpg ? Compression GIF: test\output\gif\test3(14).gif</pre> </div> <div style="width: 33%;"> <table border="1" style="margin-top: 10px; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr><td>Execution Time</td><td>5838.8046 ms</td></tr> <tr><td>Original Size</td><td>2647405 bytes</td></tr> <tr><td>Compressed Size</td><td>208707 bytes</td></tr> <tr><td>Compression Rate</td><td>92.116%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test3(15).jpg ? Compression GIF: test\output\gif\test3(15).gif</pre> </div> </div>	COMPRESSION RESULTS		Execution Time	111969.0313 ms	Original Size	2647405 bytes	Compressed Size	1323720 bytes	Compression Rate	49.9993%	Tree Depth	12	Total Nodes	5101369	Optimal Threshold	0.2623	COMPRESSION RESULTS		Execution Time	188852.1584 ms	Original Size	2647405 bytes	Compressed Size	529335 bytes	Compression Rate	80.005%	Tree Depth	12	Total Nodes	1470245	Optimal Threshold	0.5939	COMPRESSION RESULTS		Execution Time	5838.8046 ms	Original Size	2647405 bytes	Compressed Size	208707 bytes	Compression Rate	92.116%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																														
Execution Time	111969.0313 ms																																													
Original Size	2647405 bytes																																													
Compressed Size	1323720 bytes																																													
Compression Rate	49.9993%																																													
Tree Depth	12																																													
Total Nodes	5101369																																													
Optimal Threshold	0.2623																																													
COMPRESSION RESULTS																																														
Execution Time	188852.1584 ms																																													
Original Size	2647405 bytes																																													
Compressed Size	529335 bytes																																													
Compression Rate	80.005%																																													
Tree Depth	12																																													
Total Nodes	1470245																																													
Optimal Threshold	0.5939																																													
COMPRESSION RESULTS																																														
Execution Time	5838.8046 ms																																													
Original Size	2647405 bytes																																													
Compressed Size	208707 bytes																																													
Compression Rate	92.116%																																													
Tree Depth	0																																													
Total Nodes	1																																													

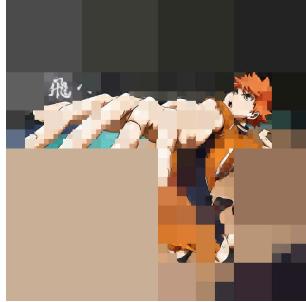
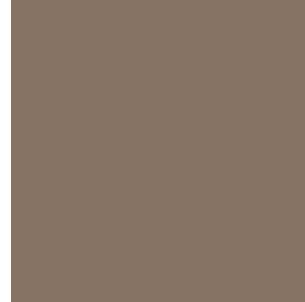
4.1.4 Kasus Eksperimen 4

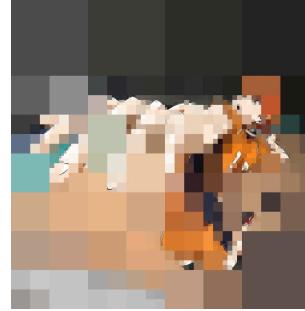
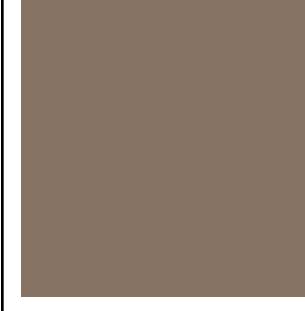
Gambar Asli :

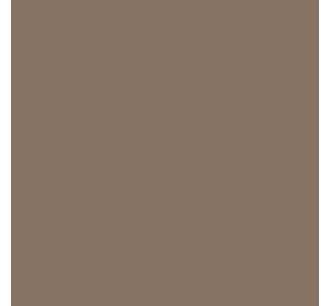
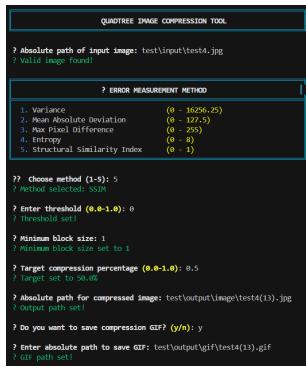
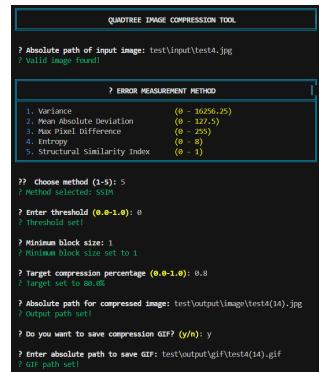
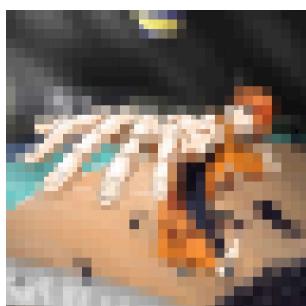
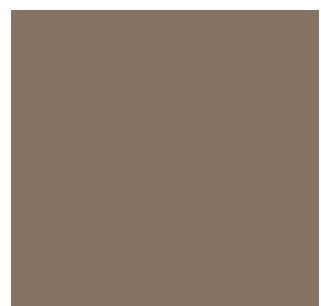


Hasil Kompresi :

Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) </div> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test4(1).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(1).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) </div> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set!</pre> <pre>? Minimum block size: 1 ? Minimum block size set to 1</pre> <pre>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre>? Absolute path for compressed image: test\output\image\test4(2).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(2).gif ? GIF path set!</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) </div> <pre>?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 1000 ? Threshold set!</pre> <pre>? Minimum block size: 20 ? Minimum block size set to 20</pre> <pre>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled</pre> <pre>? Absolute path for compressed image: test\output\image\test4(3).jpg ? Output path set!</pre> <pre>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(3).gif ? GIF path set!</pre>
Variansi			
Nilai Ambang Batas :	0	0	1000

<p>Output :</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>42125.2993 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>299051 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>48.7419%</td> </tr> <tr> <td>Tree Depth</td> <td>12</td> </tr> <tr> <td>Total Nodes</td> <td>32001</td> </tr> <tr> <td>Optimal Threshold</td> <td>2416.3903</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(1).jpg ? Compression GIF: test\output\gif\test4(1).gif</pre>	COMPRESSION RESULTS		Execution Time	42125.2993 ms	Original Size	583422 bytes	Compressed Size	299051 bytes	Compression Rate	48.7419%	Tree Depth	12	Total Nodes	32001	Optimal Threshold	2416.3903	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>20979.5904 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8800%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>16256.2500</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(2).jpg ? Compression GIF: test\output\gif\test4(2).gif</pre>	COMPRESSION RESULTS		Execution Time	20979.5904 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8800%	Tree Depth	0	Total Nodes	1	Optimal Threshold	16256.2500	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>12476.3469 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>437827 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>24.9553%</td> </tr> <tr> <td>Tree Depth</td> <td>9</td> </tr> <tr> <td>Total Nodes</td> <td>26977</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(3).jpg ? Compression GIF: test\output\gif\test4(3).gif</pre>	COMPRESSION RESULTS		Execution Time	12476.3469 ms	Original Size	583422 bytes	Compressed Size	437827 bytes	Compression Rate	24.9553%	Tree Depth	9	Total Nodes	26977
COMPRESSION RESULTS																																																	
Execution Time	42125.2993 ms																																																
Original Size	583422 bytes																																																
Compressed Size	299051 bytes																																																
Compression Rate	48.7419%																																																
Tree Depth	12																																																
Total Nodes	32001																																																
Optimal Threshold	2416.3903																																																
COMPRESSION RESULTS																																																	
Execution Time	20979.5904 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8800%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	16256.2500																																																
COMPRESSION RESULTS																																																	
Execution Time	12476.3469 ms																																																
Original Size	583422 bytes																																																
Compressed Size	437827 bytes																																																
Compression Rate	24.9553%																																																
Tree Depth	9																																																
Total Nodes	26977																																																
<p>Input :</p>	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2</p> <p>? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 0</p> <p>? threshold set!</p> <p>? Minimum block size: 1</p> <p>? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.5</p> <p>? Target set to 50.0%</p> <p>? Absolute path for compressed image: test\output\image\test4(4).jpg</p> <p>? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test4(4).gif</p> <p>? GIF path set!</p>	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2</p> <p>? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 0</p> <p>? threshold set!</p> <p>? Minimum block size: 1</p> <p>? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.8</p> <p>? Target set to 80.0%</p> <p>? Absolute path for compressed image: test\output\image\test4(5).jpg</p> <p>? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test4(5).gif</p> <p>? GIF path set!</p>	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <pre>? Absolute path of input image: test\input\test4.jpg ? Valid image found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2</p> <p>? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 120</p> <p>? threshold set!</p> <p>? Minimum block size: 20</p> <p>? Minimum block size set to 20</p> <p>? Target compression percentage (0.0-1.0): 0</p> <p>? Target compression disabled</p> <p>? Absolute path for compressed image: test\output\image\test4(6).jpg</p> <p>? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test4(6).gif</p> <p>? GIF path set!</p>																																														
<p>Mean Absolute Deviation</p>																																																	
<p>Nilai Ambang Batas :</p>	<p>0</p>	<p>0</p>	<p>120</p>																																														
<p>Output :</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>44093.3504 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>291670 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50.0070%</td> </tr> <tr> <td>Tree Depth</td> <td>12</td> </tr> <tr> <td>Total Nodes</td> <td>32105</td> </tr> <tr> <td>Optimal Threshold</td> <td>34.6169</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(4).jpg ? Compression GIF: test\output\gif\test4(4).gif</pre>	COMPRESSION RESULTS		Execution Time	44093.3504 ms	Original Size	583422 bytes	Compressed Size	291670 bytes	Compression Rate	50.0070%	Tree Depth	12	Total Nodes	32105	Optimal Threshold	34.6169	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>22377.8216 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8800%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>127.5000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(5).jpg ? Compression GIF: test\output\gif\test4(5).gif</pre>	COMPRESSION RESULTS		Execution Time	22377.8216 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8800%	Tree Depth	0	Total Nodes	1	Optimal Threshold	127.5000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>2115.3889 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8800%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(6).jpg ? Compression GIF: test\output\gif\test4(6).gif</pre>	COMPRESSION RESULTS		Execution Time	2115.3889 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8800%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	44093.3504 ms																																																
Original Size	583422 bytes																																																
Compressed Size	291670 bytes																																																
Compression Rate	50.0070%																																																
Tree Depth	12																																																
Total Nodes	32105																																																
Optimal Threshold	34.6169																																																
COMPRESSION RESULTS																																																	
Execution Time	22377.8216 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8800%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	127.5000																																																
COMPRESSION RESULTS																																																	
Execution Time	2115.3889 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8800%																																																
Tree Depth	0																																																
Total Nodes	1																																																

<p>Input :</p> <pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0% ? Absolute path for compressed image: test\output\image\test4(7).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(7).gif ? GIF path set</pre>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0% ? Absolute path for compressed image: test\output\image\test4(8).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(8).gif ? GIF path set</pre>	<pre>? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 3 ? Method selected: Max Pixel Difference ? Enter threshold (0-255): 255 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test4(9).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(9).gif ? GIF path set</pre>																																														
<p>Max Pixel Difference</p> 																																																
<p>Nilai Ambang Batas :</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td>0</td> <td>0</td> <td>255</td> </tr> </table>	0	0	255																																													
0	0	255																																														
<p>Output :</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>37789.9503 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>291888 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>49.9697%</td> </tr> <tr> <td>Tree Depth</td> <td>12</td> </tr> <tr> <td>Total Nodes</td> <td>8737</td> </tr> <tr> <td>Optimal Threshold</td> <td>246.6667</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(7).jpg ? Compression GIF: test\output\gif\test4(7).gif</pre>	COMPRESSION RESULTS		Execution Time	37789.9503 ms	Original Size	583422 bytes	Compressed Size	291888 bytes	Compression Rate	49.9697%	Tree Depth	12	Total Nodes	8737	Optimal Threshold	246.6667	<table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>36716.8953 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>240641 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>58.753%</td> </tr> <tr> <td>Tree Depth</td> <td>12</td> </tr> <tr> <td>Total Nodes</td> <td>2353</td> </tr> <tr> <td>Optimal Threshold</td> <td>255.0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(8).jpg ? Compression GIF: test\output\gif\test4(8).gif</pre>	COMPRESSION RESULTS		Execution Time	36716.8953 ms	Original Size	583422 bytes	Compressed Size	240641 bytes	Compression Rate	58.753%	Tree Depth	12	Total Nodes	2353	Optimal Threshold	255.0000	<table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>2130.9664 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8800%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(9).jpg ? Compression GIF: test\output\gif\test4(9).gif</pre>	COMPRESSION RESULTS		Execution Time	2130.9664 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8800%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																
Execution Time	37789.9503 ms																																															
Original Size	583422 bytes																																															
Compressed Size	291888 bytes																																															
Compression Rate	49.9697%																																															
Tree Depth	12																																															
Total Nodes	8737																																															
Optimal Threshold	246.6667																																															
COMPRESSION RESULTS																																																
Execution Time	36716.8953 ms																																															
Original Size	583422 bytes																																															
Compressed Size	240641 bytes																																															
Compression Rate	58.753%																																															
Tree Depth	12																																															
Total Nodes	2353																																															
Optimal Threshold	255.0000																																															
COMPRESSION RESULTS																																																
Execution Time	2130.9664 ms																																															
Original Size	583422 bytes																																															
Compressed Size	146551 bytes																																															
Compression Rate	74.8800%																																															
Tree Depth	0																																															
Total Nodes	1																																															
<p>Input :</p> <pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0-8.0): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0% ? Absolute path for compressed image: test\output\image\test4(10).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(10).gif ? GIF path set</pre>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0-8.0): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0% ? Absolute path for compressed image: test\output\image\test4(11).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(11).gif ? GIF path set</pre>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test4.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0-8.0): 8 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test4(12).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test4(12).gif ? GIF path set</pre>																																														

