

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Pengolahan Citra Digital
Kode : IES 6323
Semester : VI
Waktu : 2 x 3x 50 Menit
Pertemuan : 13 & 14

A. Kompetensi

1. Utama

Mahasiswa dapat memahami tentang sistem pengolahan citra digital dan hal yang terkait secara umum.

1. Pendukung

Mahasiswa dapat mengetahui tentang pemampatan citra

B. Pokok Bahasan

Pemampatan Citra

C. Sub Pokok Bahasan

- Pendahuluan
- Metode Pemampatan Citra

D. Kegiatan Belajar Mengajar

Tahapan Kegiatan	Kegiatan Pengajaran	Kegiatan Mahasiswa	Media & Alat Peraga
Pendahuluan	1. Mereview materi sebelumnya 2. Menjelaskan materi-materi perkuliahan yang akan dipelajari.	Mendengarkan dan memberikan komentar	Notebook, LCD, Papan Tulis

Penyajian	3. Menjelaskan tentang pemampatan citra secara umum 4. Menjelaskan tentang metode pemampatan citra	Memperhatikan, mencatat dan memberikan komentar. Mengajukan pertanyaan.	Notebook, LCD, Papan Tulis
Penutup	1. Mengajukan pertanyaan kepada mahasiswa. 2. Memberikan kesimpulan. 3. Mengingatkan akan kewajiban mahasiswa untuk pertemuan selanjutnya.	Memberikan komentar. Mengajukan dan menjawab pertanyaan.	Notebook, LCD, Papan Tulis

E.Evaluasi

Evaluasi dilakukan dengan cara memberikan pertanyaan langsung dan tidak langsung kepada mahasiswa dan dengan memberikan kuis.

RENCANA KEGIATAN BELAJAR MINGGUAN
(RKBM)

Mata Kuliah : Pengolahan Citra Digital
Kode : IES 6323
Semester : VI
Waktu : 1 x 3x 50 Menit
Pertemuan : 13 & 14

Minggu ke-	Topik (Pokok Bahasan)	Metode Pembelajaran	Estimasi Waktu (Menit)	Media
13	9.1 Pendahuluan 9.2 Metode Pemampatan Citra 9.2.1 Lossless Compression	Ceramah, Diskusi Kelas	1 x 3 x 50'	Notebook, LCD, Papan Tulis
14	9.2.2 Lossy Compression	Ceramah, Diskusi Kelas	1 x 3 x 50'	Notebook, LCD, Papan Tulis

PEMAMPATAN CITRA

Pemampatan citra atau kompresi citra (*image compression*) bertujuan meminimalkan kebutuhan memori untuk merepresentasikan citra digital. Prinsip umum yang digunakan pada proses pemampatan citra adalah mengurangi duplikasi data di dalam citra sehingga memori yang dibutuhkan untuk merepresentasikan citra, menjadi lebih sedikit daripada representasi citra semula.

9.1 Pendahuluan

Pemampatan citra kadang-kadang disalah mengertikan dengan pengkodean citra (*image encoding*), yaitu persoalan bagaimana pixel-pixel di dalam citra dikodekan dengan representasi tertentu. Pengkodean citra yang menghasilkan representasi memori yang lebih sedikit daripada representasi aslinya itulah yang dinamakan pemampatan citra.

Ada dua proses utama dalam persoalan pemampatan citra :

1. Pemampatan citra (*image compression*).

Citra dalam representasi tidak mampat dikodekan dengan representasi yang meminimumkan kebutuhan memori. Citra dengan format bitmap pada umumnya tidak dalam bentuk mampat. Citra yang sudah dimampatkan disimpan ke dalam arsip dengan format tertentu. Kita mengenal format JPG dan GIF sebagai format citra yang sudah dimampatkan.

2. Peninnampatkan citra (*image decompression*).

Citra yang sudah dimampatkan harus dapat dikembalikan lagi (*decoding*) menjadi representasi yang tidak mampat. Proses ini diperlukan jika citra tersebut ditampilkan ke layar atau disimpan ke dalam arsip dengan format tidak mampat. Dengan kata lain, peninnampatan citra mengembalikan citra yang termampatkan menjadi data bitmap.

