

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA: IRWANTO
NRP: 5106100137
DOSEN WALI: Prof. Dr. Ir. Joko Lianto Buliali, M. Sc.

2. JUDUL TUGAS AKHIR

“MODIFIKASI DAN OPTIMASI PERFORMA ALGORITMA KLASSTERING K-MEANS UNTUK KUANTISASI WARNA CITRA”

3. ABSTRAK

Kuantisasi warna merupakan operasi penting pada banyak aplikasi grafik dan pengolahan citra. Banyak metode kuantisasi berbasis algoritma klastering data. Sayangnya *k-means* tidak begitu populer sebagai algoritma klastering untuk kasus kuantisasi karena besarnya beban komputasi dan sensitivitasnya saat inisialisasi. Tugas Akhir ini mencoba untuk menelusuri performa dari *k-means* sebagai kuantiser warna. Tugas Akhir ini mengimplementasikan varian-varian yang cepat dan tepat dari *k-means* dengan beberapa skema inisialisasi dan membandingkan hasilnya dengan beberapa algoritma kuantisasi yang populer pada literatur.

4. PENDAHULUAN

4.1 Latar Belakang Masalah

Citra dengan warna asli umumnya mengandung banyak warna hingga ribuan warna yang pada akhirnya seringkali menjadi masalah saat penyimpanan, penampilan, pengiriman, atau pemrosesan. Dalam proses pengolahan citra dan grafik, kuantisasi (reduksi) warna umumnya digunakan saat tahap *preprocessing*. Di masa lalu, kuantisasi citra diperlukan karena terbatasnya perangkat keras yang tidak bisa menampilkan lebih dari 16 juta warna pada citra 24-bit. Walaupun perangkat keras telah berkembang dengan pesat, kuantisasi warna masih menjadi perhatian hingga saat ini karena banyak aplikasi grafik dan pemrosesan citra yang berkaitan dengan kuantisasi citra, seperti: (i) kompresi, (ii) segmentasi, (iii) deteksi/lokalisasi teks, (iv) analisis tekstur warna, (v) *water-marking*, (vi) *rendering*, dan (vii) sistem temu kembali berbasis isi.

Proses kuantisasi warna umumnya dibagi menjadi dua fase: desain palet (pemilihan himpunan kecil dari warna-warna yang mewakili warna asli citra) dan *mapping* piksel (menandai setiap piksel pada desain palet). Tujuan utama dari kuantisasi adalah mengurangi jumlah warna unik, N' , pada suatu citra menjadi K sehingga ($K \ll N'$) dengan distorsi sekecil mungkin. Pada banyak

aplikasi, piksel 24-bit pada citra asli direduksi menjadi 8-bit atau lebih kecil. Karena citra asli seringkali mengandung banyak warna, representasi citra yang dapat mewakili citra asli dengan ukuran palet kecil adalah suatu permasalahan.

Metode kuantisasi warna secara garis besar dapat dibedakan menjadi dua kategori: metode independen yang menetapkan keseluruhan palet tanpa memperhatikan citra secara spesifik dan metode dependen yang menetapkan palet berdasar distribusi warna citra. Meskipun sangat cepat, metode independen memberikan hasil yang kurang optimal karena mengabaikan konten citra. Berbeda dengan metode independen, metode dependen lebih memberikan keseimbangan yang baik antara efisiensi komputasi dan kualitas visual dari *output* kuantisasi.

Metode dependen dapat dikategorikan menjadi dua keluarga: metode *preclustering* dan *postclustering*. Metode *preclustering (non-uniform)* adalah metode yang didasarkan pada analisis statistik dari distribusi warna citra. Metode ini diawali dengan sebuah klaster yang mengandung semua N piksel citra. Klaster ini dibagi lagi secara rekursif sampai K klaster diperoleh. Adapun jenis-jenis dari metode ini antara lain: metode *median-cut*, *octree*, *variance-based*, *binary splitting*, *greedy orthogonal bipartitioning*, *optimal principal multilevel quantizer*, *center-cut*, dan *rwm-cut*. Sementara itu, metode *postclustering (agglomerative/uniform)* diawali dengan N klaster tunggal, masing-masing berisi satu piksel citra. Klaster-klaster ini kemudian digabung sampai didapatkan K klaster. Berbeda dengan metode *preclustering* yang menghitung palet hanya sekali, metode *postclustering* diawali dengan menentukan sebuah palet dan kemudian diperbaiki secara iteratif. Pada dasarnya semua metode klastering data dapat digunakan untuk tujuan kuantisasi. Karena metode ini melibatkan iterasi dan optimasi stokastik, metode-metode *postclustering* ini dapat menghasilkan kualitas hasil yang lebih baik bila dibandingkan dengan metode *preclustering*, walaupun berbanding lurus dengan peningkatan waktu komputasi. Algoritma *postclustering* ini meliputi *k-means*, *minmax*, *competitive-learning*, *fuzzy c-means*, *BIRCH*, dan *self-organizing maps*.

