

KOMPRESI CITRA GRAY SCALE DENGAN MODIFIKASI ALGORITMA KUANTISASI

Krisnawati
STMIK AMIKOM Yogyakarta

Abstract

Suatu file yang kapasitasnya besar dapat diperkecil dengan pemampatan (compression). Untuk file image metode kuantisasi bisa menjadi salah satu pilihan. Pada citra grayscale metode kuantisasi bekerja dengan mengurangi derajat keabuan, sehingga jumlah bit yang digunakan untuk merepresentasikan citra menjadi berkurang. Oleh karena jumlah bit berkurang maka ukuran file menjadi lebih kecil. Dari algoritma kuantisasi yang ada, dilakukan modifikasi pada saat konversi derajat keabuan lama ke derajat keabuan yang baru, dengan mengabaikan unsur penyebaran derajat keabuan. Hasilnya kapasitas citra hasil kompresi tidak berubah. Histogram citra sangat berpengaruh terhadap hasil akhir.

Keyword : *kompresi, citra grayscale, kuantisasi.*

1. Pendahuluan

Perkembangan media penyimpan berkapasitas besar mengakibatkan orang tidak lagi menemui masalah jika mempunyai file dengan ukuran yang besar. Lebih-lebih jika file yang kita punya merupakan file image. Walaupun demikian, adakalanya ukuran file yang besar tersebut terasa mengganggu jika kita harus manage media penyimpan yang kita punya untuk bermacam-macam data. Apalagi jika file tersebut akan kita kirim secara elektronik, tentunya kapasitas file menjadi masalah tersendiri.

Kompresi citra (*image compression*) adalah proses untuk meminimalkan jumlah bit yang merepresentasikan suatu citra sehingga ukuran citra menjadi lebih kecil. Pada dasarnya teknik kompresi citra digunakan untuk proses transmisi data (*data transmission*) dan penyimpanan data (*storage*). Kompresi citra banyak

diaplikasikan pada penyiaran televisi, penginderaan jarak jauh (*remote sensing*), komunikasi militer, radar dan lain-lain.

Ada banyak metode kompresi data, salah satunya adalah metode kuantisasi. Metode kuantisasi bekerja dengan mengurangi jumlah intensitas warna, sehingga jumlah bit yang digunakan untuk merepresentasikan citra menjadi berkurang. Oleh karena jumlah bit berkurang maka ukuran file menjadi lebih kecil. Dengan berkurangnya intensitas warna tentu saja ada informasi yang hilang dari citra asal. Oleh karena itu metode ini termasuk dalam lossy compression. Oleh karena itu citra yang sudah dikompresi sulit didekompresi kembali karena adanya informasi yang hilang.

Proses kompresi tentunya akan berdampak kepada banyak hal. Yang pertama adalah ukuran citra hasil kompresi. Ukuran citra diharapkan lebih kecil dari citra asal. Kedua adalah kualitas citra untuk input terhadap proses berikutnya. Sampai berapa persenkah citra asli bisa dikompresi ini tentunya tergantung pada banyak factor. Faktor inilah yang ingin diketahui pula dalam penelitian ini.

Ada beberapa teknik kompresi yang dapat dikategorikan ke dalam Lossless maupun Lossy Compression, antara lain:

1. Kompresi berbasis Statistik (Lossless)
Merepresentasikan citra dengan frekuensi kemunculan nilai intensitas tertentu.
2. Kompresi berbasis Kuantisasi (Lossy)
Mengurangi jumlah intensitas warna.
3. Kompresi berbasis Transformasi (Lossless/Lossy)
Mengoptimalkan kinerja kompresi berbasis statistik dan kuantisasi dengan cara melakukan transformasi terlebih dahulu sebelum menerapkan salah satu teknik tersebut. Sehingga kompresi bersifat lossy atau lossless tergantung teknik mana yang digunakan setelah transformasi apakah itu statistik (lossless) atau kuantisasi (lossy).
4. Kompresi berbasis fraktal (Lossy)
Fraktal merupakan bentuk rekursif yang merepresentasikan komponen dasar objek. Dalam konsep kompresi, data direpresentasikan sebagai pasangan antar elemen fraktal, pola

umum konfigurasi yang membentuk objek secara keseluruhan, dan koefisien transformasi spasial (affine) untuk masing-masing fraktal sesuai dengan posisinya dalam konfigurasi pembentuk objek.