9.2 Aplikasi Pemampatan Citra

Pemampatan citra bermanfaat untuk aplikasi yang melakukan :

1. Pengiriman data (*data transmission*) pada saluran komunikasi data

Citra yang telah dimampatkan membutuhkan waktu pengiriman yang lebih singkat dibandingkan dengan citra yang tidak dimampatkan. Contohnya aplikasi pengiriman gambar lewat fax, video conferencing,

pengiriman data medis, pengiriman gambar dari satelit luar angkasa, pengiriman gambar via telepon genggam. download gambar dari internet. dan sebagainya.

2. Penyimpanan data (data storing) di dalana media sekunder (storage)
Citra yang telah dimampatkan membutuhkan ruang memori di dalam media storage yang lebih sedikit dibandingkan dengan citra yang tidak dimampatkan. Contoh aplikasinya antara lain aplikasi basis data gambar, office automation, video storage (seperti Video Compact Disc), dll.

9.3 Kriteria Pemampatan Citra

Kriteria yang digunakan dalam mengukur metode pemampatan citra adalah:

1. Waktu pemampatan dan penirmampatan (decompression).
Waktu pemampatan citra dan pemampatannya sebaiknya cepat. Ada metode pemampatan yang waktu pemampatannya lama, namun waktu pemampatannya cepat. Ada pula metode yang waktu pemampatannya cepat tetapi waktu pemampatannya lambat. Tetapi ada pula metode yang waktu pemampatan dan pemampatannya cepat atau keduanya lambat.
2. Kebutuhan memori.
Memori yang dibutuhkan untuk merepresentasikan citra seharusnya berkurang secara berarti. Ada metode yang berhasil memampatkan dengan persentase yang besar, ada pula yang kecil. Pada beberapa metode, ukuran memori hasil pemampatan bergantung pada citra itu sendiri. Citra yang mengandung banyak elemen duplikasi (misalnya citra langit cerah tanpa awan, citra lantai keramik) umumnya berhasil dimampatkan dengan memori yang lebih sedikit dibandingkan dengan memampatkan citra yang mengandung banyak objek (misalnya citra pemandangan alam).
3. Kualitas pemampatan (fidelity)
Informasi yang hilang akibat pemampatan seharusnya seminimal mungkin sehingga kualitas hasil pemampatan tetap dipertahankan. Kualitas pemampatan dengan kebutuhan memori biasanya berbanding terbalik. Kualitas pemampatan yang bagus umumnya dicapai pada proses pemampatan yang menghasilkan pengurangan memori yang tidak begitu besar, demikian pula sebaliknya.

Ukuran kualitas hasil pemampatan citra menjadi ukuran kuantitatif dengan menggunakan besaran PSNR (peak signal-to-noise ratio). PSNR dihitung untuk mengukur perbedaan antara citra semula dengan citra hasil pemampatan (tentu saja citra hasil pemampatan harus dimampatkan terlebih dahulu) dengan citra semula, dengan rumus :

$$PSNR = 20 \times \log_{10} \left(\frac{b}{rms} \right) \quad (9.1)$$

dengan b adalah nilai sinyal terbesar (pada citra hitam-putih, $b = 255$) dan rms adalah akar pangkat dua dari selisih antara citra semula dengan citra hasil pemampatan. Nilai rms dihitung dengan rumus :

$$rms = \sqrt{\frac{1}{Lebar \times Tinggi} \sum_{i=1}^N \sum_{j=1}^N (f_{ij} - f'_{ij})^2} \quad (9.2)$$

yang dalam hal ini, f dan f' masing-masing menyatakan nilai pixel citra semula dan nilai pixel citra hasil pemampatan. PSNR memiliki satuan decibel (M). Persamaan (9.2) menyatakan bahwa PSNR hanya dapat dihitung setelah proses pemampatan citra. Dari persamaan (9.2) terlihat bahwa PSNR berbanding terbalik dengan rms . Nilai rms yang rendah yang menyiratkan bahwa citra hasil pemampatan tidak jauh berbeda dengan citra semula akan menghasilkan PSNR yang tinggi, yang berarti kualitas pemampatannya bagus. Semakin besar nilai PSNR, semakin bagus kualitas pemampatannya. Seberapa besar nilai PSNR yang bagus tidak dapat dinyatakan secara eksplisit, bergantung pada citra yang dimampatkan.