Tugas akhir ini menelusuri performa dari algoritma klastering *k-means* sebagai kuantiser warna. Tugas akhir ini mengimplementasikan sejumlah variasi dari setiap skema inisialisasi yang berbeda dan membandingkan hasilnya dengan beberapa metode kuantisasi warna yang telah populer.

4.2 Perumusan Masalah

Permasalahan yang diangkat dalam pembuatan tugas akhir ini adalah sebagai berikut:

1. Bagaimana memodifikasi algoritma konvensional klastering *k-means* untuk kuantisasi warna citra?
2. Bagaimana mempercepat performa algoritma *k-means*?
3. Bagaimana melakukan inisialisasi algoritma *k-means* pada kasus kuantisasi warna citra?

4.3 Batasan Masalah

Dari permasalahan yang telah disebutkan di atas, batasan-batasan dalam tugas akhir ini adalah:

1. File input berupa citra berwarna berukuran 512×512 piksel.
2. Citra yang dipakai adalah citra yang biasa dipakai sebagai citra tes pada literatur kuantisasi.
3. Membandingkan tingkat efisiensi antara algoritma klastering *k-means* konvensional dengan *k-means* yang telah dimodifikasi untuk kuantisasi warna.
4. Membandingkan tingkat efisiensi antara metode algoritma klastering *k-means* yang telah dimodifikasi dengan metode kuantisasi warna citra yang lain.
5. Implementasi algoritma menggunakan bahasa pemrograman Matlab.

4.4 Tujuan Tugas Akhir

Tugas akhir ini memiliki beberapa tujuan yang rinciannya dapat dituliskan sebagai berikut:

1. Memodifikasi metode algoritma klastering konvensional *k-means* untuk kuantisasi warna citra.
2. Merancang dan membuat sebuah algoritma yang dapat digunakan untuk menyelesaikan kuantisasi warna citra.
3. Mendapatkan solusi yang optimal dan efisien untuk kuantisasi warna dengan algoritma klastering *k-means* yang dimodifikasi.
4. Membangun sebuah aplikasi dengan mengimplementasikan metode algoritma *k-means* yang telah dimodifikasi untuk membantu menyelesaikan kuantisasi warna citra.

5. RINGKASAN TUGAS AKHIR

Pada Tugas akhir ini penulis mencoba mengimplementasikan algoritma klastering data *k-means* yang telah dimodifikasi untuk kuantisasi warna citra. Adapun langkah-langkah modifikasi dari algoritma klastering konvensional *k-means* adalah sebagai berikut:

1. Data Sampling

Cara mudah untuk mempercepat *k-means* adalah dengan mengurangi jumlah data, yang dapat dicapai dengan *subsampling* input data citra. Tugas akhir ini menggunakan dua metode *subsampling* deterministik. Metode pertama melibatkan *subsampling* 2 : 1 dalam arah horisontal dan vertikal, sehingga hanya seperempat dari input piksel citra yang dihitung. Metode kedua hanya melibatkan piksel-piksel dengan warna unik. Piksel-piksel ini dapat ditentukan secara efisien menggunakan *hash-table* yang menggunakan *chaining* untuk resolusi yang sama (bertabrakan) dan fungsi seragam *hash* dalam bentuk: $h_a(x) = (\sum_{i=1}^3 a_i x_i) \bmod m$, dimana $x = (x_1, x_2, x_3)$ menyatakan komponen piksel merah (x_1), hijau (x_2), dan biru (x_3), m adalah jumlah warna primer, dan elemen dari urutan $a = (a_1, a_2, a_3)$ dipilih secara acak dari set $\{0, 1, \dots, m-1\}$. Metode *subsampling* kedua ini mengurangi data citra dimana kebanyakan citra mengandung warna-warna yang sama dalam jumlah yang sangat besar.

2. Sample Weighting

Kelemahan dari metode kedua *subsampling* di atas adalah metode ini mengabaikan distribusi warna dari citra asli. Untuk mengatasi hal ini, setiap titik diberi bobot secara proposional sesuai frekuensinya. Langkah pembobotan ini secara umum menghasilkan histogram warna satu dimensi. Bobot dinormalisasi dengan jumlah piksel pada citra untuk menghindari ketidakstabilan numerik pada perhitungan. Oleh karena itu, algoritma klastering *k-means* konvensional perlu dimodifikasi.

