

Algoritma Greedy dalam Kompresi Gambar

Fauzan Muhammad Rifqy and 13513081

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

fauzanrifqy@students.itb.ac.id

Abstrak—Makalah ini membahas mengenai pemanfaatan algoritma greedy dalam melakukan kompresi gambar yang menggunakan prinsip kode Huffman. Algoritma greedy banyak digunakan dalam berbagai macam permasalahan seperti persoalan penukaran uang, minimisasi waktu, dan lain sebagainya. Tujuan penggunaan algoritma greedy ialah untuk mendapatkan solusi global dengan pendekatan penyelesaian dari solusi lokal.

Kata Kunci—Algoritma Greedy, Huffman, Kompresi.

I. PENDAHULUAN

Kompresi Gambar dapat dikatakan sebagai salah satu area dibidang *Image Processing* yang telah berkembang dengan sangat baik.

Kompresi gambar memampatkan data yang dimiliki oleh sebuah gambar dan meminimalisasi redundansi data sehingga hasil kompresi dapat disimpan dalam memori dengan ukuran yang lebih rendah.

Ada beragam jenis/format kompresi gambar yang telah banyak dikenal, diantaranya yaitu :

- GIF (Graphics Interchange Format)
- JPG/JPEG (Joint Photographic Expert Group)
- BMP (Bitmap Image)
- PNG (Portable Network Graphic)
- TIFF (Tagged Image File Format)
- dll.



Image from <http://feelgrafix.com/782564-cute-flower-wallpaper.html>

Setiap gambar terdiri dari sejumlah pixel yang tersusun dalam bidang dua dimensi yang tersusun diskrit. Masing-masing pixel memiliki warna yang beragam tersusun dalam bit data yang tersimpan.

Jumlah seluruh kemungkinan warna yang dihasilkan

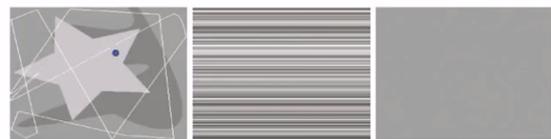
oleh sebuah pixel berbeda-beda untuk setiap format kompresi gambar.

Ketika kita menyimpan data dari sebuah gambar, semisal kita mempunyai gambar dengan resolusi 1000x1000 pixel. Artinya jumlah pixel dalam bidang ialah 1.000.000 pixel, dimana untuk masing-masing pixel memiliki representasi data warna (biasanya terdiri dari merah, hijau, biru) terdiri dari 24 *bits of data*. Itu artinya untuk 1 buah gambar kita dapat memerlukan 1000x1000x24 *bits of data* untuk menyimpannya dalam memory.

Video juga merupakan kumpulan data gambar yang di susun berupa frame/waktu. Dimana pada video standar terdapat 30 gambar dalam 1 detik. Maka dapat kita hitung pulang data video untuk 1 detik merupakan data gambar yang dikalikan 30.

Data gambar yang kita simpan memiliki angka yang sangat besar setelah kita tinjau. Maka perlu adanya pemampatan data agar gambar yang kita simpan tidak memakan ruang memori yang besar.

Pada sebagian gambar terdapat nilai pixel yang muncul lebih sering dibandingkan nilai pixel yang lain. sebagai contoh :



Digital Image Processing, 3rd ed.
FIGURE 8.1

Ketiga gambar diatas memiliki resolusi yang sama. Namun kita lihat pada gambar 1 misalnya, bahwa nilai warna abu muncul lebih banyak, tanpa adanya kompresi gambar, gambar tersebut akan memiliki 8 bits data untuk masing-masing pixel, namun kita fahami bahwa seharusnya hal tersebut tidaklah kita harapkan, mengingat untuk sekian banyak pixel kita akan selalu memiliki nilai 8 bits data yang sama.

Begitu juga pada gambar kedua kita lihat bahwa untuk setiap baris nilai 8 bits dari setiap pixel memiliki nilai yang sama.

Pada gambar ketiga seluruh nilai pixel memiliki nilai yang sama.