Entropy																																																	
Nilai Ambang Batas :	0	0	8																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>38050.8440 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>291839 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>49.9781%</td> </tr> <tr> <td>Tree Depth</td> <td>9</td> </tr> <tr> <td>Total Nodes</td> <td>4813</td> </tr> <tr> <td>Optimal Threshold</td> <td>5.9375</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(10).jpg ? Compression GIF: test\output\gif\test4(10).gif</pre>	COMPRESSION RESULTS		Execution Time	38050.8440 ms	Original Size	583422 bytes	Compressed Size	291839 bytes	Compression Rate	49.9781%	Tree Depth	9	Total Nodes	4813	Optimal Threshold	5.9375	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>20118.7128 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8802%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>8.0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(11).jpg ? Compression GIF: test\output\gif\test4(11).gif</pre>	COMPRESSION RESULTS		Execution Time	20118.7128 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8802%	Tree Depth	0	Total Nodes	1	Optimal Threshold	8.0000	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>2083.1717 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8802%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test4(12).jpg ? Compression GIF: test\output\gif\test4(12).gif</pre>	COMPRESSION RESULTS		Execution Time	2083.1717 ms	Original Size	583422 bytes	Compressed size	146551 bytes	Compression Rate	74.8802%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	38050.8440 ms																																																
Original Size	583422 bytes																																																
Compressed Size	291839 bytes																																																
Compression Rate	49.9781%																																																
Tree Depth	9																																																
Total Nodes	4813																																																
Optimal Threshold	5.9375																																																
COMPRESSION RESULTS																																																	
Execution Time	20118.7128 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8802%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	8.0000																																																
COMPRESSION RESULTS																																																	
Execution Time	2083.1717 ms																																																
Original Size	583422 bytes																																																
Compressed size	146551 bytes																																																
Compression Rate	74.8802%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Input :																																																	
Structural Similarity Index																																																	
Nilai	0	0	1																																														

Ambang Batas :																																																	
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>109263.8604 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>291436 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50.0471%</td> </tr> <tr> <td>Tree Depth</td> <td>12</td> </tr> <tr> <td>Total Nodes</td> <td>2929</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.6995</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test4(13).jpg ? Compression GIF: test\output\gif\test4(13).gif</p>	COMPRESSION RESULTS		Execution Time	109263.8604 ms	Original Size	583422 bytes	Compressed Size	291436 bytes	Compression Rate	50.0471%	Tree Depth	12	Total Nodes	2929	Optimal Threshold	0.6995	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>37712.2924 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8888%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> <tr> <td>Optimal Threshold</td> <td>1.0000</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test4(14).jpg ? Compression GIF: test\output\gif\test4(14).gif</p>	COMPRESSION RESULTS		Execution Time	37712.2924 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8888%	Tree Depth	0	Total Nodes	1	Optimal Threshold	1.0000	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>2346.9447 ms</td> </tr> <tr> <td>Original Size</td> <td>583422 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>146551 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>74.8888%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test4(15).jpg ? Compression GIF: test\output\gif\test4(15).gif</p>	COMPRESSION RESULTS		Execution Time	2346.9447 ms	Original Size	583422 bytes	Compressed Size	146551 bytes	Compression Rate	74.8888%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	109263.8604 ms																																																
Original Size	583422 bytes																																																
Compressed Size	291436 bytes																																																
Compression Rate	50.0471%																																																
Tree Depth	12																																																
Total Nodes	2929																																																
Optimal Threshold	0.6995																																																
COMPRESSION RESULTS																																																	
Execution Time	37712.2924 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8888%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Optimal Threshold	1.0000																																																
COMPRESSION RESULTS																																																	
Execution Time	2346.9447 ms																																																
Original Size	583422 bytes																																																
Compressed Size	146551 bytes																																																
Compression Rate	74.8888%																																																
Tree Depth	0																																																
Total Nodes	1																																																

4.1.5 Kasus Eksperimen 5

Gambar Asli :



Hasil Kompresi :

Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test5.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0% ? Absolute path for compressed image: test\output\image\test5(1).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(1).gif ? GIF path set</pre>	<pre>? Absolute path of input image: test\input\test5.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0% ? Absolute path for compressed image: test\output\image\test5(2).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(2).gif ? GIF path set</pre>	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test5.jpg ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 1000 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test5(3).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(3).gif ? GIF path set</pre>

Variansi																																											
Nilai Ambang Batas :	0	0	1000																																								
Output :	<p>COMPRESSION RESULTS</p> <table border="1"> <tr><td>Execution Time</td><td>891.745 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>6287 bytes</td></tr> <tr><td>Compression Rate</td><td>50.3671%</td></tr> <tr><td>Tree Depth</td><td>8</td></tr> <tr><td>Total Nodes</td><td>721</td></tr> <tr><td>Optimal Threshold</td><td>4872.8599</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\tests(1).jpg ? Compression GIF: test\output\gif\tests(1).gif</p>	Execution Time	891.745 ms	Original Size	12667 bytes	Compressed Size	6287 bytes	Compression Rate	50.3671%	Tree Depth	8	Total Nodes	721	Optimal Threshold	4872.8599	<p>COMPRESSION RESULTS</p> <table border="1"> <tr><td>Execution Time</td><td>726.3329 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>2634 bytes</td></tr> <tr><td>Compression Rate</td><td>79.2056%</td></tr> <tr><td>Tree Depth</td><td>4</td></tr> <tr><td>Total Nodes</td><td>21</td></tr> <tr><td>Optimal Threshold</td><td>6907.0414</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\tests(2).jpg ? Compression GIF: test\output\gif\tests(2).gif</p>	Execution Time	726.3329 ms	Original Size	12667 bytes	Compressed Size	2634 bytes	Compression Rate	79.2056%	Tree Depth	4	Total Nodes	21	Optimal Threshold	6907.0414	<p>COMPRESSION RESULTS</p> <table border="1"> <tr><td>Execution Time</td><td>337.3263 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>8368 bytes</td></tr> <tr><td>Compression Rate</td><td>33.9386%</td></tr> <tr><td>Tree Depth</td><td>5</td></tr> <tr><td>Total Nodes</td><td>881</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\tests(3).jpg ? Compression GIF: test\output\gif\tests(3).gif</p>	Execution Time	337.3263 ms	Original Size	12667 bytes	Compressed Size	8368 bytes	Compression Rate	33.9386%	Tree Depth	5	Total Nodes	881
Execution Time	891.745 ms																																										
Original Size	12667 bytes																																										
Compressed Size	6287 bytes																																										
Compression Rate	50.3671%																																										
Tree Depth	8																																										
Total Nodes	721																																										
Optimal Threshold	4872.8599																																										
Execution Time	726.3329 ms																																										
Original Size	12667 bytes																																										
Compressed Size	2634 bytes																																										
Compression Rate	79.2056%																																										
Tree Depth	4																																										
Total Nodes	21																																										
Optimal Threshold	6907.0414																																										
Execution Time	337.3263 ms																																										
Original Size	12667 bytes																																										
Compressed Size	8368 bytes																																										
Compression Rate	33.9386%																																										
Tree Depth	5																																										
Total Nodes	881																																										
Input :	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <p>? Absolute path of input image: test\input\tests(5).jpg ? Valid image found!</p> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1024.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 0 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 0.5</p> <p>? Absolute path for compressed image: test\output\image\tests(5).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\tests(5).gif ? GIF path set!</p>	<p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1024.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 128 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 0.8</p> <p>? Absolute path for compressed image: test\output\image\tests(5).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\tests(5).gif ? GIF path set!</p>	<p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1024.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>? Choose method (1-5): 2 ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 128 ? Threshold set!</p> <p>? Minimum block size: 20 ? Minimum block size set to 20</p> <p>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled</p> <p>? Absolute path for compressed image: test\output\image\tests(5).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\tests(5).gif ? GIF path set!</p>																																								
Mean Absolute Deviation																																											
Nilai Ambang Batas :	0	0	120																																								