9.4 Jenis Pemampatan Citra

Ada empat pendekatan yang digunakan dalam pemampatan citra :

1. Pendekatan statistik.

Pemampatan citra didasarkan pada frekuensi kemunculan derajat keabuan pixel di dalam seluruh bagian gambar.

Contoh metode: Huffman Coding.

2. Pendekatan ruang

Pemampatan citra didasarkan pada hubungan spasial antara pixel-pixel di dalam suatu kelompok yang memiliki derajat keabuan yang sama di dalam suatu daerah di dalam gambar.

Contoh metode : Run-Length Encoding.

3. Pendekatan kuantisasi

Pemampatan citra dilakukan dengan mengurangi jumlah derajat keabuan yang tersedia.

Contoh metode : metode pemampatan kuantisasi.

4. Pendekatan fraktal

Pemampatan citra didasarkan pada kenyataan bahwa kemiripan bagian-bagian di dalam citra dapat dieksploitasi dengan suatu matriks transformasi. Contoh metode: Fractal Image Compression.

9.5 Klasifikasi Metode Pemampatan

Metode pemampatan citra dapat diklasifikasikan ke dalam dua kelompok besar :

1. Metode lossless

Metode lossless selalu menghasilkan citra hasil penirmpatan yang tepat sama dengan citra semula, pixel per pixel. Tidak ada informasi yang hilang akibat pemampatan. Sayangnya nisbah (ratio) pemampatan citra metode lossless sangat rendah.

Contoh metode lossless adalah metode Human.

Nisbah pemampatan citra, dihitung dengan rumus

$$\text{Nisbah} = 100\% - \left(\frac{\text{ukuran citra hasil pempatan}}{\text{ukuran citra semula}} \times 100\% \right)$$

(10.3)

Metode lossless cocok untuk memampatkan citra yang mengandung informasi penting yang tidak boleh rusak akibat pemampatan. Misalnya memampatkan gambar hasil diagnosa medis.

2. Metode lossy

Metode lossy menghasilkan citra hasil pemampatan yang hampir sama dengan citra semula. Ada informasi, yang hilang akibat pemampatan, tetapi dapat ditolerir oleh persepsi mata. Mata tidak dapat membedakan perubahan kecil pada gambar. Metode pemampatan lossy menghasilkan nisbah pemampatan yang tinggi daripada metode lossless.

9.6 Metode Pemampatan Huffman

Metode pemampatan Huffman menggunakan prinsip bahwa nilai (atau derajat) keabuan yang sering muncul di dalam citra akan dikodekan dengan jumlah bit

yang lebih sedikit sedangkan nilai keabuan yang frekuensi kemunculannya sedikit dikodekan dengan jumlah b_i , yang lebih panjang.

Algoritma metode Huffman :

1. Urutkan secara menaik (ascending order) nilai-nilai keabuan berdasarkan frekuensi kemunculannya (atau berdasarkan peluang kemunculan, p_k , yaitu frekuensi kemunculan (n_k) dibagi dengan jumlah pixel di dalam gambar (n). Setiap nilai keabuan dinyatakan sebagai pohon bersimpul tunggal. Setiap simpul di-assign dengan frekuensi kemunculan nilai keabuan tersebut.
2. Gabung dua buah pohon yang mempunyai frekuensi kemunculan paling kecil pada sebuah akar. Akar mempunyai frekuensi yang merupakan jumlah dari frekuensi dua, buah pohon penyusunnya.
3. Ulangi langkah 2 sampai tersisa hanya satu buah pohon biner.

Agar pemilihan dua pohon yang akan digabungkan berlangsung cepat, maka semua pohon yang ada selalu terurut menaik berdasarkan frekuensi.

4. Beri label setiap sisi pada pohon biner. Sisi kiri dilabeli dengan 0 dan sisi kanan dilabeli dengan 1.

Simpul-simpul daun pada pohon biner menyatakan nilai keabuan yang terdapat di dalam citra semula. Untuk mengkodekan setiap pixel di dalam citra. Lakukan langkah kelima berikut :

5. Telusuri pohon biner dari akar ke daun. Barisan label-label sisi dari akar ke daun menyatakan kode Huffman untuk derajat keabuan yang bersesuaian.