II. DASAR TEORI

A. Gambar Digital

Gambar digital merupakan sekumpulan nilai digital yang merepresentasikan gambar dua dimensi. Gambar digital terdiri dari sekumpulan elemen diskrit yang disebut pixel. Gambar digital memiliki jumlah baris dan kolom yang tetap, baris x kolom inilah yang nantinya disebut sebagai resolusi dari sebuah gambar digital.

B. Jenis Format Kompresi Gambar Digital

Berikut merupakan beberapa jenis format kompresi gambar digital

1. GIF (Graphics Interchange Format)

GIF memiliki kombinasi warna hingga 256 warna. Biasa digunakan untuk keperluan gambar sederhana seperti icon ataupun logo.

2. JPG/JPEG (Joint Photographic Expert Group)

JPG/JPEG memiliki kombinasi warna hingga 16.7 juta warna. Digunakan untuk pencitraan warna dengan kualitas tinggi.

3. BMP (Bitmap Image)

BMP merupakan representasi citra grafis yang dikembangkan oleh microsoft, terdiri dari susunan titik. Untuk gambar hitam-putih diawali dengan 1 bit data, sedangkan untuk warna memiliki bit data lebih besar.

4. PNG (Portable Network Graphic)

PNG merupakan format gambar terbaru. Data yang disimpan memiliki kemampuan penyimpanan data alpha channel.

5. TIFF (Tagged Image File Format)

TIFF merupakan format gambar yang fleksibel. ia dapat menyimpan data mulai dari 8-bit per warna hingga 16-bit.

C. Algoritma Greedy

Algoritma Greedy merupakan metode yang paling populer yang digunakan dalam persoalan optimasi.

Algoritma greedy merupakan sebuah metode pendekatan algoritma yang mengambil langkah optimum dengan mengambil pilihan yang paling optimum pada daerah lokal dengan harapan nantinya mendapatkan solusi yang mengarah ke optimum global.

Elemen-elemen pada algoritma greedy :

1. Himpunan Kandidat (C)

Himpunan yang didalamnya terdapat elemen-elemen yang dapat membentuk solusi.

2. Himpunan Solusi (S)

Himpunan yang elemen-elemennya merupakan sebuah solusi.

3. Fungsi Seleksi

Fungsi yang memilih elemen-elemen pada himpunan kandidat yang dapat diambil untuk masuk pada himpunan solusi.

4. Fungsi Kelayakan

Fungsi yang memeriksa elemen yang telah di seleksi untuk masuk ke Himpunan Solusi tidak melanggar konstrain.

5. Fungsi Obyektif

Fungsi yang mengindikasikan bahwa Himpunan solusi telah lengkap.

D. Prinsip Kode Huffman

Kode Huffman merupakan serangkaian kode data yang mampu merepresentasikan data dengan memori yang lebih kecil dengan mempertimbangkan kemunculan data yang lebih sering keluar untuk dimunculkan dalam bentuk kode data yang lebih pendek, sedangkan kode data yang muncul lebih jarang dimunculkan dalam bentuk kode yang lebih panjang.

E. Algoritma Greedy dalam Pembentukan Kode Huffman

Pertama-tama kita periksa seluruh karakter (dalam hal gambar digital kita membaca setiap data bits yang dimiliki oleh sebuah pixel) di dalam data untuk menghitung frekuensi kemunculan nilai data tersebut.

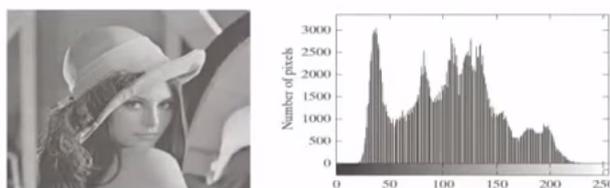
Setiap nilai data penyusun data dinyatakan sebagai pohon bersimpul tunggal.

Untuk 2 pohon yang memiliki frekuensi terendah gabungkan pada sebuah akar yang nilai frekuensinya merupakan penjumlahan dari nilai frekuensinya mula-mula. Lakukan terus menerus hingga tersisa satu pohon huffman.