<p>Output :</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>879,9418 ms</td> </tr> <tr> <td>Original size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>6294 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50,3118%</td> </tr> <tr> <td>Tree Depth</td> <td>8</td> </tr> <tr> <td>Total Nodes</td> <td>725</td> </tr> <tr> <td>Optimal Threshold</td> <td>59,2477</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(4).jpg ? Compression GIF: test\output\gif\test5(4).gif</pre>	COMPRESSION RESULTS		Execution Time	879,9418 ms	Original size	12667 bytes	Compressed Size	6294 bytes	Compression Rate	50,3118%	Tree Depth	8	Total Nodes	725	Optimal Threshold	59,2477	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>715,3608 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>2114 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>83,3116%</td> </tr> <tr> <td>Tree Depth</td> <td>1</td> </tr> <tr> <td>Total Nodes</td> <td>5</td> </tr> <tr> <td>Optimal Threshold</td> <td>72,7513</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(5).jpg ? Compression GIF: test\output\gif\test5(5).gif</pre>	COMPRESSION RESULTS		Execution Time	715,3608 ms	Original Size	12667 bytes	Compressed Size	2114 bytes	Compression Rate	83,3116%	Tree Depth	1	Total Nodes	5	Optimal Threshold	72,7513	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>226,5652 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1490 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>88,2372%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(6).jpg ? Compression GIF: test\output\gif\test5(6).gif</pre>	COMPRESSION RESULTS		Execution Time	226,5652 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88,2372%	Tree Depth	0	Total Nodes	1																																																														
COMPRESSION RESULTS																																																																																																														
Execution Time	879,9418 ms																																																																																																													
Original size	12667 bytes																																																																																																													
Compressed Size	6294 bytes																																																																																																													
Compression Rate	50,3118%																																																																																																													
Tree Depth	8																																																																																																													
Total Nodes	725																																																																																																													
Optimal Threshold	59,2477																																																																																																													
COMPRESSION RESULTS																																																																																																														
Execution Time	715,3608 ms																																																																																																													
Original Size	12667 bytes																																																																																																													
Compressed Size	2114 bytes																																																																																																													
Compression Rate	83,3116%																																																																																																													
Tree Depth	1																																																																																																													
Total Nodes	5																																																																																																													
Optimal Threshold	72,7513																																																																																																													
COMPRESSION RESULTS																																																																																																														
Execution Time	226,5652 ms																																																																																																													
Original Size	12667 bytes																																																																																																													
Compressed Size	1490 bytes																																																																																																													
Compression Rate	88,2372%																																																																																																													
Tree Depth	0																																																																																																													
Total Nodes	1																																																																																																													
<p>Input :</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test5.jpg</td> <td>? Valid image found!</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance</td> <td>(0 - 1626,25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 127,5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 9)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td>?? Choose method (1-5): 3</td> <td>? Method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 0</td> <td>? Threshold set!</td> </tr> <tr> <td>? Minimum block size: 1</td> <td>? Minimum block size set to 1</td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0.5</td> <td>? Target set to 50.0%</td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test5(7).jpg</td> <td>? Output path set!</td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y</td> <td>? Do you want to save compression GIF? (y/n): y</td> </tr> <tr> <td>? Enter absolute path to save GIF: test\output\gif\test5(7).gif</td> <td>? Enter absolute path to save GIF: test\output\gif\test5(7).gif</td> </tr> <tr> <td>? GIF path set!</td> <td>? GIF path set!</td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test5.jpg	? Valid image found!	 		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 9)	5. Structural Similarity Index	(0 - 1)	 		?? Choose method (1-5): 3	? Method selected: Max Pixel Difference	? Enter threshold (0-255): 0	? Threshold set!	? Minimum block size: 1	? Minimum block size set to 1	? Target compression percentage (0.0-1.0): 0.5	? Target set to 50.0%	? Absolute path for compressed image: test\output\image\test5(7).jpg	? Output path set!	? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y	? Enter absolute path to save GIF: test\output\gif\test5(7).gif	? Enter absolute path to save GIF: test\output\gif\test5(7).gif	? GIF path set!	? GIF path set!	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test5.jpg</td> <td>? Valid image found!</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance</td> <td>(0 - 1626,25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 127,5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 9)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td>?? Choose method (1-5): 3</td> <td>? Method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 0</td> <td>? Threshold set!</td> </tr> <tr> <td>? Minimum block size: 1</td> <td>? Minimum block size set to 1</td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0.8</td> <td>? Target set to 80.0%</td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test5(8).jpg</td> <td>? Output path set!</td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y</td> <td>? Do you want to save compression GIF? (y/n): y</td> </tr> <tr> <td>? Enter absolute path to save GIF: test\output\gif\test5(8).gif</td> <td>? Enter absolute path to save GIF: test\output\gif\test5(8).gif</td> </tr> <tr> <td>? GIF path set!</td> <td>? GIF path set!</td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test5.jpg	? Valid image found!	 		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 9)	5. Structural Similarity Index	(0 - 1)	 		?? Choose method (1-5): 3	? Method selected: Max Pixel Difference	? Enter threshold (0-255): 0	? Threshold set!	? Minimum block size: 1	? Minimum block size set to 1	? Target compression percentage (0.0-1.0): 0.8	? Target set to 80.0%	? Absolute path for compressed image: test\output\image\test5(8).jpg	? Output path set!	? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y	? Enter absolute path to save GIF: test\output\gif\test5(8).gif	? Enter absolute path to save GIF: test\output\gif\test5(8).gif	? GIF path set!	? GIF path set!	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test5.jpg</td> <td>? Valid image found!</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance</td> <td>(0 - 1626,25)</td> </tr> <tr> <td>2. Mean Absolute Deviation</td> <td>(0 - 127,5)</td> </tr> <tr> <td>3. Max Pixel Difference</td> <td>(0 - 255)</td> </tr> <tr> <td>4. Entropy</td> <td>(0 - 9)</td> </tr> <tr> <td>5. Structural Similarity Index</td> <td>(0 - 1)</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td>?? Choose method (1-5): 3</td> <td>? Method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 255</td> <td>? Threshold set!</td> </tr> <tr> <td>? Minimum block size: 20</td> <td>? Minimum block size set to 20</td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0</td> <td>? Target compression disabled</td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test5(9).jpg</td> <td>? Output path set!</td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y</td> <td>? Do you want to save compression GIF? (y/n): y</td> </tr> <tr> <td>? Enter absolute path to save GIF: test\output\gif\test5(9).gif</td> <td>? Enter absolute path to save GIF: test\output\gif\test5(9).gif</td> </tr> <tr> <td>? GIF path set!</td> <td>? GIF path set!</td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test5.jpg	? Valid image found!	 		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 9)	5. Structural Similarity Index	(0 - 1)	 		?? Choose method (1-5): 3	? Method selected: Max Pixel Difference	? Enter threshold (0-255): 255	? Threshold set!	? Minimum block size: 20	? Minimum block size set to 20	? Target compression percentage (0.0-1.0): 0	? Target compression disabled	? Absolute path for compressed image: test\output\image\test5(9).jpg	? Output path set!	? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y	? Enter absolute path to save GIF: test\output\gif\test5(9).gif	? Enter absolute path to save GIF: test\output\gif\test5(9).gif	? GIF path set!	? GIF path set!
QUADTREE IMAGE COMPRESSION TOOL																																																																																																														
? Absolute path of input image: test\input\test5.jpg	? Valid image found!																																																																																																													
? ERROR MEASUREMENT METHOD																																																																																																														
1. Variance	(0 - 1626,25)																																																																																																													
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																													
3. Max Pixel Difference	(0 - 255)																																																																																																													
4. Entropy	(0 - 9)																																																																																																													
5. Structural Similarity Index	(0 - 1)																																																																																																													
?? Choose method (1-5): 3	? Method selected: Max Pixel Difference																																																																																																													
? Enter threshold (0-255): 0	? Threshold set!																																																																																																													
? Minimum block size: 1	? Minimum block size set to 1																																																																																																													
? Target compression percentage (0.0-1.0): 0.5	? Target set to 50.0%																																																																																																													
? Absolute path for compressed image: test\output\image\test5(7).jpg	? Output path set!																																																																																																													
? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y																																																																																																													
? Enter absolute path to save GIF: test\output\gif\test5(7).gif	? Enter absolute path to save GIF: test\output\gif\test5(7).gif																																																																																																													
? GIF path set!	? GIF path set!																																																																																																													
QUADTREE IMAGE COMPRESSION TOOL																																																																																																														
? Absolute path of input image: test\input\test5.jpg	? Valid image found!																																																																																																													
? ERROR MEASUREMENT METHOD																																																																																																														
1. Variance	(0 - 1626,25)																																																																																																													
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																													
3. Max Pixel Difference	(0 - 255)																																																																																																													
4. Entropy	(0 - 9)																																																																																																													
5. Structural Similarity Index	(0 - 1)																																																																																																													
?? Choose method (1-5): 3	? Method selected: Max Pixel Difference																																																																																																													
? Enter threshold (0-255): 0	? Threshold set!																																																																																																													
? Minimum block size: 1	? Minimum block size set to 1																																																																																																													
? Target compression percentage (0.0-1.0): 0.8	? Target set to 80.0%																																																																																																													
? Absolute path for compressed image: test\output\image\test5(8).jpg	? Output path set!																																																																																																													
? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y																																																																																																													
? Enter absolute path to save GIF: test\output\gif\test5(8).gif	? Enter absolute path to save GIF: test\output\gif\test5(8).gif																																																																																																													
? GIF path set!	? GIF path set!																																																																																																													
QUADTREE IMAGE COMPRESSION TOOL																																																																																																														
? Absolute path of input image: test\input\test5.jpg	? Valid image found!																																																																																																													
? ERROR MEASUREMENT METHOD																																																																																																														
1. Variance	(0 - 1626,25)																																																																																																													
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																													
3. Max Pixel Difference	(0 - 255)																																																																																																													
4. Entropy	(0 - 9)																																																																																																													
5. Structural Similarity Index	(0 - 1)																																																																																																													
?? Choose method (1-5): 3	? Method selected: Max Pixel Difference																																																																																																													
? Enter threshold (0-255): 255	? Threshold set!																																																																																																													
? Minimum block size: 20	? Minimum block size set to 20																																																																																																													
? Target compression percentage (0.0-1.0): 0	? Target compression disabled																																																																																																													
? Absolute path for compressed image: test\output\image\test5(9).jpg	? Output path set!																																																																																																													
? Do you want to save compression GIF? (y/n): y	? Do you want to save compression GIF? (y/n): y																																																																																																													
? Enter absolute path to save GIF: test\output\gif\test5(9).gif	? Enter absolute path to save GIF: test\output\gif\test5(9).gif																																																																																																													
? GIF path set!	? GIF path set!																																																																																																													
<p>Max Pixel Difference</p> 																																																																																																														
<p>Nilai Ambang Batas :</p>	<p>0</p>	<p>0</p>	<p>255</p>																																																																																																											
<p>Output :</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>824,7805 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>6873 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>45,7409%</td> </tr> <tr> <td>Tree Depth</td> <td>8</td> </tr> <tr> <td>Total Nodes</td> <td>673</td> </tr> <tr> <td>Optimal Threshold</td> <td>255,0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(7).jpg ? Compression GIF: test\output\gif\test5(7).gif</pre>	COMPRESSION RESULTS		Execution Time	824,7805 ms	Original Size	12667 bytes	Compressed Size	6873 bytes	Compression Rate	45,7409%	Tree Depth	8	Total Nodes	673	Optimal Threshold	255,0000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>912,1869 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>6873 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>45,7409%</td> </tr> <tr> <td>Tree Depth</td> <td>8</td> </tr> <tr> <td>Total Nodes</td> <td>673</td> </tr> <tr> <td>Optimal Threshold</td> <td>255,0000</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(8).jpg ? Compression GIF: test\output\gif\test5(8).gif</pre>	COMPRESSION RESULTS		Execution Time	912,1869 ms	Original Size	12667 bytes	Compressed Size	6873 bytes	Compression Rate	45,7409%	Tree Depth	8	Total Nodes	673	Optimal Threshold	255,0000	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>219,6406 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1490 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>88,2372%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(9).jpg ? Compression GIF: test\output\gif\test5(9).gif</pre>	COMPRESSION RESULTS		Execution Time	219,6406 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88,2372%	Tree Depth	0	Total Nodes	1																																																														
COMPRESSION RESULTS																																																																																																														
Execution Time	824,7805 ms																																																																																																													
Original Size	12667 bytes																																																																																																													
Compressed Size	6873 bytes																																																																																																													
Compression Rate	45,7409%																																																																																																													
Tree Depth	8																																																																																																													
Total Nodes	673																																																																																																													
Optimal Threshold	255,0000																																																																																																													
COMPRESSION RESULTS																																																																																																														
Execution Time	912,1869 ms																																																																																																													
Original Size	12667 bytes																																																																																																													
Compressed Size	6873 bytes																																																																																																													
Compression Rate	45,7409%																																																																																																													
Tree Depth	8																																																																																																													
Total Nodes	673																																																																																																													
Optimal Threshold	255,0000																																																																																																													
COMPRESSION RESULTS																																																																																																														
Execution Time	219,6406 ms																																																																																																													
Original Size	12667 bytes																																																																																																													
Compressed Size	1490 bytes																																																																																																													
Compression Rate	88,2372%																																																																																																													
Tree Depth	0																																																																																																													
Total Nodes	1																																																																																																													

<p>Input :</p> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <p>? Absolute path for compressed image: test\output\image\test5(10).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(10).gif ? GIF path set!</p>	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <p>? Absolute path of input image: test\input\tests.jpg ? Valid image found!</p> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <p>? Absolute path for compressed image: test\output\image\test5(11).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(11).gif ? GIF path set!</p>	<p>QUADTREE IMAGE COMPRESSION TOOL</p> <p>? Absolute path of input image: test\input\tests.jpg ? Valid image found!</p> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 4 ? Method selected: Entropy ? Enter threshold (0.0-8.0): 8 ? Threshold set! ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test5(12).jpg ? Output path set! <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test5(12).gif ? GIF path set!</p> </pre>																																									
<p>Entropy</p> 																																											
<p>Nilai Ambang Batas :</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 33.33%;">Nilai Ambang Batas :</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">8</td> </tr> </table>	Nilai Ambang Batas :	0	0	8																																							
Nilai Ambang Batas :	0	0	8																																								
<p>Output :</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 33.33%;"> COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>927.2138 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>6311 bytes</td></tr> <tr><td>Compression Rate</td><td>50.1776%</td></tr> <tr><td>Tree Depth</td><td>5</td></tr> <tr><td>Total Nodes</td><td>449</td></tr> <tr><td>Optimal Threshold</td><td>6.4545</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(10).jpg ? Compression GIF: test\output\gif\test5(10).gif</p> </td><td style="text-align: center; vertical-align: top; width: 33.33%;"> COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>637.5510 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> <tr><td>Optimal Threshold</td><td>7.2772</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(11).jpg ? Compression GIF: test\output\gif\test5(11).gif</p> </td><td style="text-align: center; vertical-align: top; width: 33.33%;"> COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>222.6636 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(12).jpg ? Compression GIF: test\output\gif\test5(12).gif</p> </td></tr> </table>	COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>927.2138 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>6311 bytes</td></tr> <tr><td>Compression Rate</td><td>50.1776%</td></tr> <tr><td>Tree Depth</td><td>5</td></tr> <tr><td>Total Nodes</td><td>449</td></tr> <tr><td>Optimal Threshold</td><td>6.4545</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(10).jpg ? Compression GIF: test\output\gif\test5(10).gif</p>	Execution Time	927.2138 ms	Original Size	12667 bytes	Compressed Size	6311 bytes	Compression Rate	50.1776%	Tree Depth	5	Total Nodes	449	Optimal Threshold	6.4545	COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>637.5510 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> <tr><td>Optimal Threshold</td><td>7.2772</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(11).jpg ? Compression GIF: test\output\gif\test5(11).gif</p>	Execution Time	637.5510 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88.2372%	Tree Depth	0	Total Nodes	1	Optimal Threshold	7.2772	COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>222.6636 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(12).jpg ? Compression GIF: test\output\gif\test5(12).gif</p>	Execution Time	222.6636 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88.2372%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>927.2138 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>6311 bytes</td></tr> <tr><td>Compression Rate</td><td>50.1776%</td></tr> <tr><td>Tree Depth</td><td>5</td></tr> <tr><td>Total Nodes</td><td>449</td></tr> <tr><td>Optimal Threshold</td><td>6.4545</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(10).jpg ? Compression GIF: test\output\gif\test5(10).gif</p>	Execution Time	927.2138 ms	Original Size	12667 bytes	Compressed Size	6311 bytes	Compression Rate	50.1776%	Tree Depth	5	Total Nodes	449	Optimal Threshold	6.4545	COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>637.5510 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> <tr><td>Optimal Threshold</td><td>7.2772</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(11).jpg ? Compression GIF: test\output\gif\test5(11).gif</p>	Execution Time	637.5510 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88.2372%	Tree Depth	0	Total Nodes	1	Optimal Threshold	7.2772	COMPRESSION RESULTS <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Execution Time</td><td>222.6636 ms</td></tr> <tr><td>Original Size</td><td>12667 bytes</td></tr> <tr><td>Compressed Size</td><td>1490 bytes</td></tr> <tr><td>Compression Rate</td><td>88.2372%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </table> <p>? Output Files: ? Compressed Image: test\output\image\test5(12).jpg ? Compression GIF: test\output\gif\test5(12).gif</p>	Execution Time	222.6636 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88.2372%	Tree Depth	0	Total Nodes	1	
Execution Time	927.2138 ms																																										
Original Size	12667 bytes																																										
Compressed Size	6311 bytes																																										
Compression Rate	50.1776%																																										
Tree Depth	5																																										
Total Nodes	449																																										
Optimal Threshold	6.4545																																										
Execution Time	637.5510 ms																																										
Original Size	12667 bytes																																										
Compressed Size	1490 bytes																																										
Compression Rate	88.2372%																																										
Tree Depth	0																																										
Total Nodes	1																																										
Optimal Threshold	7.2772																																										
Execution Time	222.6636 ms																																										
Original Size	12667 bytes																																										
Compressed Size	1490 bytes																																										
Compression Rate	88.2372%																																										
Tree Depth	0																																										
Total Nodes	1																																										

Input :

```

? ERROR MEASUREMENT METHOD
1. Variance          (0 - 16256.25)
2. Mean Absolute Deviation (0 - 127.5)
3. Max Pixel Difference (0 - 255)
4. Entropy           (0 - 8)
5. Structural Similarity Index (0 - 1)

?? Choose method (1-5): 5
? Method selected: SSIM

? Enter threshold (0.0-1.0): 0
? Threshold set!

? Minimum block size: 20
? Minimum block size set to 20

? Target compression percentage (0.0-1.0): 0.5
? Target set to 50.0%
```

? Absolute path for compressed image: test\output\image\test5(13).jpg
? Output path set!

? Do you want to save compression GIF? (y/n): y
? Enter absolute path to save GIF: test\output\gif\test5(13).gif
? GIF path set!

QUADTREE IMAGE COMPRESSION TOOL

? Absolute path of input image: test\input\tests.jpg
? Valid image found!

```

? ERROR MEASUREMENT METHOD
1. Variance          (0 - 16256.25)
2. Mean Absolute Deviation (0 - 127.5)
3. Max Pixel Difference (0 - 255)
4. Entropy           (0 - 8)
5. Structural Similarity Index (0 - 1)

?? Choose method (1-5): 5
? Method selected: SSIM

? Enter threshold (0.0-1.0): 0
? Threshold set!

? Minimum block size: 1
? Minimum block size set to 1

? Target compression percentage (0.0-1.0): 0.8
? Target set to 80.0%
```

? Absolute path for compressed image: test\output\image\test5(14).jpg
? Output path set!

? Do you want to save compression GIF? (y/n): y
? Enter absolute path to save GIF: test\output\gif\test5(14).gif
? GIF path set!

QUADTREE IMAGE COMPRESSION TOOL

? Absolute path of input image: test\input\tests.jpg
? Valid image found!

```

? ERROR MEASUREMENT METHOD
1. Variance          (0 - 16256.25)
2. Mean Absolute Deviation (0 - 127.5)
3. Max Pixel Difference (0 - 255)
4. Entropy           (0 - 8)
5. Structural Similarity Index (0 - 1)

?? Choose method (1-5): 5
? Method selected: SSIM

? Enter threshold (0.0-1.0): 1
? Threshold set!

? Minimum block size: 20
? Minimum block size set to 20

? Target compression percentage (0.0-1.0): 0
? Target compression disabled

? Absolute path for compressed image: test\output\image\test5(15).jpg  

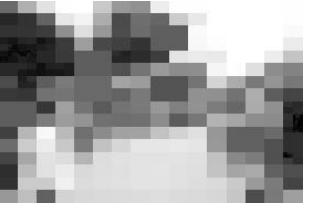
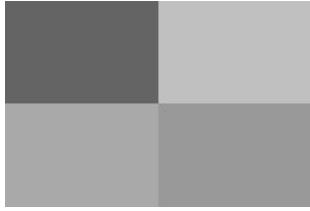
? Output path set!


? Do you want to save compression GIF? (y/n): y  

? Enter absolute path to save GIF: test\output\gif\test5(15).gif  

? GIF path set!