Setiap kode Huffman merupakan kode prefiks, yang artinya tidak ada kode biner suatu nilai keabuan yang merupakan awalan bagi kode biner derajat keabuan yang lain. Dengan cara ini, tidak ada ambiguitas pada proses penirmpampatan citra.

Contoh 9.1.

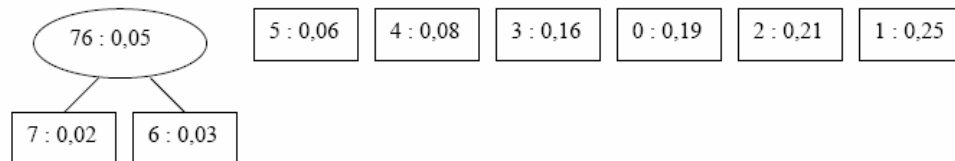
Misalkan terdapat citra yang berukuran 64×64 dengan 8 derajat keabuan (k) dan jumlah seluruh pixel (n) = $64 \times 64 = 4096$

k	n_k	$p(k) = n_k/n$
0	790	0.19
1	1023	0.25
2	850	0.21
3	656	0.16
4	329	0.08
5	245	0.06
6	122	0.03
7	81	0.02

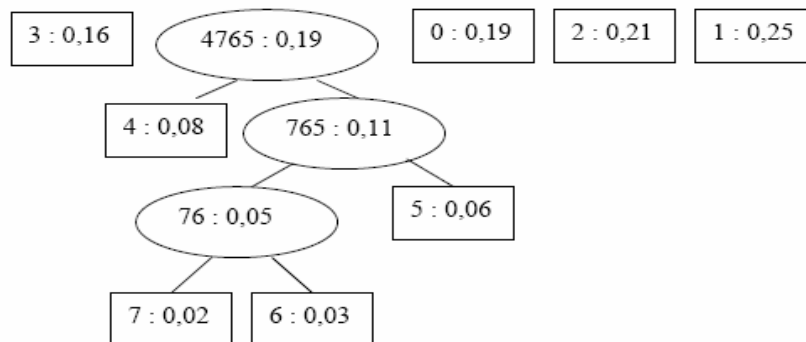
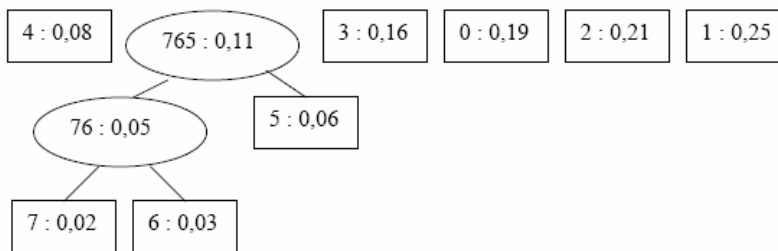
Langkah 1 :