1.1 Proses Kompresi

1.1.1 Kompresi Berbasis Kuantisasi

Kompresi berbasis kuantisasi menggunakan metode pengurangan jumlah intensitas warna, sehingga dapat mengurangi jumlah bit yang digunakan untuk merepresentasikan citra. Kompresi ini bersifat lossy, karena intensitas warna berkurang, sehingga kualitas gambar hasil kompresi menjadi kurang baik.

Secara algoritma teknik kompresi ini diberikan sebagai berikut:

1. Buat histogram citra asal.
Histogram citra adalah grafik yang menunjukkan distribusi dari intensitas citra. Histogram citra menyatakan frekuensi kemunculan berbagai derajat keabuan dalam citra. Teknik pemodelan histogram mengubah citra hingga memiliki histogram sesuai keinginan. Untuk meningkatkan kualitas citra salah satunya dapat dilakukan dengan ekualisasi histogram. Dengan ekualisasi histogram dapat diperoleh histogram citra dengan distribusi seragam.
2. Tentukan jumlah kelompok dalam histogram. Jumlah tersebut menunjukkan tingkat intensitas warna citra kompresi.
3. Buat kelompok intensitas warna baru dengan menghitung jumlah piksel/jumlah kelompok.
4. Atur pengelompokan.
5. Lakukan kuantisasi.
6. Kodekan nilai intensitas warna hasil kuantisasi ke dalam citra kompresi.

Contoh: Diketahui citra array berukuran 6x6 piksel (8 derajat keabuan, 3 bit) sebagai berikut :

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

Citra diatas memiliki histogram sebagai berikut:

K	N_k
0	2
1	6
2	10
3	3
4	5
5	4
6	2
7	4

Misalnya citra akan dikompresi menjadi 4 derajat keabuan (0 – 3), sehingga setiap derajat keabuan direpresentasikan dalam 2 bit.

Citra asli terdiri dari 36 piksel, sehingga jika dibagi menjadi 4 kelompok maka dibuat $36/4= 9$ piksel tiap kelompok sehingga histogram baru menjadi sebagai berikut:

K	N_k	n_b	k_b
0	2	8	0
1	6		
2	10	10	1
3	3	8	2
4	5		
5	4	10	3
6	2		
7	4		

Dari histogram baru tersebut diperoleh citra baru (dengan 4 derajat keabuan, 2 bit) sebagai berikut:

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

Ukuran citra sebelum kompresi = 36 piksel x 3 bit = 108 bit

Ukuran citra setelah kompresi = 36 piksel x 2 bit = 72 bit

$$\begin{aligned} \text{Rasio kompresi} &= 100\% - \left(\frac{\text{UkuranCitraHasilKompresi}}{\text{UkuranCitraSemula}} \times 100\% \right) \\ &= 100\% - ((72/108) \times 100\%) \\ &= 100\% - 66.67\% \\ &= 33.33\% \end{aligned}$$

1.1.2 Pengembangan Algoritma Kompresi

Pengembangan dilakukan dengan mengubah proses pengkodean warna baru. Pengkodean dilakukan dengan cara yang lebih sederhana dengan mengelompokkan intensitas warna yang ada ke intensitas warna kompresi yang diinginkan.

Contoh:

Citra asal terdiri dari 8 derajat keabuan akan dikompresi menjadi 4 derajat keabuan.

K	N_k	k_b
0	2	0
1	6	0
2	10	1
3	3	1
4	5	2
5	4	2
6	2	3
7	4	3

Histogram citra terkompresi menjadi sebagai berikut:

K	N _k
0	8
1	13
2	9
3	6

Dari histogram diatas didapat matrik citra terkompresi sebagai berikut:

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

Untuk ukuran citra kompresi tidak berubah dari algoritma asal yakni 36 piksel x 3 bit = 108 bit.