```

Structural Similarity Index																																																	
Nilai Ambang Batas :	0	0	1																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>1492.8821 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>6312 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50.1697%</td> </tr> <tr> <td>Tree Depth</td> <td>8</td> </tr> <tr> <td>Total Nodes</td> <td>641</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.9890</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(13).jpg ? Compression GIF: test\output\gif\test5(13).gif</pre>	COMPRESSION RESULTS		Execution Time	1492.8821 ms	Original Size	12667 bytes	Compressed Size	6312 bytes	Compression Rate	50.1697%	Tree Depth	8	Total Nodes	641	Optimal Threshold	0.9890	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>1093.3868 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>2114 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>83.3110%</td> </tr> <tr> <td>Tree Depth</td> <td>1</td> </tr> <tr> <td>Total Nodes</td> <td>5</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.9922</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(14).jpg ? Compression GIF: test\output\gif\test5(14).gif</pre>	COMPRESSION RESULTS		Execution Time	1093.3868 ms	Original Size	12667 bytes	Compressed Size	2114 bytes	Compression Rate	83.3110%	Tree Depth	1	Total Nodes	5	Optimal Threshold	0.9922	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>242.6334 ms</td> </tr> <tr> <td>Original Size</td> <td>12667 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1490 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>88.2372%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test5(15).jpg ? Compression GIF: test\output\gif\test5(15).gif</pre>	COMPRESSION RESULTS		Execution Time	242.6334 ms	Original Size	12667 bytes	Compressed Size	1490 bytes	Compression Rate	88.2372%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	1492.8821 ms																																																
Original Size	12667 bytes																																																
Compressed Size	6312 bytes																																																
Compression Rate	50.1697%																																																
Tree Depth	8																																																
Total Nodes	641																																																
Optimal Threshold	0.9890																																																
COMPRESSION RESULTS																																																	
Execution Time	1093.3868 ms																																																
Original Size	12667 bytes																																																
Compressed Size	2114 bytes																																																
Compression Rate	83.3110%																																																
Tree Depth	1																																																
Total Nodes	5																																																
Optimal Threshold	0.9922																																																
COMPRESSION RESULTS																																																	
Execution Time	242.6334 ms																																																
Original Size	12667 bytes																																																
Compressed Size	1490 bytes																																																
Compression Rate	88.2372%																																																
Tree Depth	0																																																
Total Nodes	1																																																

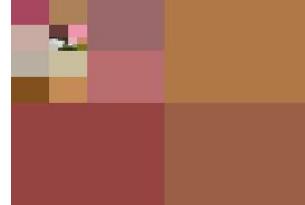
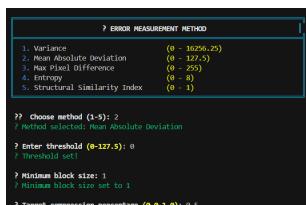
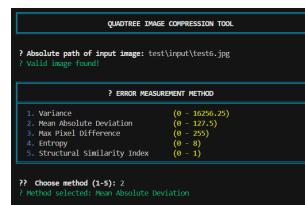
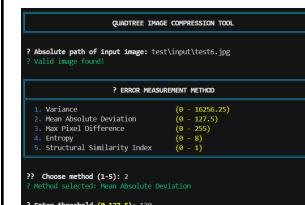
4.1.6 Kasus Eksperimen 6

Gambar Asli :

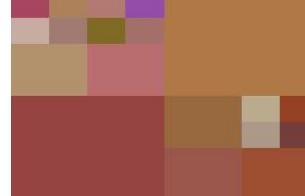
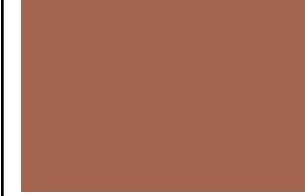


Hasil Kompresi :

Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%

<p>Input :</p> <pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test6.jpg ? Valid Image Found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>?? Choose method (1-5): 1 Method selected: Variance</p> <p>? Enter threshold (0-1626.25): 0 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</p> <p>? Absolute path for compressed image: test\output\image\test6(1).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test6(1).gif ? GIF path set!</p>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test6.jpg ? Valid Image Found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>?? Choose method (1-5): 1 Method selected: Variance</p> <p>? Enter threshold (0-1626.25): 0 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</p> <p>? Absolute path for compressed image: test\output\image\test6(2).jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test6(2).gif ? GIF path set!</p>	<pre>QUOTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test6.jpg ? Valid Image Found!</pre> <p>? ERROR MEASUREMENT METHOD</p> <ol style="list-style-type: none"> 1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) <p>?? Choose method (1-5): 1 Method selected: Variance</p> <p>? Enter threshold (0-1626.25): 1000 ? Threshold set!</p> <p>? Minimum block size: 20 ? Minimum block size set to 20</p> <p>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled</p> <p>? Absolute path for compressed image: test\output\image\test6().jpg ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y</p> <p>? Enter absolute path to save GIF: test\output\gif\test6().gif ? GIF path set!</p>																																														
<p>Variansi</p>   																																																
<p>Nilai Ambang Batas :</p> <table style="width: 100%; text-align: center;"> <tr> <td>0</td> <td>0</td> <td>1000</td> </tr> </table>	0	0	1000																																													
0	0	1000																																														
<p>Output :</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>938.9523 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>7122 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>49.9684%</td> </tr> <tr> <td>Tree Depth</td> <td>8</td> </tr> <tr> <td>Total Nodes</td> <td>749</td> </tr> <tr> <td>Optimal Threshold</td> <td>3248.7883</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6(1).jpg ? Compression GIF: test\output\gif\test6(1).gif</p>	COMPRESSION RESULTS		Execution Time	938.9523 ms	Original Size	14235 bytes	Compressed Size	7122 bytes	Compression Rate	49.9684%	Tree Depth	8	Total Nodes	749	Optimal Threshold	3248.7883	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>850.8522 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>2694 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>81.0748%</td> </tr> <tr> <td>Tree Depth</td> <td>7</td> </tr> <tr> <td>Total Nodes</td> <td>61</td> </tr> <tr> <td>Optimal Threshold</td> <td>4529.7544</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6(2).jpg ? Compression GIF: test\output\gif\test6(2).gif</p>	COMPRESSION RESULTS		Execution Time	850.8522 ms	Original Size	14235 bytes	Compressed Size	2694 bytes	Compression Rate	81.0748%	Tree Depth	7	Total Nodes	61	Optimal Threshold	4529.7544	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>415.7174 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>11133 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>21.7914%</td> </tr> <tr> <td>Tree Depth</td> <td>5</td> </tr> <tr> <td>Total Nodes</td> <td>1109</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6().jpg ? Compression GIF: test\output\gif\test6().gif</p>	COMPRESSION RESULTS		Execution Time	415.7174 ms	Original Size	14235 bytes	Compressed Size	11133 bytes	Compression Rate	21.7914%	Tree Depth	5	Total Nodes	1109
COMPRESSION RESULTS																																																
Execution Time	938.9523 ms																																															
Original Size	14235 bytes																																															
Compressed Size	7122 bytes																																															
Compression Rate	49.9684%																																															
Tree Depth	8																																															
Total Nodes	749																																															
Optimal Threshold	3248.7883																																															
COMPRESSION RESULTS																																																
Execution Time	850.8522 ms																																															
Original Size	14235 bytes																																															
Compressed Size	2694 bytes																																															
Compression Rate	81.0748%																																															
Tree Depth	7																																															
Total Nodes	61																																															
Optimal Threshold	4529.7544																																															
COMPRESSION RESULTS																																																
Execution Time	415.7174 ms																																															
Original Size	14235 bytes																																															
Compressed Size	11133 bytes																																															
Compression Rate	21.7914%																																															
Tree Depth	5																																															
Total Nodes	1109																																															
<p>Input :</p>   																																																

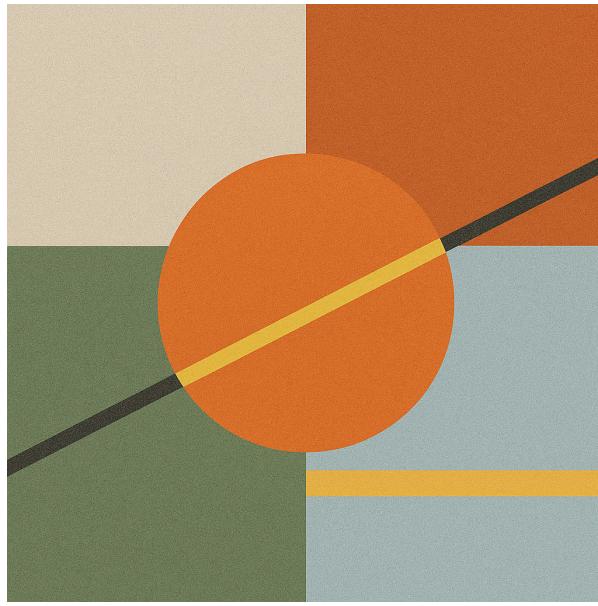
Mean Absolute Deviation																																																	
Nilai Ambang Batas :	0	0	120																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>882.2727 ms</td></tr> <tr> <td>Original Size</td><td>14235 bytes</td></tr> <tr> <td>Compressed Size</td><td>7132 bytes</td></tr> <tr> <td>Compression Rate</td><td>49.8981%</td></tr> <tr> <td>Tree Depth</td><td>8</td></tr> <tr> <td>Total Nodes</td><td>769</td></tr> <tr> <td>Optimal Threshold</td><td>46.1424</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6(4).jpg ? Compression GIF: test\output\gif\test6(4).gif</p>	COMPRESSION RESULTS		Execution Time	882.2727 ms	Original Size	14235 bytes	Compressed Size	7132 bytes	Compression Rate	49.8981%	Tree Depth	8	Total Nodes	769	Optimal Threshold	46.1424	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>800.5058 ms</td></tr> <tr> <td>Original Size</td><td>14235 bytes</td></tr> <tr> <td>Compressed Size</td><td>3082 bytes</td></tr> <tr> <td>Compression Rate</td><td>78.3401%</td></tr> <tr> <td>Tree Depth</td><td>8</td></tr> <tr> <td>Total Nodes</td><td>73</td></tr> <tr> <td>Optimal Threshold</td><td>57.3962</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6(5).jpg ? Compression GIF: test\output\gif\test6(5).gif</p>	COMPRESSION RESULTS		Execution Time	800.5058 ms	Original Size	14235 bytes	Compressed Size	3082 bytes	Compression Rate	78.3401%	Tree Depth	8	Total Nodes	73	Optimal Threshold	57.3962	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>238.9038 ms</td></tr> <tr> <td>Original Size</td><td>14235 bytes</td></tr> <tr> <td>Compressed Size</td><td>1492 bytes</td></tr> <tr> <td>Compression Rate</td><td>89.5188%</td></tr> <tr> <td>Tree Depth</td><td>0</td></tr> <tr> <td>Total Nodes</td><td>1</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test6(6).jpg ? Compression GIF: test\output\gif\test6(6).gif</p>	COMPRESSION RESULTS		Execution Time	238.9038 ms	Original Size	14235 bytes	Compressed Size	1492 bytes	Compression Rate	89.5188%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	882.2727 ms																																																
Original Size	14235 bytes																																																
Compressed Size	7132 bytes																																																
Compression Rate	49.8981%																																																
Tree Depth	8																																																
Total Nodes	769																																																
Optimal Threshold	46.1424																																																
COMPRESSION RESULTS																																																	
Execution Time	800.5058 ms																																																
Original Size	14235 bytes																																																
Compressed Size	3082 bytes																																																
Compression Rate	78.3401%																																																
Tree Depth	8																																																
Total Nodes	73																																																
Optimal Threshold	57.3962																																																
COMPRESSION RESULTS																																																	
Execution Time	238.9038 ms																																																
Original Size	14235 bytes																																																
Compressed Size	1492 bytes																																																
Compression Rate	89.5188%																																																
Tree Depth	0																																																
Total Nodes	1																																																
Input :																																																	
Max Pixel Difference																																																	
Nilai Ambang Batas :	0	0	255																																														