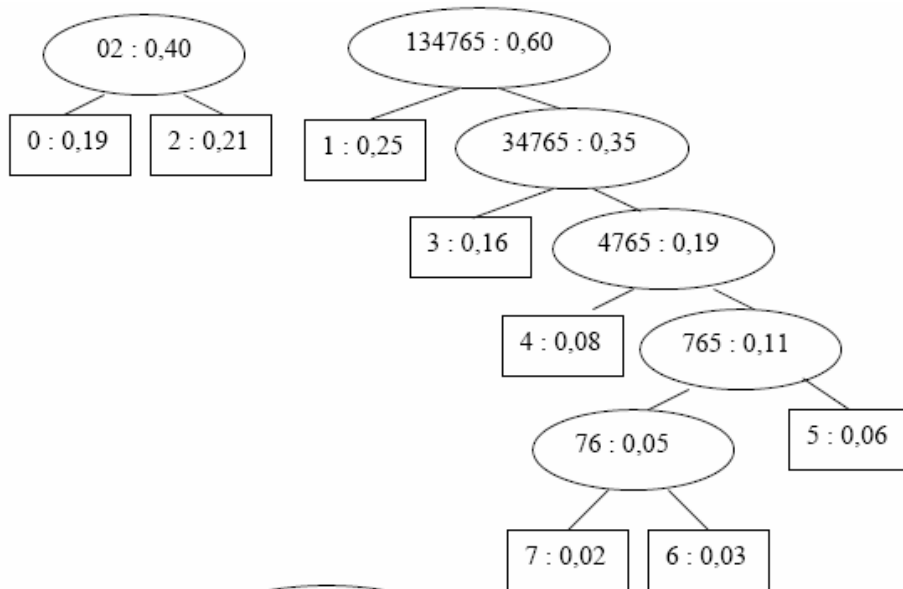
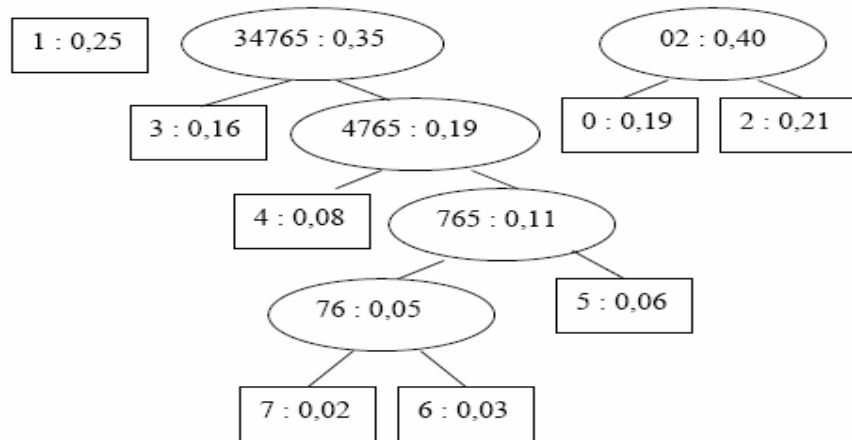
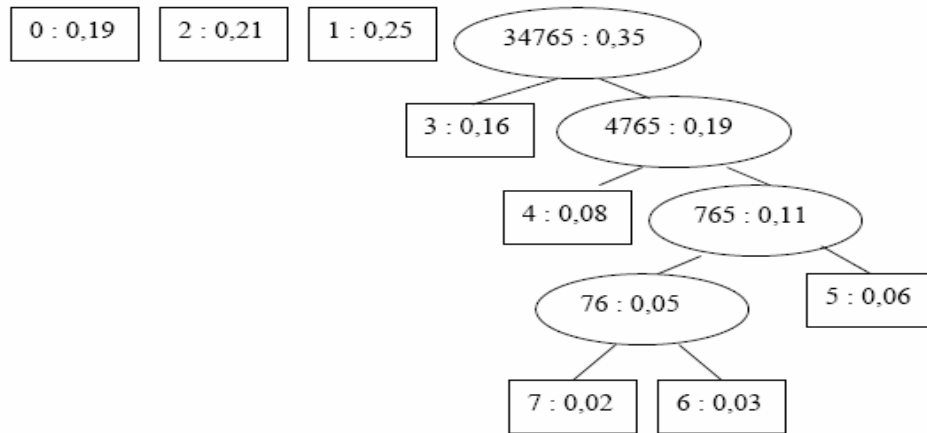