III. ALGORITMA GREEDY DALAM PEMBENTUKAN KODE HUFFMAN PADA KOMPRESI GAMBAR

A. Penentuan Frekuensi Nilai

Diawal kita tentukan frekuensi untuk masing-masing nilai pada pixel. Kita perhatikan bahwa seringkali nilai dari sebuah pixel muncul beberapa kali pada pixel yang lain.



Digital Image Processing, 3rd ed.

Perhatikan bahwa pada contoh gambar diatas nilai dari pixel di daerah mendekati 50 muncul lebih banyak dibandingkan nilai pixel yang lain.

Gambar yang terdapat disebelah kanan (disebut Histogram) memperlihatkan kemunculan nilai pixel di daerah mendekati 50 di gambar mencapai angka 3000.

Sedangkan kita perhatikan pada daerah mendekati 200 kita lihat nilai pixel yang muncul lebih sedikit yaitu tidak lebih dari 1000.

Dari data tersebut kita simpulkan bahwa ada nilai pixel yang muncul dalam gambar sangat banyak namun ada juga yang sedikit muncul di gambar atau bahkan tidak muncul sama sekali.

B. Kode Huffman

Untuk memahami persoalan dengan lebih jelas, kita ilustrasikan dengan contoh persoalan sebagai berikut :

Misalkan kita memiliki sebuah nilai pixel dengan 8-bits kode warna dimana pada gambar digital terdapat kemunculan 4 buah nilai pixel yaitu {87,128,186,255} dengan masing-masing memiliki frekuensi kemunculan {0.25, 0.47, 0.25, 0.03}.

Kode untuk masing-masing nilai pixel dalam 8-bits adalah sebagai berikut :

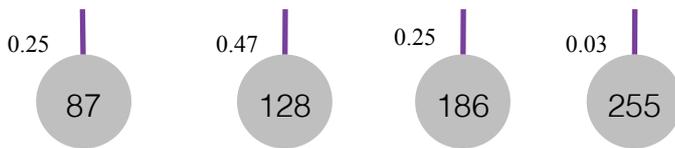
- 87 : 01010111
- 128 : 10000000
- 186 : 11000100
- 255 : 11111111

Sehingga kita dapatkan :

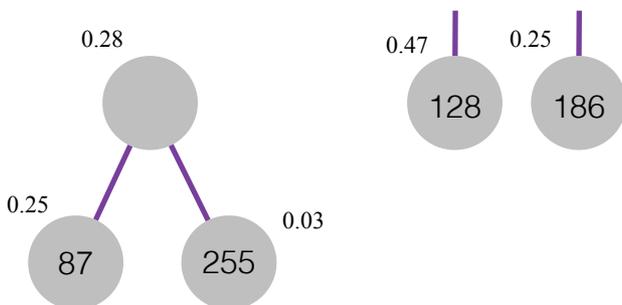
| r_k | $p(r_k)$ | Code |
|-------|----------|----------|
| 87 | 0.25 | 1010111 |
| 128 | 0.47 | 10000000 |
| 186 | 0.25 | 11000100 |
| 255 | 0.03 | 11111111 |

Algoritma Greedy Membentuk Kode Huffman :