Output :	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>870,9386 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>7062 bytes</td></tr> <tr><td>Compression Rate</td><td>50,3899%</td></tr> <tr><td>Tree Depth</td><td>7</td></tr> <tr><td>Total Nodes</td><td>353</td></tr> <tr><td>Optimal Threshold</td><td>227.0000</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(7).jpg ? Compression GIF: test\output\gif\test6(7).gif</pre>	COMPRESSION RESULTS		Execution Time	870,9386 ms	Original Size	14235 bytes	Compressed Size	7062 bytes	Compression Rate	50,3899%	Tree Depth	7	Total Nodes	353	Optimal Threshold	227.0000	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>722,3441 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>2425 bytes</td></tr> <tr><td>Compression Rate</td><td>82,9645%</td></tr> <tr><td>Tree Depth</td><td>4</td></tr> <tr><td>Total Nodes</td><td>17</td></tr> <tr><td>Optimal Threshold</td><td>254.3333</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(8).jpg ? Compression GIF: test\output\gif\test6(8).gif</pre>	COMPRESSION RESULTS		Execution Time	722,3441 ms	Original Size	14235 bytes	Compressed Size	2425 bytes	Compression Rate	82,9645%	Tree Depth	4	Total Nodes	17	Optimal Threshold	254.3333	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>296,6204 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>1492 bytes</td></tr> <tr><td>Compression Rate</td><td>89,5188%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(9).jpg ? Compression GIF: test\output\gif\test6(9).gif</pre>	COMPRESSION RESULTS		Execution Time	296,6204 ms	Original Size	14235 bytes	Compressed Size	1492 bytes	Compression Rate	89,5188%	Tree Depth	0	Total Nodes	1																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	870,9386 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	7062 bytes																																																																																																																																						
Compression Rate	50,3899%																																																																																																																																						
Tree Depth	7																																																																																																																																						
Total Nodes	353																																																																																																																																						
Optimal Threshold	227.0000																																																																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	722,3441 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	2425 bytes																																																																																																																																						
Compression Rate	82,9645%																																																																																																																																						
Tree Depth	4																																																																																																																																						
Total Nodes	17																																																																																																																																						
Optimal Threshold	254.3333																																																																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	296,6204 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	1492 bytes																																																																																																																																						
Compression Rate	89,5188%																																																																																																																																						
Tree Depth	0																																																																																																																																						
Total Nodes	1																																																																																																																																						
Input :	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th></tr> </thead> <tbody> <tr><td>? Absolute path of input image:</td><td>test\input\test6.jpg</td></tr> <tr><td>? Valid image found!</td><td></td></tr> <tr><td>? ERROR MEASUREMENT METHOD</td><td></td></tr> <tr><td>1. Variance</td><td>(0 - 1626,25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 255)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> <tr><td>?? Choose method (1-5): 4</td><td></td></tr> <tr><td>? Method selected: Entropy</td><td></td></tr> <tr><td>? Enter threshold (0.0-1.0): 0</td><td></td></tr> <tr><td>? Threshold set!</td><td></td></tr> <tr><td>? Minimum block size: 1</td><td></td></tr> <tr><td>? Minimum block size set to 1</td><td></td></tr> <tr><td>? Target compression percentage (0.0-1.0): 0.5</td><td></td></tr> <tr><td>? Target set to 50.0%</td><td></td></tr> <tr><td>? Absolute path for compressed image: test\output\image\test6(10).jpg</td><td></td></tr> <tr><td>? Output path set!</td><td></td></tr> <tr><td>? Do you want to save compression GIF? (y/n): y</td><td></td></tr> <tr><td>? Enter absolute path to save GIF: test\output\gif\test6(10).gif</td><td></td></tr> <tr><td>? GIF path set!</td><td></td></tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image:	test\input\test6.jpg	? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 255)	5. Structural Similarity Index	(0 - 1)	?? Choose method (1-5): 4		? Method selected: Entropy		? Enter threshold (0.0-1.0): 0		? Threshold set!		? Minimum block size: 1		? Minimum block size set to 1		? Target compression percentage (0.0-1.0): 0.5		? Target set to 50.0%		? Absolute path for compressed image: test\output\image\test6(10).jpg		? Output path set!		? Do you want to save compression GIF? (y/n): y		? Enter absolute path to save GIF: test\output\gif\test6(10).gif		? GIF path set!		<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th></tr> </thead> <tbody> <tr><td>? Absolute path of input image:</td><td>test\input\test6.jpg</td></tr> <tr><td>? Valid image found!</td><td></td></tr> <tr><td>? ERROR MEASUREMENT METHOD</td><td></td></tr> <tr><td>1. Variance</td><td>(0 - 1626,25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 255)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> <tr><td>?? Choose method (1-5): 4</td><td></td></tr> <tr><td>? Method selected: Entropy</td><td></td></tr> <tr><td>? Enter threshold (0.0-1.0): 0</td><td></td></tr> <tr><td>? Threshold set!</td><td></td></tr> <tr><td>? Minimum block size: 1</td><td></td></tr> <tr><td>? Minimum block size set to 1</td><td></td></tr> <tr><td>? Target compression percentage (0.0-1.0): 0.8</td><td></td></tr> <tr><td>? Target set to 80.0%</td><td></td></tr> <tr><td>? Absolute path for compressed image: test\output\image\test6(11).jpg</td><td></td></tr> <tr><td>? Output path set!</td><td></td></tr> <tr><td>? Do you want to save compression GIF? (y/n): y</td><td></td></tr> <tr><td>? Enter absolute path to save GIF: test\output\gif\test6(11).gif</td><td></td></tr> <tr><td>? GIF path set!</td><td></td></tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image:	test\input\test6.jpg	? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 255)	5. Structural Similarity Index	(0 - 1)	?? Choose method (1-5): 4		? Method selected: Entropy		? Enter threshold (0.0-1.0): 0		? Threshold set!		? Minimum block size: 1		? Minimum block size set to 1		? Target compression percentage (0.0-1.0): 0.8		? Target set to 80.0%		? Absolute path for compressed image: test\output\image\test6(11).jpg		? Output path set!		? Do you want to save compression GIF? (y/n): y		? Enter absolute path to save GIF: test\output\gif\test6(11).gif		? GIF path set!		<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th></tr> </thead> <tbody> <tr><td>? Absolute path of input image:</td><td>test\input\test6.jpg</td></tr> <tr><td>? Valid image found!</td><td></td></tr> <tr><td>? ERROR MEASUREMENT METHOD</td><td></td></tr> <tr><td>1. Variance</td><td>(0 - 1626,25)</td></tr> <tr><td>2. Mean Absolute Deviation</td><td>(0 - 127,5)</td></tr> <tr><td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr><td>4. Entropy</td><td>(0 - 255)</td></tr> <tr><td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> <tr><td>?? Choose method (1-5): 4</td><td></td></tr> <tr><td>? Method selected: Entropy</td><td></td></tr> <tr><td>? Enter threshold (0.0-1.0): 0</td><td></td></tr> <tr><td>? Threshold set!</td><td></td></tr> <tr><td>? Minimum block size: 20</td><td></td></tr> <tr><td>? Minimum block size set to 20</td><td></td></tr> <tr><td>? Target compression percentage (0.0-1.0): 0</td><td></td></tr> <tr><td>? Target compression disabled</td><td></td></tr> <tr><td>? Absolute path for compressed image: test\output\image\test6(12).jpg</td><td></td></tr> <tr><td>? Output path set!</td><td></td></tr> <tr><td>? Do you want to save compression GIF? (y/n): y</td><td></td></tr> <tr><td>? Enter absolute path to save GIF: test\output\gif\test6(12).gif</td><td></td></tr> <tr><td>? GIF path set!</td><td></td></tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image:	test\input\test6.jpg	? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance	(0 - 1626,25)	2. Mean Absolute Deviation	(0 - 127,5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 255)	5. Structural Similarity Index	(0 - 1)	?? Choose method (1-5): 4		? Method selected: Entropy		? Enter threshold (0.0-1.0): 0		? Threshold set!		? Minimum block size: 20		? Minimum block size set to 20		? Target compression percentage (0.0-1.0): 0		? Target compression disabled		? Absolute path for compressed image: test\output\image\test6(12).jpg		? Output path set!		? Do you want to save compression GIF? (y/n): y		? Enter absolute path to save GIF: test\output\gif\test6(12).gif		? GIF path set!	
QUADTREE IMAGE COMPRESSION TOOL																																																																																																																																							
? Absolute path of input image:	test\input\test6.jpg																																																																																																																																						
? Valid image found!																																																																																																																																							
? ERROR MEASUREMENT METHOD																																																																																																																																							
1. Variance	(0 - 1626,25)																																																																																																																																						
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																																																						
3. Max Pixel Difference	(0 - 255)																																																																																																																																						
4. Entropy	(0 - 255)																																																																																																																																						
5. Structural Similarity Index	(0 - 1)																																																																																																																																						
?? Choose method (1-5): 4																																																																																																																																							
? Method selected: Entropy																																																																																																																																							
? Enter threshold (0.0-1.0): 0																																																																																																																																							
? Threshold set!																																																																																																																																							
? Minimum block size: 1																																																																																																																																							
? Minimum block size set to 1																																																																																																																																							
? Target compression percentage (0.0-1.0): 0.5																																																																																																																																							
? Target set to 50.0%																																																																																																																																							
? Absolute path for compressed image: test\output\image\test6(10).jpg																																																																																																																																							
? Output path set!																																																																																																																																							
? Do you want to save compression GIF? (y/n): y																																																																																																																																							
? Enter absolute path to save GIF: test\output\gif\test6(10).gif																																																																																																																																							
? GIF path set!																																																																																																																																							
QUADTREE IMAGE COMPRESSION TOOL																																																																																																																																							
? Absolute path of input image:	test\input\test6.jpg																																																																																																																																						
? Valid image found!																																																																																																																																							
? ERROR MEASUREMENT METHOD																																																																																																																																							
1. Variance	(0 - 1626,25)																																																																																																																																						
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																																																						
3. Max Pixel Difference	(0 - 255)																																																																																																																																						
4. Entropy	(0 - 255)																																																																																																																																						
5. Structural Similarity Index	(0 - 1)																																																																																																																																						
?? Choose method (1-5): 4																																																																																																																																							
? Method selected: Entropy																																																																																																																																							
? Enter threshold (0.0-1.0): 0																																																																																																																																							
? Threshold set!																																																																																																																																							
? Minimum block size: 1																																																																																																																																							
? Minimum block size set to 1																																																																																																																																							
? Target compression percentage (0.0-1.0): 0.8																																																																																																																																							
? Target set to 80.0%																																																																																																																																							
? Absolute path for compressed image: test\output\image\test6(11).jpg																																																																																																																																							
? Output path set!																																																																																																																																							
? Do you want to save compression GIF? (y/n): y																																																																																																																																							
? Enter absolute path to save GIF: test\output\gif\test6(11).gif																																																																																																																																							
? GIF path set!																																																																																																																																							
QUADTREE IMAGE COMPRESSION TOOL																																																																																																																																							
? Absolute path of input image:	test\input\test6.jpg																																																																																																																																						
? Valid image found!																																																																																																																																							
? ERROR MEASUREMENT METHOD																																																																																																																																							
1. Variance	(0 - 1626,25)																																																																																																																																						
2. Mean Absolute Deviation	(0 - 127,5)																																																																																																																																						
3. Max Pixel Difference	(0 - 255)																																																																																																																																						
4. Entropy	(0 - 255)																																																																																																																																						
5. Structural Similarity Index	(0 - 1)																																																																																																																																						
?? Choose method (1-5): 4																																																																																																																																							
? Method selected: Entropy																																																																																																																																							
? Enter threshold (0.0-1.0): 0																																																																																																																																							
? Threshold set!																																																																																																																																							
? Minimum block size: 20																																																																																																																																							
? Minimum block size set to 20																																																																																																																																							
? Target compression percentage (0.0-1.0): 0																																																																																																																																							
? Target compression disabled																																																																																																																																							
? Absolute path for compressed image: test\output\image\test6(12).jpg																																																																																																																																							
? Output path set!																																																																																																																																							
? Do you want to save compression GIF? (y/n): y																																																																																																																																							
? Enter absolute path to save GIF: test\output\gif\test6(12).gif																																																																																																																																							
? GIF path set!																																																																																																																																							
Entropy																																																																																																																																							
Nilai Ambang Batas :	0	0	8																																																																																																																																				
Output :	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>978,2289 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>7116 bytes</td></tr> <tr><td>Compression Rate</td><td>50,0105%</td></tr> <tr><td>Tree Depth</td><td>5</td></tr> <tr><td>Total Nodes</td><td>257</td></tr> <tr><td>Optimal Threshold</td><td>6.7727</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(10).jpg ? Compression GIF: test\output\gif\test6(10).gif</pre>	COMPRESSION RESULTS		Execution Time	978,2289 ms	Original Size	14235 bytes	Compressed Size	7116 bytes	Compression Rate	50,0105%	Tree Depth	5	Total Nodes	257	Optimal Threshold	6.7727	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>740,9645 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>2888 bytes</td></tr> <tr><td>Compression Rate</td><td>79,7120%</td></tr> <tr><td>Tree Depth</td><td>3</td></tr> <tr><td>Total Nodes</td><td>25</td></tr> <tr><td>Optimal Threshold</td><td>7.5924</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(11).jpg ? Compression GIF: test\output\gif\test6(11).gif</pre>	COMPRESSION RESULTS		Execution Time	740,9645 ms	Original Size	14235 bytes	Compressed Size	2888 bytes	Compression Rate	79,7120%	Tree Depth	3	Total Nodes	25	Optimal Threshold	7.5924	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr><th colspan="2">COMPRESSION RESULTS</th></tr> </thead> <tbody> <tr><td>Execution Time</td><td>229,9858 ms</td></tr> <tr><td>Original Size</td><td>14235 bytes</td></tr> <tr><td>Compressed Size</td><td>1492 bytes</td></tr> <tr><td>Compression Rate</td><td>89,5188%</td></tr> <tr><td>Tree Depth</td><td>0</td></tr> <tr><td>Total Nodes</td><td>1</td></tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test6(12).jpg ? Compression GIF: test\output\gif\test6(12).gif</pre>	COMPRESSION RESULTS		Execution Time	229,9858 ms	Original Size	14235 bytes	Compressed Size	1492 bytes	Compression Rate	89,5188%	Tree Depth	0	Total Nodes	1																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	978,2289 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	7116 bytes																																																																																																																																						
Compression Rate	50,0105%																																																																																																																																						
Tree Depth	5																																																																																																																																						
Total Nodes	257																																																																																																																																						
Optimal Threshold	6.7727																																																																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	740,9645 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	2888 bytes																																																																																																																																						
Compression Rate	79,7120%																																																																																																																																						
Tree Depth	3																																																																																																																																						
Total Nodes	25																																																																																																																																						
Optimal Threshold	7.5924																																																																																																																																						
COMPRESSION RESULTS																																																																																																																																							
Execution Time	229,9858 ms																																																																																																																																						
Original Size	14235 bytes																																																																																																																																						
Compressed Size	1492 bytes																																																																																																																																						
Compression Rate	89,5188%																																																																																																																																						
Tree Depth	0																																																																																																																																						
Total Nodes	1																																																																																																																																						