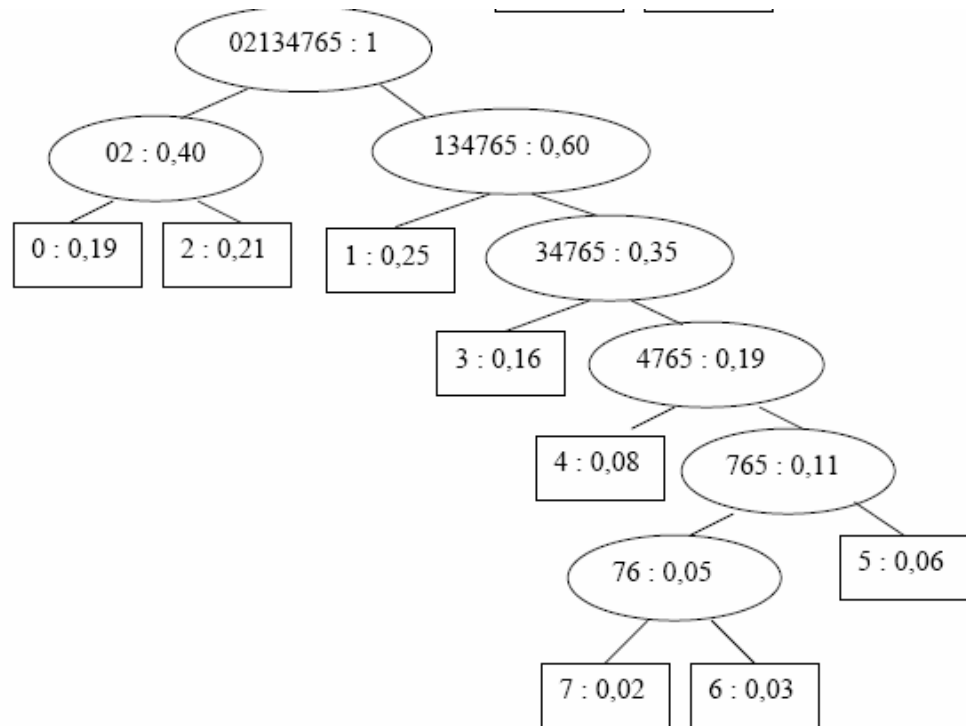
Langkah 2 :



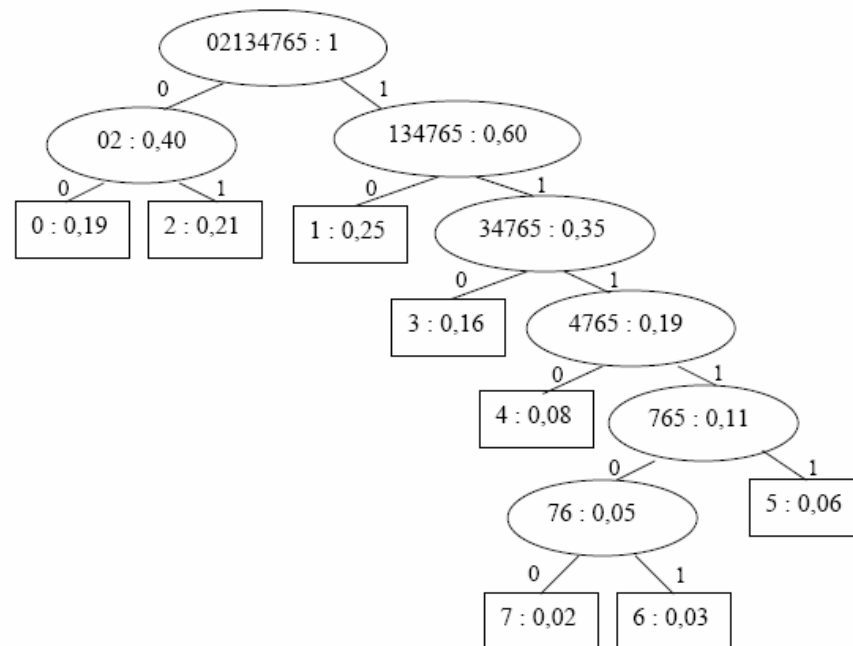
Langkah 3 : Ulangi langkah 1 & 2 sampai tersisa 1 pohon biner
Ingat, frekuensi yang lebih kecil diletakkan di sisi kiri







Langkah 4 : Beri label 0 untuk sisi kiri, sisi kanan label 1



Langkah 5 : Telusuri pohon biner dari akar ke daun untuk menentukan kode Huffman yang sesuai dengan derajat keabuan.