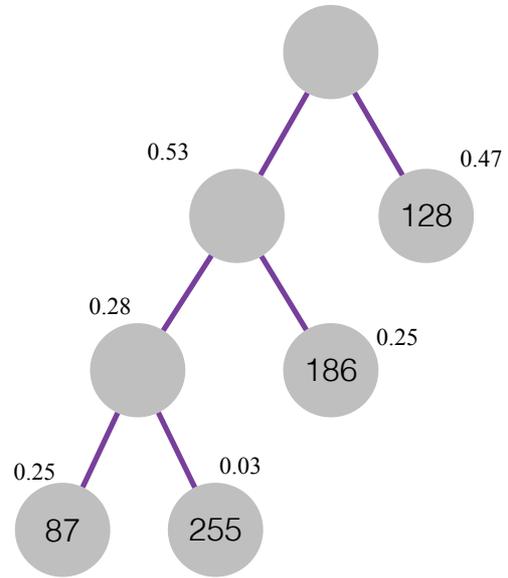
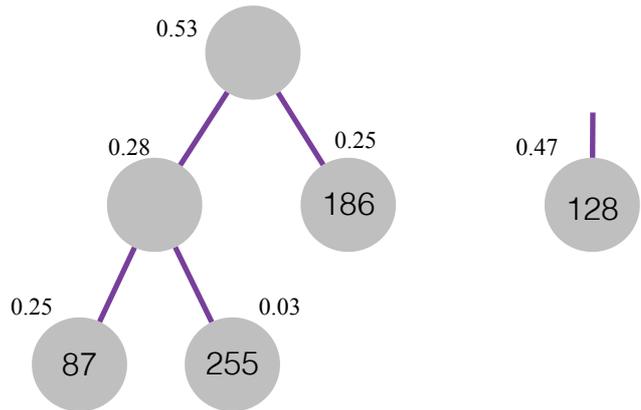
1. Nyatakan data dalam bentuk pohon bersimpul tunggal.



2. 2 buah pohon dengan frekuensi ($p(r_k)$) terkecil digabungkan pada sebuah akar.



3. Lakukan kembali hingga tersisa 1 buah pohon.



4. Kemudian kita ubah ke bentuk kode Huffman dimana untuk nilai yang ke kanan adalah '1' dan kirinya adalah '0' sehingga kita dapatkan :

| r_k | Huffman Code | Length(r_k) |
|-------|--------------|-----------------|
| 87 | 01 | 2 |
| 128 | 1 | 1 |
| 186 | 000 | 3 |
| 255 | 001 | 3 |

C. Kalkulasi Jumlah bit

Setelah kita melakukan kompresi maka kita akan coba membandingkan seberapa besar perbedaan ukuran antara gambar digital yang di kompresi dengan gambar tanpa kompresi.

Misalkan gambar digital yang telah kita kompresi sebelumnya memiliki resolusi 1000x1000 dengan masing-masing pixel memiliki nilai 8-bits (Sebelum Kompresi), sehingga ukuran penyimpanan gambar dalam memori ialah :

$$1000 \times 1000 \times 8\text{-bits} = 8\text{Mb} = 2\text{MB.}$$

Sedangkan pada gambar digital setelah kompresi kita hitung bahwa untuk setiap pixel kita akan memerlukan :

$$(0.25 * 2) + (0.47 * 1) + (0.25 * 3) + (0.03 * 3) = 1.81\text{-bits}$$

Sehingga kita dapatkan ukuran penyimpanan gambar digital setelah kompresi ialah :

$$1000 \times 1000 \times 1.81 = 1.81 \text{ Mb} = 0.4524 \text{ MB}$$

IV. KESIMPULAN

Gambar Digital seringkali memiliki redundansi data pada setiap nilai pixelnya sehingga kompresi gambar digital sangat bermanfaat dan sangat dibutuhkan untuk mengurangi ukuran filenya ketika disimpan dalam memori. Hal ini akan sangat berdampak ketika gambar disimpan dalam video dimana nilai ukurannya dalam file sangat besar.

Algoritma greedy dapat digunakan dalam implementasi huffman kode dan sangat bermanfaat dalam kompresi gambar dan video. Terlihat jelas perbedaan besar dari ruang memori yang dibutuhkan oleh gambar digital sebelum di kompresi dan setelah di kompresi.

Algoritma greedy menjadi salah satu metode efektif dalam mengimplementasikan huffman kode sebagai salah satu persoalan optimasi.

DAFTAR PUSTAKA

1. Munir, Rinaldi "Diktat Strategi Algoritma", Bandung : Penerbit ITB, 2009.
2. www.imageprocessingplace.com diakses pada tanggal 2-3 Mei 2015.
3. <https://ouwgan.wordpress.com/komputer/jenis-jenis-format-penyimpan-pada-gambar/> diakses pada tanggal 3 Mei 2015.
4. Gonzalez & Woods, "Digital Image Processing 3rd ed.", 2008.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Mei 2015



Fauzan Muhammad Rifqy-13513081