<p>Input :</p> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 5 ? Method selected: SIMD ? Enter threshold (0.0-1.0): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</pre> <pre> ? Absolute path of input image: test\input\test6.jpg ? Valid image found!</pre> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 5 ? Method selected: SIMD ? Enter threshold (0.0-1.0): 0 ? Threshold set ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre> ? Absolute path for compressed image: test\output\image\test6(11).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test6(11).gif ? GIF path set!</pre>	<pre> ? QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test6.jpg ? Valid image found!</pre> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 5 ? Method selected: SIMD ? Enter threshold (0.0-1.0): 1 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre> ? Absolute path for compressed image: test\output\image\test6(14).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test6(14).gif ? GIF path set!</pre>	<pre> ? QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test6.jpg ? Valid image found!</pre> <pre> ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 5 ? Method selected: SIMD ? Enter threshold (0.0-1.0): 1 ? Threshold set ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</pre> <pre> ? Absolute path for compressed image: test\output\image\test6(15).jpg ? Output path set ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test6(15).gif ? GIF path set!</pre>																																														
<p>Structural Similarity Index</p>																																																
<p>Nilai Ambang Batas :</p>	<p>0</p>	<p>0</p>																																														
<p>Output :</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>1532.6249 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>7110 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50.0527%</td> </tr> <tr> <td>Tree Depth</td> <td>5</td> </tr> <tr> <td>Total Nodes</td> <td>269</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.9706</td> </tr> </tbody> </table> <pre> ? Output Files: ? Compressed Image: test\output\image\test6(13).jpg ? Compression GIF: test\output\gif\test6(13).gif</pre>	COMPRESSION RESULTS		Execution Time	1532.6249 ms	Original Size	14235 bytes	Compressed Size	7110 bytes	Compression Rate	50.0527%	Tree Depth	5	Total Nodes	269	Optimal Threshold	0.9706	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>1270.5206 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>3006 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>78.8830%</td> </tr> <tr> <td>Tree Depth</td> <td>3</td> </tr> <tr> <td>Total Nodes</td> <td>37</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.9867</td> </tr> </tbody> </table> <pre> ? Output Files: ? Compressed Image: test\output\image\test6(14).jpg ? Compression GIF: test\output\gif\test6(14).gif</pre>	COMPRESSION RESULTS		Execution Time	1270.5206 ms	Original Size	14235 bytes	Compressed Size	3006 bytes	Compression Rate	78.8830%	Tree Depth	3	Total Nodes	37	Optimal Threshold	0.9867	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>298.9619 ms</td> </tr> <tr> <td>Original Size</td> <td>14235 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1492 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>89.5188%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre> ? Output Files: ? Compressed Image: test\output\image\test6(15).jpg ? Compression GIF: test\output\gif\test6(15).gif</pre>	COMPRESSION RESULTS		Execution Time	298.9619 ms	Original Size	14235 bytes	Compressed Size	1492 bytes	Compression Rate	89.5188%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																
Execution Time	1532.6249 ms																																															
Original Size	14235 bytes																																															
Compressed Size	7110 bytes																																															
Compression Rate	50.0527%																																															
Tree Depth	5																																															
Total Nodes	269																																															
Optimal Threshold	0.9706																																															
COMPRESSION RESULTS																																																
Execution Time	1270.5206 ms																																															
Original Size	14235 bytes																																															
Compressed Size	3006 bytes																																															
Compression Rate	78.8830%																																															
Tree Depth	3																																															
Total Nodes	37																																															
Optimal Threshold	0.9867																																															
COMPRESSION RESULTS																																																
Execution Time	298.9619 ms																																															
Original Size	14235 bytes																																															
Compressed Size	1492 bytes																																															
Compression Rate	89.5188%																																															
Tree Depth	0																																															
Total Nodes	1																																															

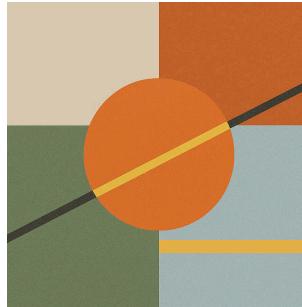
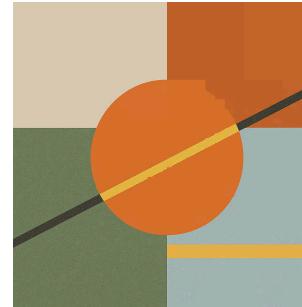
4.1.7 Kasus Eksperimen 7

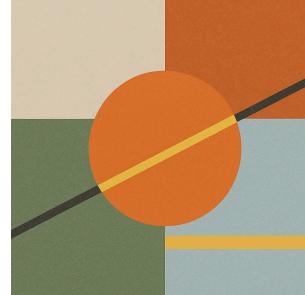
Gambar Asli :

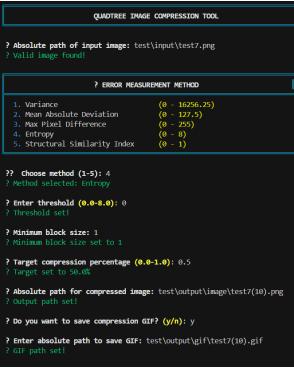
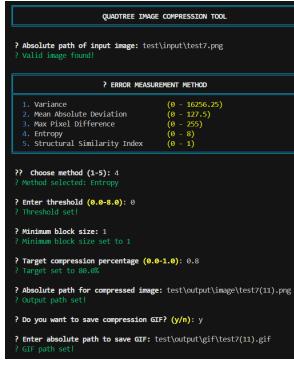
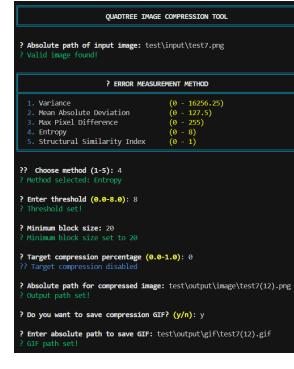
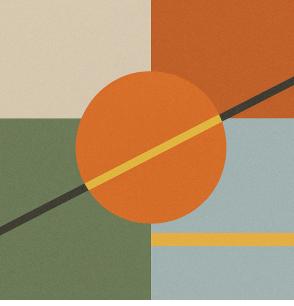
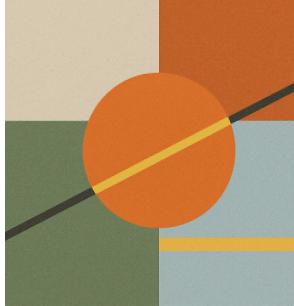
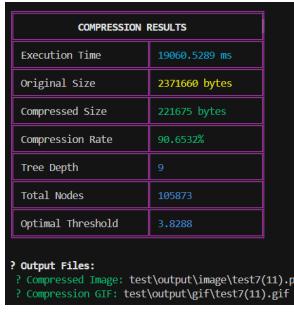
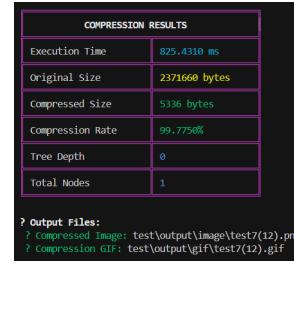
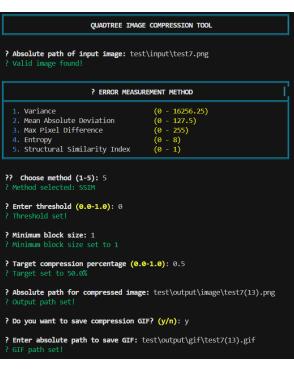
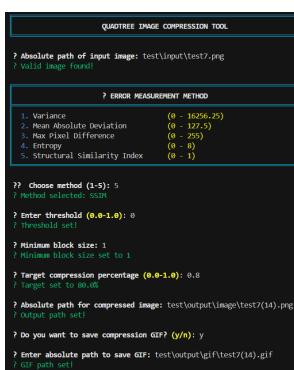
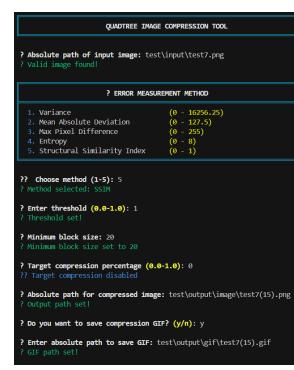


Hasil Kompresi :

Metode	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 0	Ukuran Blok Minimum : 20
	Target Kompresi : 50%	Target Kompresi : 80%	Target Kompresi : 0%
Input :	<pre>? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.5 ? Target set to 0.5 ? Absolute path for compressed image: test\output\image\test7(1).png ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(1).gif ? GIF path set!</pre>	<pre>? Absolute path of input image: test\input\test7.png ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 0 ? Threshold set! ? Minimum block size: 1 ? Minimum block size set to 1 ? Target compression percentage (0.0-1.0): 0.8 ? Target set to 0.8 ? Absolute path for compressed image: test\output\image\test7(2).png ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(2).gif ? GIF path set!</pre>	<pre>? QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test7.png ? Valid image found ? ERROR MEASUREMENT METHOD 1. Variance (0 - 16256.25) 2. Mean Absolute Deviation (0 - 327.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1) ?? Choose method (1-5): 1 ? Method selected: Variance ? Enter threshold (0-16256.25): 1000 ? threshold set! ? Minimum block size: 20 ? Minimum block size set to 20 ? Target compression percentage (0.0-1.0): 0 ? Target compression disabled ? Absolute path for compressed image: test\output\image\test7(3).png ? Output path set! ? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(3).gif ? GIF path set!</pre>
Variansi			
Nilai	0	0	1000

Ambang Batas :																																																	
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>15412.4121 ms</td></tr> <tr> <td>Original Size</td><td>2371660 bytes</td></tr> <tr> <td>Compressed Size</td><td>1185975 bytes</td></tr> <tr> <td>Compression Rate</td><td>49.993%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>519385</td></tr> <tr> <td>Optimal Threshold</td><td>74.1250</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(1).png ? Compression GIF: test\output\gif\test7(1).gif</p>	COMPRESSION RESULTS		Execution Time	15412.4121 ms	Original Size	2371660 bytes	Compressed Size	1185975 bytes	Compression Rate	49.993%	Tree Depth	10	Total Nodes	519385	Optimal Threshold	74.1250	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>11603.6757 ms</td></tr> <tr> <td>Original Size</td><td>2371660 bytes</td></tr> <tr> <td>Compressed Size</td><td>473943 bytes</td></tr> <tr> <td>Compression Rate</td><td>89.0104%</td></tr> <tr> <td>Tree Depth</td><td>10</td></tr> <tr> <td>Total Nodes</td><td>190305</td></tr> <tr> <td>Optimal Threshold</td><td>104.7656</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(2).png ? Compression GIF: test\output\gif\test7(2).gif</p>	COMPRESSION RESULTS		Execution Time	11603.6757 ms	Original Size	2371660 bytes	Compressed Size	473943 bytes	Compression Rate	89.0104%	Tree Depth	10	Total Nodes	190305	Optimal Threshold	104.7656	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td><td>2481.4069 ms</td></tr> <tr> <td>Original Size</td><td>2371660 bytes</td></tr> <tr> <td>Compressed Size</td><td>18628 bytes</td></tr> <tr> <td>Compression Rate</td><td>99.2146%</td></tr> <tr> <td>Tree Depth</td><td>7</td></tr> <tr> <td>Total Nodes</td><td>285</td></tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(3).png ? Compression GIF: test\output\gif\test7(3).gif</p>	COMPRESSION RESULTS		Execution Time	2481.4069 ms	Original Size	2371660 bytes	Compressed Size	18628 bytes	Compression Rate	99.2146%	Tree Depth	7	Total Nodes	285
COMPRESSION RESULTS																																																	
Execution Time	15412.4121 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	1185975 bytes																																																
Compression Rate	49.993%																																																
Tree Depth	10																																																
Total Nodes	519385																																																
Optimal Threshold	74.1250																																																
COMPRESSION RESULTS																																																	
Execution Time	11603.6757 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	473943 bytes																																																
Compression Rate	89.0104%																																																
Tree Depth	10																																																
Total Nodes	190305																																																
Optimal Threshold	104.7656																																																
COMPRESSION RESULTS																																																	
Execution Time	2481.4069 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	18628 bytes																																																
Compression Rate	99.2146%																																																
Tree Depth	7																																																
Total Nodes	285																																																
Input :	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test7.png ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td><td>(0 - 16256.25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127.5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <p>? Choose method (1-5): ? ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 0 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</p> <p>? Absolute path for compressed image: test\output\image\test7(4).png ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(4).gif ? GIF path set!</p>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256.25)	2. Mean Absolute Deviation	(0 - 127.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test7.png ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td><td>(0 - 16256.25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127.5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <p>? Choose method (1-5): ? ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 0 ? Threshold set!</p> <p>? Minimum block size: 1 ? Minimum block size set to 1</p> <p>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</p> <p>? Absolute path for compressed image: test\output\image\test7(5).png ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(5).gif ? GIF path set!</p>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256.25)	2. Mean Absolute Deviation	(0 - 127.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)	<pre>QUADTREE IMAGE COMPRESSION TOOL ? Absolute path of input image: test\input\test7.png ? Valid image found!</pre> <table border="1"> <thead> <tr> <th colspan="2">? ERROR MEASUREMENT METHOD</th> </tr> </thead> <tbody> <tr> <td>1. Variance</td><td>(0 - 16256.25)</td></tr> <tr> <td>2. Mean Absolute Deviation</td><td>(0 - 127.5)</td></tr> <tr> <td>3. Max Pixel Difference</td><td>(0 - 255)</td></tr> <tr> <td>4. Entropy</td><td>(0 - 8)</td></tr> <tr> <td>5. Structural Similarity Index</td><td>(0 - 1)</td></tr> </tbody> </table> <p>? Choose method (1-5): ? ? Method selected: Mean Absolute Deviation</p> <p>? Enter threshold (0-127.5): 128 ? Threshold set!</p> <p>? Minimum block size: 20 ? Minimum block size set to 20</p> <p>? Target compression percentage (0.0-1.0): 0 ? Target compression disabled</p> <p>? Absolute path for compressed image: test\output\image\test7(6).png ? Output path set!</p> <p>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(6).gif ? GIF path set!</p>	? ERROR MEASUREMENT METHOD		1. Variance	(0 - 16256.25)	2. Mean Absolute Deviation	(0 - 127.5)	3. Max Pixel Difference	(0 - 255)	4. Entropy	(0 - 8)	5. Structural Similarity Index	(0 - 1)										
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256.25)																																																
2. Mean Absolute Deviation	(0 - 127.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256.25)																																																
2. Mean Absolute Deviation	(0 - 127.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
? ERROR MEASUREMENT METHOD																																																	
1. Variance	(0 - 16256.25)																																																
2. Mean Absolute Deviation	(0 - 127.5)																																																
3. Max Pixel Difference	(0 - 255)																																																
4. Entropy	(0 - 8)																																																
5. Structural Similarity Index	(0 - 1)																																																
Mean Absolute Deviation																																																	
Nilai Ambang Batas :	0	0	120																																														