Derajat keabuan	Kode Huffman	Ukuran	Banyaknya piksel
0	00	2 bit	790
1	10	2 bit	1023
2	01	2 bit	850
3	110	3 bit	656
4	1110	4 bit	329
5	11111	5 bit	245
6	111101	6 bit	122
7	111100	6 bit	81

$$\begin{aligned}
 \text{Ukuran citra setelah kompresi} &= (790 \times 2 \text{ bit}) + (1023 \times 2 \text{ bit}) + (850 \times 2 \text{ bit}) + \\
 &\quad (656 \times 3 \text{ bit}) + (329 \times 4 \text{ bit}) + (245 \times 5 \text{ bit}) + \\
 &\quad (122 \times 6 \text{ bit}) + (81 \times 6 \text{ bit}) \\
 &= 11053 \text{ bit}
 \end{aligned}$$

$$\text{Ukuran citra sebelum kompresi} = 4096 \text{ piksel} \times 3 \text{ bit} = 12288 \text{ bit}$$

$$\begin{aligned}
 \text{Ratioskompresi} &= 100\% - \left(\frac{\text{ukuran citra hasil kompresi}}{\text{ukuran citra asli}} \times 100\% \right) \\
 &= 100\% - \left(\frac{11053}{12288} \times 100\% \right) \\
 &= 10\%, \text{ artinya citra semula telah dimampatkan sebanyak } 10\%
 \end{aligned}$$

10,7 Metode Pemampatan Run-Length Encoding (RLE)

Metode RLE cocok digunakan untuk memampatkan citra yang memiliki kelompok-kelompok pixel berderajat keabuan lama. Pemampatan citra dengan metode RLE dilakukan dengan membuat rangkaian pasangan nilai (p, q) untuk setiap baris pixel, nilai pertama (p) menyatakan derajat keabuan, sedangkan nilai kedua (q) menyatakan jumlah pixel berurutan yang memiliki derajat keabuan tersebut (dinamakan run length).

Contoh 9.2. Tinjau citra 10 x 10 pixel dengan 8 derajat keabuan yang dinyatakan sebagai matriks derajat keabuan sebagai berikut

0	0	0	0	0	2	2	2	2	2
0	0	0	1	1	1	1	2	2	2
1	1	1	1	1	1	1	1	1	1
4	4	4	4	3	3	3	3	2	2
3	3	3	5	5	7	7	7	7	6
2	2	6	0	0	0	0	1	1	0
3	3	4	4	3	2	2	2	1	1
0	0	0	0	0	0	0	0	1	1
1	1	1	1	0	0	0	2	2	2
3	3	3	2	2	2	1	1	1	1

Semuanya ada 100 buah nilai.

Pasangan nilai untuk setiap baris run yang dihasilkan dengan metode pemampatan RLE :

(0,5), (2,5)
 (0,3), (1,4), (2,3)
 (1,10)
 (4,4), (3,4), (2,2)
 (3,3), (5,2), (7,4), (6,1)
 (2,2), (6,1), (0,4), (1,2), (0,1)
 (3,2), (4,2), (3,1), (2,2), (1,2)
 (0,8), (1,2)
 (1,4), (0,3), (2,3)
 (3,3), (2,3), (1,4)

semuanya ada 31 pasangan nilai atau $31 \times 2 = 62$ nilai.

Ukuran citra sebelum pemampatan (1 derajat keabuan = 3 bit) adalah

$$= 100 \times 3 \text{ bit} = 300 \text{ bit},$$

sedangkan ukuran citra setelah pemampatan (derajat keabuan = 3 bit, run length = 4 bit): $(31 \times 3) + (31 \times 4) \text{ bit} = 217 \text{ bit}$

$$\text{Nisbah pemampatan} = \left(100\% - \frac{217}{300} \times 100\% \right) = 27.67\%$$

yang artinya 27.67% dari citra semula telah dimampatkan.

9.8 Metode Pemampatan Kuantisas (quantizing Compression)

Metode ini mengurangi jumlah derajat keabuan, misalnya dari 256 menjadi 16, yang tentu saja mengurangi jumlah bit yang dibutuhkan untuk mempresentasikan citra.

Misalkan P adalah jumlah pixel di dalam citra semula, akan dimampatkan menjadi n derajat keabuan. Algoritmanya adalah sebagai berikut :

Algoritma metode kuantisasi:

1. Buat histogram citra semula. (citra yang akan dimampatkan).
2. Identifikasi n buah kelompok di dalam histogram sedemikian sehingga setiap kelompok mempunyai kira-kira P/n buah pixel.
3. Nyatakan setiap kelompok dengan derajat keabuan 0 sampai $n - 1$. Setiap pixel di dalam kelompok dikodekan kembali dengan nilai derajat keabuan yang baru.