2. Pembahasan

Allgoritma diatas diimplementasikan dengan MatLab 6.5. Gambar lama yang semula mempunyai 256 derajat keabuan (8 bit), dikompresi menjadi 128 derajat keabuan (7 bit). Langkah pertama adalah melakukan pembacaan terhadap file citra sebagai berikut:

```
clc;clear;  
j=imread('anakblm.jpg');  
gblama=j;
```

Selanjutnya dicari ukuran matrik citra. Ukuran ini nantinya akan digunakan sebagai indek perulangan dalam proses selanjutnya. Selain itu juga harus diketahui derajat keabuan minimum dan maksimun untuk menentukan ukuran vektor histogram citra.

```
[m,n]=size(gblama);  
array=double(gblama);  
figure,imshow(gblama);
```

```
%cari derajat keabuan min dan max
```

```
abumax=max(max(array));  
abumin=min(min(array));
```

Histogram citra asal dapat diketahui dengan cara menghitung kemunculan untuk setiap derajat keabuan yang ada di dalam citra.

%bikin histogram image yg pake fungsi imhist

```
arraybaru=zeros(1,256);  
for i=1:256  
    for j=1:m  
        for k=1:n  
            if (array(j,k)==i)  
                arraybaru(i)=arraybaru(i)+1;  
            end  
        end  
    end  
end;  
end;
```

```
%hitung jumlah pixel  
jpixel=m*n;
```

```
%gambar histogram  
figure,plot(arraybaru);
```

```
%cari rentang derajat keabuan  
derajatabu=(abumax-abumin)+1;
```

```
%membuat histogram kompresi  
%256 drjt keabuan (8 bit) menjadi  
%128 derajat keabuan (7bit)  
hiskompresi=zeros(1,128);  
j=1;  
for i=1:128  
    hiskompresi(i)=arraybaru(j)+arraybaru(j+1);  
    j=j+2;  
end
```

```

%bikin array image baru dari hasil kompresi
gbbaru=zeros(m,n);
for i=1:m
    for j=1:n
        if mod(double(gblama(i,j)),2)==0
            gbbaru(i,j)=((double(gblama(i,j))+1))/2;
        else
            gbbaru(i,j)=double(gblama(i,j))/2;
        end
    end
end
end

```

```

%konversi array gambar terkompresi
%dari double ke uint8
gbbaru1=uint8(gbbaru);

```

```

%Tampilkan histogram gambar hasil kompresi
%dan gambar hasil kompresi
figure, plot(hiskompresi);
figure, imshow(gbbaru1);

```

```

%Simpan gambar ke media penyimpan
imwrite(gbbaru1,'anakkomp.jpg');

```

Hasil pada saat program dijalankan dengan beberapa sampel gambar yang berbeda adalah sbb:

Gambar	Ukuran file sebelum kompresi	Ukuran file setelah kompresi	Rasio kompresi
1	57.5 KB	33.8 KB	41.22 %
2	4.27 KB	2.84 KB	33.49 %
3	2.5 KB	1.84 KB	26.40%
4	15.2 KB	10.5 KB	30.92 %
5	3.89 KB	2.72 KB	30.08%

6	4.86 KB	3.54 KB	27.16 %
7	3.49 KB	2.51 KB	28.08 %
8	5.25 KB	3.66 KB	30.29 %
9	2.39 KB	1.76 KB	26.36%
10	5.12 KB	3.62 KB	29.36%

Rerata rasio kompresi didapat sebesar 30.33%

3. Kesimpulan

Modifikasi yang dilakukan terhadap algoritma asal tidak mengubah kapasitas file dari citra hasil kompresi. Presentase kompresi dipengaruhi oleh histogram citra yang diberikan citra asal. Tingkat kemunculan derajat keabuan suatu citra, antara citra yang satu dengan citra yang lainpun berbeda-beda.

Daftar Pustaka

Departmen Teknik Elektro, *Modul Praktikum Pengolahan Citra dan Pengenalan Pola*, Institut Teknologi Bandung.

Paul Wintz, 2000, *Digital Image Processing*, Prentice-Hall.

MatLab 6 Help.

William J Palm, 2004, *Introduction to MatLab 6 for Engineers*, The McGraw-Hill Companies, Inc.

www.iprg.ee.itb.ac.id/lectures.html

www.datacompression.info/Quantization.shtml

www.cbloom.com/src/index_im.html