Output :	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>8888.7424 ms</td> </tr> <tr> <td>Original size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1185840 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>49.994%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>521229</td> </tr> <tr> <td>Optimal Threshold</td> <td>7.0196</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(4).png ? Compression GIF: test\output\gif\test7(4).gif</pre>	COMPRESSION RESULTS		Execution Time	8888.7424 ms	Original size	2371660 bytes	Compressed Size	1185840 bytes	Compression Rate	49.994%	Tree Depth	10	Total Nodes	521229	Optimal Threshold	7.0196	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>11647.8399 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>474455 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>79.994%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>193381</td> </tr> <tr> <td>Optimal Threshold</td> <td>8.2439</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(5).png ? Compression GIF: test\output\gif\test7(5).gif</pre>	COMPRESSION RESULTS		Execution Time	11647.8399 ms	Original Size	2371660 bytes	Compressed Size	474455 bytes	Compression Rate	79.994%	Tree Depth	10	Total Nodes	193381	Optimal Threshold	8.2439	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>810.5435 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>5336 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>99.775%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(6).png ? Compression GIF: test\output\gif\test7(6).gif</pre>	COMPRESSION RESULTS		Execution Time	810.5435 ms	Original Size	2371660 bytes	Compressed Size	5336 bytes	Compression Rate	99.775%	Tree Depth	0	Total Nodes	1														
COMPRESSION RESULTS																																																															
Execution Time	8888.7424 ms																																																														
Original size	2371660 bytes																																																														
Compressed Size	1185840 bytes																																																														
Compression Rate	49.994%																																																														
Tree Depth	10																																																														
Total Nodes	521229																																																														
Optimal Threshold	7.0196																																																														
COMPRESSION RESULTS																																																															
Execution Time	11647.8399 ms																																																														
Original Size	2371660 bytes																																																														
Compressed Size	474455 bytes																																																														
Compression Rate	79.994%																																																														
Tree Depth	10																																																														
Total Nodes	193381																																																														
Optimal Threshold	8.2439																																																														
COMPRESSION RESULTS																																																															
Execution Time	810.5435 ms																																																														
Original Size	2371660 bytes																																																														
Compressed Size	5336 bytes																																																														
Compression Rate	99.775%																																																														
Tree Depth	0																																																														
Total Nodes	1																																																														
Input :	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test7.png ? Valid image found!</td> <td></td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</td> <td></td> </tr> <tr> <td colspan="2">? Choose method (1-5): 3 ? method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 0 ? threshold set!</td> <td></td> </tr> <tr> <td>? Minimum block size: 1 ? Minimum block size set to 1</td> <td></td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%</td> <td></td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test7(7).png ? Output path set!</td> <td></td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(7).gif ? gif path set!</td> <td></td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test7.png ? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)		? Choose method (1-5): 3 ? method selected: Max Pixel Difference		? Enter threshold (0-255): 0 ? threshold set!		? Minimum block size: 1 ? Minimum block size set to 1		? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%		? Absolute path for compressed image: test\output\image\test7(7).png ? Output path set!		? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(7).gif ? gif path set!		<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test7.png ? Valid image found!</td> <td></td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</td> <td></td> </tr> <tr> <td colspan="2">? Choose method (1-5): 3 ? method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 0 ? threshold set!</td> <td></td> </tr> <tr> <td>? Minimum block size: 1 ? Minimum block size set to 1</td> <td></td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%</td> <td></td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test7(8).png ? Output path set!</td> <td></td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(8).gif ? gif path set!</td> <td></td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test7.png ? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)		? Choose method (1-5): 3 ? method selected: Max Pixel Difference		? Enter threshold (0-255): 0 ? threshold set!		? Minimum block size: 1 ? Minimum block size set to 1		? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%		? Absolute path for compressed image: test\output\image\test7(8).png ? Output path set!		? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(8).gif ? gif path set!		<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">QUADTREE IMAGE COMPRESSION TOOL</th> </tr> </thead> <tbody> <tr> <td>? Absolute path of input image: test\input\test7.png ? Valid image found!</td> <td></td> </tr> <tr> <td colspan="2">? ERROR MEASUREMENT METHOD</td> </tr> <tr> <td>1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)</td> <td></td> </tr> <tr> <td colspan="2">? Choose method (1-5): 3 ? method selected: Max Pixel Difference</td> </tr> <tr> <td>? Enter threshold (0-255): 255 ? threshold set!</td> <td></td> </tr> <tr> <td>? Minimum block size: 255 ? Minimum block size set to 255</td> <td></td> </tr> <tr> <td>? Target compression percentage (0.0-1.0): 0 ? Target compression disable</td> <td></td> </tr> <tr> <td>? Absolute path for compressed image: test\output\image\test7(9).png ? Output path set!</td> <td></td> </tr> <tr> <td>? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(9).gif ? gif path set!</td> <td></td> </tr> </tbody> </table>	QUADTREE IMAGE COMPRESSION TOOL		? Absolute path of input image: test\input\test7.png ? Valid image found!		? ERROR MEASUREMENT METHOD		1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)		? Choose method (1-5): 3 ? method selected: Max Pixel Difference		? Enter threshold (0-255): 255 ? threshold set!		? Minimum block size: 255 ? Minimum block size set to 255		? Target compression percentage (0.0-1.0): 0 ? Target compression disable		? Absolute path for compressed image: test\output\image\test7(9).png ? Output path set!		? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(9).gif ? gif path set!	
QUADTREE IMAGE COMPRESSION TOOL																																																															
? Absolute path of input image: test\input\test7.png ? Valid image found!																																																															
? ERROR MEASUREMENT METHOD																																																															
1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)																																																															
? Choose method (1-5): 3 ? method selected: Max Pixel Difference																																																															
? Enter threshold (0-255): 0 ? threshold set!																																																															
? Minimum block size: 1 ? Minimum block size set to 1																																																															
? Target compression percentage (0.0-1.0): 0.5 ? Target set to 50.0%																																																															
? Absolute path for compressed image: test\output\image\test7(7).png ? Output path set!																																																															
? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(7).gif ? gif path set!																																																															
QUADTREE IMAGE COMPRESSION TOOL																																																															
? Absolute path of input image: test\input\test7.png ? Valid image found!																																																															
? ERROR MEASUREMENT METHOD																																																															
1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)																																																															
? Choose method (1-5): 3 ? method selected: Max Pixel Difference																																																															
? Enter threshold (0-255): 0 ? threshold set!																																																															
? Minimum block size: 1 ? Minimum block size set to 1																																																															
? Target compression percentage (0.0-1.0): 0.8 ? Target set to 80.0%																																																															
? Absolute path for compressed image: test\output\image\test7(8).png ? Output path set!																																																															
? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(8).gif ? gif path set!																																																															
QUADTREE IMAGE COMPRESSION TOOL																																																															
? Absolute path of input image: test\input\test7.png ? Valid image found!																																																															
? ERROR MEASUREMENT METHOD																																																															
1. Variance (0 - 1626.25) 2. Mean Absolute Deviation (0 - 127.5) 3. Max Pixel Difference (0 - 255) 4. Entropy (0 - 8) 5. Structural Similarity Index (0 - 1)																																																															
? Choose method (1-5): 3 ? method selected: Max Pixel Difference																																																															
? Enter threshold (0-255): 255 ? threshold set!																																																															
? Minimum block size: 255 ? Minimum block size set to 255																																																															
? Target compression percentage (0.0-1.0): 0 ? Target compression disable																																																															
? Absolute path for compressed image: test\output\image\test7(9).png ? Output path set!																																																															
? Do you want to save compression GIF? (y/n): y ? Enter absolute path to save GIF: test\output\gif\test7(9).gif ? gif path set!																																																															
Max Pixel Difference																																																															
Nilai Ambang Batas :	0	0	255																																																												
Output :	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>16171.6430 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1165317 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>50.8649%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>521689</td> </tr> <tr> <td>Optimal Threshold</td> <td>27.3333</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(7).png ? Compression GIF: test\output\gif\test7(7).gif</pre>	COMPRESSION RESULTS		Execution Time	16171.6430 ms	Original Size	2371660 bytes	Compressed Size	1165317 bytes	Compression Rate	50.8649%	Tree Depth	10	Total Nodes	521689	Optimal Threshold	27.3333	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>11856.9836 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>479687 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>79.7742%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>187925</td> </tr> <tr> <td>Optimal Threshold</td> <td>40.3333</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(8).png ? Compression GIF: test\output\gif\test7(8).gif</pre>	COMPRESSION RESULTS		Execution Time	11856.9836 ms	Original Size	2371660 bytes	Compressed Size	479687 bytes	Compression Rate	79.7742%	Tree Depth	10	Total Nodes	187925	Optimal Threshold	40.3333	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>783.0901 ms</td> </tr> <tr> <td>Original size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>5336 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>99.775%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <pre>? Output Files: ? Compressed Image: test\output\image\test7(9).png ? Compression GIF: test\output\gif\test7(9).gif</pre>	COMPRESSION RESULTS		Execution Time	783.0901 ms	Original size	2371660 bytes	Compressed Size	5336 bytes	Compression Rate	99.775%	Tree Depth	0	Total Nodes	1														
COMPRESSION RESULTS																																																															
Execution Time	16171.6430 ms																																																														
Original Size	2371660 bytes																																																														
Compressed Size	1165317 bytes																																																														
Compression Rate	50.8649%																																																														
Tree Depth	10																																																														
Total Nodes	521689																																																														
Optimal Threshold	27.3333																																																														
COMPRESSION RESULTS																																																															
Execution Time	11856.9836 ms																																																														
Original Size	2371660 bytes																																																														
Compressed Size	479687 bytes																																																														
Compression Rate	79.7742%																																																														
Tree Depth	10																																																														
Total Nodes	187925																																																														
Optimal Threshold	40.3333																																																														
COMPRESSION RESULTS																																																															
Execution Time	783.0901 ms																																																														
Original size	2371660 bytes																																																														
Compressed Size	5336 bytes																																																														
Compression Rate	99.775%																																																														
Tree Depth	0																																																														
Total Nodes	1																																																														

Input :			
Entropy			
Nilai Ambang Batas :	0	0	8
Output :			
Input :			

Structural Similarity Index																																																	
Nilai Ambang Batas :	0	0	1																																														
Output :	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>42084.4301 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>1185342 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>58.0206%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>622021</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.3915</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(13).png ? Compression GIF: test\output\gif\test7(13).gif</p>	COMPRESSION RESULTS		Execution Time	42084.4301 ms	Original Size	2371660 bytes	Compressed Size	1185342 bytes	Compression Rate	58.0206%	Tree Depth	10	Total Nodes	622021	Optimal Threshold	0.3915	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>34186.1977 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>474378 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>79.9981%</td> </tr> <tr> <td>Tree Depth</td> <td>10</td> </tr> <tr> <td>Total Nodes</td> <td>260541</td> </tr> <tr> <td>Optimal Threshold</td> <td>0.5234</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(14).png ? Compression GIF: test\output\gif\test7(14).gif</p>	COMPRESSION RESULTS		Execution Time	34186.1977 ms	Original Size	2371660 bytes	Compressed Size	474378 bytes	Compression Rate	79.9981%	Tree Depth	10	Total Nodes	260541	Optimal Threshold	0.5234	<table border="1"> <thead> <tr> <th colspan="2">COMPRESSION RESULTS</th> </tr> </thead> <tbody> <tr> <td>Execution Time</td> <td>695.0548 ms</td> </tr> <tr> <td>Original Size</td> <td>2371660 bytes</td> </tr> <tr> <td>Compressed Size</td> <td>5336 bytes</td> </tr> <tr> <td>Compression Rate</td> <td>99.7750%</td> </tr> <tr> <td>Tree Depth</td> <td>0</td> </tr> <tr> <td>Total Nodes</td> <td>1</td> </tr> </tbody> </table> <p>? Output Files: ? Compressed Image: test\output\image\test7(15).png ? Compression GIF: test\output\gif\test7(15).gif</p>	COMPRESSION RESULTS		Execution Time	695.0548 ms	Original Size	2371660 bytes	Compressed Size	5336 bytes	Compression Rate	99.7750%	Tree Depth	0	Total Nodes	1
COMPRESSION RESULTS																																																	
Execution Time	42084.4301 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	1185342 bytes																																																
Compression Rate	58.0206%																																																
Tree Depth	10																																																
Total Nodes	622021																																																
Optimal Threshold	0.3915																																																
COMPRESSION RESULTS																																																	
Execution Time	34186.1977 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	474378 bytes																																																
Compression Rate	79.9981%																																																
Tree Depth	10																																																
Total Nodes	260541																																																
Optimal Threshold	0.5234																																																
COMPRESSION RESULTS																																																	
Execution Time	695.0548 ms																																																
Original Size	2371660 bytes																																																
Compressed Size	5336 bytes																																																
Compression Rate	99.7750%																																																
Tree Depth	0																																																
Total Nodes	1																																																

4.2 Analisis Pengujian dan Algoritma

Melalui berbagai eksperimen yang dilakukan. Seluruh jenis parameter dengan metode Quadtree efektif dalam melangsungkan sebuah kompresi terhadap gambar. Poin utama dari yang dilakukan dalam program ini adalah menghomogenisasi nilai warna dari blok-blok pecahan gambar yang lebih besar. Karena pemecahan sebuah blok gambar menjadi 4 bagian, dengan luas area gambar n yang memiliki lebar W dan tinggi H , nilai kompleksitas algoritmanya adalah $O(\log_4(n))$ (nilai asimptotik adalah $O(\log(n))$). Hal ini didapatkan dari dalam pemecahan menjadi 4 blok kecil, setiap blok akan memiliki luas $n/4$, dalam pemecahan kedua terdapat 16 blok dengan luas $n/16$. Akan tetapi, waktu yang tetap dibutuhkan tetap sama yaitu n (jumlah piksel) dalam setiap tahapannya. Dengan mengasumsikan titik terjauh/terlama dari pemecahan adalah L , ukuran tujuan blok terkecil adalah 1 piksel (terburuk), maka pada level L , akan ada total piksel sejumlah 4^L . Berdasarkan konsep logaritma, maka nilai L adalah $\log_4(n)$. Dengan pengetahuan ini, kita dapat mencari kompleksitas algoritma total dari setiap metode. Kompleksitas waktu pemecahan ini merupakan nilai waktu yang dibutuhkan untuk mencapai kedalaman, sedangkan setiap metode merupakan kompleksitas algoritma yang dibutuhkan dalam pemecahan setiap levelnya.