Contoh 9.3. Tinjau citra yang berukuran 5 x 13 pixel :

2	9	6	4	8	2	6	3	8	5	9	3	7
3	8	5	4	7	6	3	8	2	8	4	7	3
3	8	4	7	4	9	2	3	8	2	7	4	9
3	9	4	7	2	7	6	2	1	6	5	3	0
2	0	4	3	8	9	5	4	7	1	2	8	3

yang akan dimampatkan menjadi citra dengan 4 derajat keabuan (0 s/d 3), jadi setiap derajat keabuan direpresentasikan dengan 2 bit.

Histogram citra semula :

Derajat Keabuan	Jumlah Pikel
0	2
1	2
2	9
3	11
4	9
5	4
6	5
7	8
8	9
9	6

Ada 65 pixel, dikelompokkan menjadi 4 kelompok derajat, keabuan tiap kelompok ada sebanyak rata-rata $65/4 = 16.25$ pixel per kelompok :

Kelompok	Nilai Keabuan	Jumlah Pixel	Jumlah Pixel dalam kelompok
1	0	2	13
	1	2	
	2	9	
2	3	11	20
	4	9	
3	5	4	17
	6	5	
	7	8	
4	8	9	15
	9	6	

Citra setelah dimampatkan menjadi :

```

0 3 2 1 3 0 2 1 3 2 3 1 2
1 3 2 1 2 2 1 3 0 3 1 2 1
1 3 1 2 1 3 0 1 3 0 2 1 3
1 3 1 2 0 2 2 0 0 2 2 1 0
0 0 1 1 3 3 2 1 2 0 0 3 0

```

Ukuran citra sebelum pemampatan (1 derajat keabuan = 4 bit):

$$65 \times 4 \text{ bit} = 260 \text{ bit}$$

Ukuran citra setelah pemampatan (1 derajat keabuan = 2 bit):

$$65 \times 2 \text{ bit} = 130 \text{ bit}$$

$$\text{Nisbah pemampatan} = \left(100\% - \frac{130}{260} \times 100\% \right) = 50\%,$$

yang artinya 50% dari citra semula telah dimampatkan.

Kelemahan metode pemampatan kuantisasi adalah banyaknya informasi yang hilang, tapi kehilangan informasi ini dapat diminimalkan dengan menjamin bahwa tiap kelompok mempunyai jumlah pixel yang hampir sama.

9.9 Metode Pemampatan Fraktal

Metode pemampatan fraktal adalah metode yang relatif baru. Prinsipnya adalah mencari bagian di dalam citra yang memiliki kemiripan dengan bagian lainnya namun ukurannya lebih besar (*self similarity*). Kemudian dicari matriks yang mentransformasikan bagian yang lebih besar tersebut dengan bagian yang lebih kecil. Kita cukup hanya menyimpan elemen-elemen dari sekumpulan matriks transformasi tersebut (yang disebut matriks transformasi affine). Pada proses penrimampatan, matriks transformasi affine di-iterasi sejumlah kali terhadap sembarang citra awal. Hasil iterasi akan konvergen ke citra semula. Metode ini menghasilkan nisbah pemampatan yang tinggi namun waktu pemampatannya relatif lama, sedangkan waktu penrimampatannya berlangsung cepat.

Latihan 9

Citra ukuran 7 x 13 piksel (4 bit), dikuantisasi dengan 3 bit. Tentukan citra hasil kuantisasi!

2	9	6	13	13	13	6	3	3	3	3	3	7
3	8	5	4	6	6	3	8	8	8	8	8	3
11	9	4	10	2	7	12	13	14	14	14	11	12
1	0	0	0	8	9	5	15	7	1	2	8	3
1	9	6	4	8	2	6	15	15	15	15	3	7
10	8	4	10	15	9	2	15	8	2	7	4	9
3	8	5	4	7	6	3	8	8	8	4	7	3