Walaupun algoritma dasar yang digunakan sama yaitu pemecahan sebuah blok besar menjadi beberapa blok kecil, ada persyaratan yang berbeda-beda untuk setiap metode. Di dalam persyaratan yang ada, terdapat sebuah variabel yaitu nilai error atau nilai rata-rata warna yang dapat ditemukan dalam satu blok yang sama. Dengan metode variansi, nilai error akan diukur dengan seberapa jauh piksel-piksel dalam blok tersebut dari rata-ratanya. Sedangkan untuk metode *Mean Absolute Deviation* (MAD) mengukur rata-rata deviasi absolut dari tiap piksel dengan rata-ratanya. Metode *entropy* akan mengukur distribusi nilai piksel dalam suatu blok. Terakhir, metode *Structural Similarity Index* (SSIM) akan mengukur dengan kesamaan blok asli dan blok yang telah terkompresi.

Dalam metode variansi, formula pengukuran error yang digunakan adalah :

$$\sigma_c^2 = \frac{1}{N} \sum_{i=1}^N (P_{i,c} - \mu_c)^2$$

$$\sigma_{RGB}^2 = \frac{\sigma_R^2 + \sigma_G^2 + \sigma_B^2}{3}$$

σ_c^2 = Variansi tiap kanal warna c (R, G, B) dalam satu blok

$P_{i,c}$ = Nilai piksel pada posisi i untuk kanal warna c

μ_c = Nilai rata-rata tiap piksel dalam satu blok

N = Banyaknya piksel dalam satu blok

Perhitungan yang dilakukan terdapat pada perhitungan mean, dan iterasi selisih kuadrat setiap piksel terhadap mean pada setiap piksel. Sehingga untuk perhitungan mean, dibutuhkan waktu $O(W \cdot H)$ dengan W adalah lebar gambar dan H adalah tinggi gambar. Perhitungan iterasi juga sebanyak $O(W \cdot H)$. Tidak hanya itu, karena setiap kanal warna dihitung, maka totalnya adalah $O(3 \cdot 2 \cdot W \cdot H)$. Akan tetapi, dalam penulisan asimptotik, nilai kompleksitas waktunya adalah $O(W \cdot H)$ atau sama dengan $O(n)$ (n adalah jumlah piksel total). Waktu inilah yang dibutuhkan dalam menghitung nilai variansi dalam setiap level. Sehingga dengan L kedalaman, dan $O(\log(n))$ kedalaman, maka total kompleksitas waktunya adalah $O(n) * O(\log n) = O(n \log n)$.

Sedangkan dalam metode *Mean Absolute Deviation* (MAD), perhitungan error dilakukan dengan rumus

$$MAD_c = \frac{1}{N} \sum_{i=1}^N |P_{i,c} - \mu_c|$$

$$MAD_{RGB} = \frac{MAD_R + MAD_G + MAD_B}{3}$$

MAD_c = Mean Absolute Deviation tiap kanal warna c (R, G, B)
dalam satu blok

$P_{i,c}$ = Nilai piksel pada posisi i untuk kanal warna c

μ_c = Nilai rata-rata tiap piksel dalam satu blok

N = Banyaknya piksel dalam satu blok

Proses waktu yang dibutuhkan oleh metode ini mirip dengan perhitungan menggunakan variansi. Dalam metode ini, dibutuhkan perhitungan rata-rata dengan waktu $O(W \cdot H)$. Kemudian iterasi jumlah deviasi absolut dengan waktu yang sama yaitu $O(W \cdot H)$. Sehingga totalnya adalah $O(3 \cdot 2 \cdot W \cdot H)$. Dalam penulisan asimptotik tetap $O(W \cdot H)$ dengan total kompleksitas waktu adalah $O(n) * O(\log n) = O(n \log n)$.

Metode *Max Pixel Difference* menggunakan rumus

$$D_c = \max(P_{i,c}) - \min(P_{i,c})$$

$$D_{RGB} = \frac{D_R + D_G + D_B}{3}$$

D_c = Selisih antara piksel dengan nilai max dan min tiap kanal warna c (R, G, B) dalam satu blok

$P_{i,c}$ = Nilai piksel pada posisi i untuk channel warna c

Sehingga pada eksekusinya, metode ini hanya membutuhkan iterasi sekali saja terhadap semua piksel untuk mencari nilai maksimum dan minimum sehingga kompleksitas waktu totalnya adalah $O(W \cdot H)$. Sehingga total kompleksitas waktunya adalah $O(n) * O(\log n) = O(n \log n)$.

Pada metode entropi, rumus yang digunakan adalah

$$H_c = - \sum_{i=1}^N P_c(i) \log_2(P_c(i))$$

$$H_{RGB} = \frac{H_R + H_G + H_B}{3}$$

H_c = Nilai entropi tiap kanal warna c (R, G, B) dalam satu blok

$P_c(i)$ = Probabilitas piksel dengan nilai i dalam satu blok untuk tiap kanal warna c (R, G, B)

Di dalam metode ini, dibutuhkan pembuatan sebuah histogram yang isinya merupakan luas gambar sehingga waktu yang dibutuhkan adalah $O(W*H)$. Proses selanjutnya adalah perhitungan entropi yang membutuhkan waktu $O(K)$ dengan K adalah 256 yaitu jumlah level piksel. Sehingga total waktu yang dibutuhkan adalah $O(W*H + K)$. Akan tetapi, dengan penulisan asimptotik, total kompleksitas waktunya adalah $O(n) * O(\log n) = O(n \log n)$.

Metode terakhir yaitu *Structural Similarity Index* (SSIM) memiliki rumus

$$SSIM_c(x, y) = \frac{(2\mu_{x,c}\mu_{y,c} + C_1)(2\sigma_{xy,c} + C_2)}{(\mu_{x,c}^2 + \mu_{y,c}^2 + C_1)(\sigma_{x,c}^2 + \sigma_{y,c}^2 + C_2)}$$

$$SSIM_{RGB} = w_R \cdot SSIM_R + w_G \cdot SSIM_G + w_B \cdot SSIM_B$$

Metode ini sebenarnya cukup kompleks karena harus menghitung rata-rata, variansi, dan kovariansi untuk dua buah gambar (biasa dan sudah terkompresi). Meskipun demikian, apabila ditulis dengan penulisan asimptotik, nilai kompleksitas waktunya masih pada $O(n)$. Sehingga total kompleksitas waktunya adalah $O(n) * O(\log n) = O(n \log n)$.

Sebagai sebuah kesimpulan, nilai total kompleksitas waktu dari setiap metode yang digunakan sama yaitu $O(n \log n)$. Hal ini bermakna bahwa dengan jumlah piksel total sebanyak n, maka terdapat $\log n$ operasi tambahan bagi setiap pikselnya. Nilai ini merupakan hasil dari algoritma *Divide and Conquer* yang diaplikasikan dalam program ini. Nilai $\log n$ sendiri berasal dari jumlah pembagian (kedalaman pohon) yang ditelusuri dan n adalah jumlah pemrosesan pada setiap level kedalamannya.

Walaupun dalam penulisan asimptotiknya sama, setiap metode memiliki jumlah operasi yang berbeda-beda. Metode yang paling berat adalah *Structural Similarity Index* karena harus menghitung berbagai variabel pada 2 gambar. Disusul oleh metode entropi yang disebabkan oleh perhitungan entropinya. Kemudian metode variansi dan *Mean Absolute Distribution* yang menghitung $O(n)$ sebanyak 6 kali. Metode yang paling efisien adalah *Max Pixel Difference* karena hanya harus melakukan iterasi sebanyak sekali.

Untuk memperkuat argumen ini, kita dapat menganalisis kompleksitas waktu menggunakan Teorema Master. Pada algoritma, proses rekursif akan membagi satu buah blok gambar menjadi 4 sehingga ukuran setiap bloknya adalah $(n/4)$. Dengan demikian, nilai kompleksitas waktu rekurensinya dapat dinyatakan dengan $T(n) = 4 * T(n/4) + O(n)$. Dapat dianalisis menggunakan teorema master didapatkan nilai $a = 4$ (jumlah sub-masalah), nilai $b = 4$ (ukuran sub-masalah), dan $f(n) = O(n)$ (waktu pengolahan blok dan perhitungan error), sehingga nilai d adalah 1. Dalam teori master, kompleksitas waktu algoritma dapat dilihat dengan hubungan antara a dengan b^d . Karakteristik teorema master dapat dilihat di bawah ini :

$$\begin{aligned} O(n^d) & \quad \text{jika } a < b^d \\ O(n^d \log n) & \quad \text{jika } a = b^d \\ O(n^{\log_b a}) & \quad \text{jika } a > b^d \end{aligned}$$

Melalui analisis rekurensinya, kita mendapatkan nilai $a = b$ dipangkatkan d . Karena nilai a sama dengan b , dan d adalah pangkat 1. Maka dari itu, dapat disimpulkan berdasarkan teorema master dan analisis algoritma, didapatkan nilai kompleksitas waktu dari algoritma kompresi menggunakan metode Quadtree adalah $O(n \log n)$.

4.3 Analisis Fitur Bonus

Di dalam program ini, terdapat 3 fitur bonus yang telah diimplementasikan. Fitur tersebut mencakup target persentase kompresi, metode *Structural Similarity Index* (SSIM), dan pembuatan GIF proses kompresi. Ketiga fitur ini telah tersedia dalam program dan dapat diakses melalui *interface*.

Fitur target kompresi memungkinkan pengguna untuk mengatur nilai pengurangan ukuran hasil kompresi melalui nilai *floating point*. Tanpa fitur ini, pengguna hanya bisa mengatur nilai ambang batas error yang diterima dan ukuran blok minimum yang diinginkan. Dengan adanya fitur ini, algoritma akan menyesuaikan nilai ambang batas secara otomatis dan dinamis agar dapat memenuhi target yang diberikan oleh pengguna. Dalam aplikasinya, proses yang dilakukan adalah melakukan kompresi secara bertahap. Setelah melakukan kompresi, maka gambar hasil akan disimpan sementara dan nilai persentase kompresi akan dihitung. Proses ini akan dilakukan berulang kali hingga persentase kompresi mencapai target yang diinginkan. Implementasi fitur ini mencakup penggunaan fungsi `compress` dan `findOptimalThreshold` pada class `QuadtreeCompression` dan fungsi `calcCompressionPercentage` serta `writeCompressesImage` pada class `IO`.

Metode *Structural Similarity Index* (SSIM) adalah salah satu metode perhitungan error yang dapat digunakan. Pada dasarnya, metode ini akan menghitung perbedaan nilai warna antara blok gambar sebelum dan sesudah dikompresi. Di dalam setiap blok, akan

dihitung nilai SSIM untuk setiap kanal warna yang pada akhirnya akan digabungkan. Perhitungan ini mencakup perhitungan rata-rata, variansi, dan kovariansi intensitas warna pada setiap gambar sebelum akan dibandingkan. Implementasinya dapat dilihat pada fungsi calculateSSIM pada class ErrorMetrics. Di dalam class ErrorMetrics juga dapat ditemukan fungsi-fungsi pembantu seperti calculateSSIM, calculateChannelSSIM, calculateCovariance, dan calculateBlockSSIM.

Fitur terakhir adalah pembuatan GIF proses kompresi. Melalui GIF ini, akan didapatkan visualisasi mengenai proses pembentukan Quadtree dalam kompresi gambar ini. Dalam berkas GIF, dapat dilihat proses secara bertahap dari kompresi dimulai dari gambar orisinil hingga ke hasil akhir berdasarkan kedalaman dari pohon Quadtree yang telah dibuat. Hal ini diimplementasikan pada method createCompressionGif pada class IO. Dalam pembuatannya, terdapat beberapa fungsi pembantu untuk rekonstruksi gambar yang dapat dilihat pada fungsi reconstructGIFFromQuadtree dan reconstructNodeforGIF.

LAMPIRAN

Tabel Kelengkapan Spesifikasi

No	Poin	Ya	Tidak
1	Program berhasil dikompilasi tanpa kesalahan.	✓	
2	Program berhasil dijalankan	✓	
3	Program berhasil melakukan kompresi gambar sesuai parameter yang ditentukan.	✓	
4	Mengimplementasi seluruh metode perhitungan error wajib.	✓	
5	[Bonus] Implementasi persentase kompresi sebagai parameter tambahan.	✓	
6	[Bonus] Implementasi Structural Similarity Index (SSIM) sebagai metode pengukuran error.	✓	
7	[Bonus] Output berupa GIF Visualisasi Proses pembentukan Quadtree dalam Kompresi Gambar.	✓	
8	Program dan laporan dibuat (kelompok) sendiri.	✓	

DAFTAR PUSTAKA

- Munir, Rinaldi. 2025. “Algoritma Divide and Conquer (Bagian 1)” ([https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/07-Algoritma-Divide-and-Conquer-\(2025\)-Bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/07-Algoritma-Divide-and-Conquer-(2025)-Bagian1.pdf), diakses 8 April 2025)
- Munir, Rinaldi. 2025. “Algoritma Divide and Conquer (Bagian 2)” ([https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/05-Algoritma-Greedy-\(2025\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/05-Algoritma-Greedy-(2025)-Bag2.pdf), diakses 8 April 2025)
- Munir, Rinaldi. 2025. “Algoritma Divide and Conquer (Bagian 3)” ([https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/09-Algoritma-Divide-and-Conquer-\(2025\)-Bagian3.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/09-Algoritma-Divide-and-Conquer-(2025)-Bagian3.pdf), diakses 8 April 2025)
- Munir, Rinaldi. 2025. “Algoritma Divide and Conquer (Bagian 4)” ([https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/10-Algoritma-Divide-and-Conquer-\(2025\)-Bagian4.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/10-Algoritma-Divide-and-Conquer-(2025)-Bagian4.pdf), diakses 8 April 2025)