3. Sort-means Algorithm

Tahapan algoritma klastering *k-means* melibatkan banyak perhitungan jarak yang redundan. Berikut adalah algoritma klastering *k-means* yang dimodifikasi untuk kuantisasi citra:

input: $X = \{x_1, x_2, \dots, x_{N'}\} \in R^D$ ($N' \times D$ input data set)

$W = \{w_1, w_2, \dots, w_{N'}\} \in [0,1]$ (N' bobot)

output: $C = \{c_1, c_2, \dots, c_K\} \in R^D$ (K klaster pusat)

Pilih random subset C dari X sebagai inisial set dari klaster pusat

while kriteria terminasi tidak membaca **do**

Hitung jarak berpasangan antara pusat klaster;

for ($i = 1; i \leq K; i = i + 1$) **do**

for ($j = 1 + 1; j \leq K; j = j + 1$) **do**

$d[i][j] = d_{ji} = \|c_i - c_j\|^2$

end

Buat matrik M ukuran $K \times K$ dimana setiap baris i adalah permutasi dari 1, 2, ..., K yang mewakili klaster dengan urutan jarak menaik dari pusat dari c_i ;

for ($i = 1; i \leq N'; i = i + 1$) **do**

Nyatakan S_p sebagai klaster dimana x_i dimasukkan pada iterasi sebelumnya;

$P = m[i];$

$min_dist = prev_dist = \|x_i - c_p\|$

Update center terdekat jika diperlukan;

for ($j = 2; j \leq K; j = j + 1$) **do**

$t = m[p][j];$

if $d[p][t] \geq 4 \text{ prev_dist}$ **then**

Tidak ada pusat terdekat lagi. Stop checking; break;

end

$dist = \|x_i - c_t\|^2$

if $dist \leq min_dist$ **then**

c_t lebih dekat x_i dari pada c_p ;

$min_dist = dist;$

$m[i] = t;$

end

end

end

Hitung ulang pusat klaster;

for ($k = 1; k \leq K; k = k + 1$) **do**

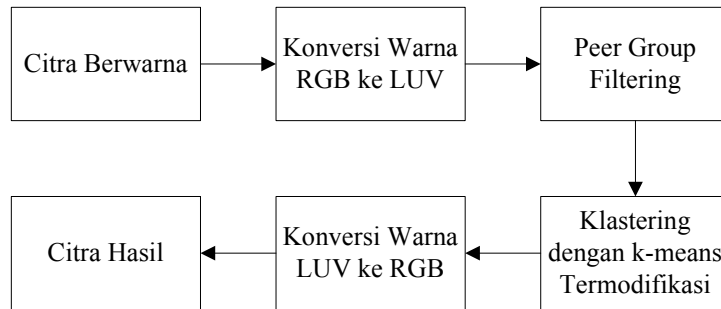
Hitung pusat baru c_t sebagai rata-rata bobot dari titik yang dekat dengan pusat baru;

$$c_k = \frac{(\sum_{m[i]=k} W_i X_i)}{\sum_{m[i]=k} W_i}$$

end

end

Diagram alir dari proses pengolahan citra pada tugas akhir ini dapat dilihat pada gambar di bawah ini.



Gambar 1. Diagram alir proses pengolahan citra dengan algoritma k-means termodifikasi

6. METODE PENELITIAN

Metode penelitian yang digunakan dalam tugas akhir ini dipaparkan sebagai berikut berikut:

1. Studi Literatur

Tahap ini akan dilakukan pencarian serta pemahaman mengenai hal-hal yang berhubungan dengan permasalahan yang ada antara lain: algoritma klastering *k-means* dan kuantisasi citra. Selain itu studi ini juga dimaksudkan untuk memahami teori musik yang berkaitan dengan aspek yang mendukung algoritma. Literatur yang digunakan meliputi buku referensi, *paper*, dan dokumentasi dari internet.

2. Studi Praktik

Studi praktik ini berkaitan dengan pembelajaran awal penggunaan teknologi sebelum digunakan untuk implementasi aplikasi. Semua yang sudah dipelajari dan dipahami pada tahap studi teori literatur diharapkan bisa dipraktikkan dengan dukungan teknologi yang ada.

3. Perancangan Perangkat Lunak dan Desain Sistem

Tahap ini meliputi perancangan sistem dengan menggunakan studi literatur dan mempelajari konsep teknologi dari perangkat lunak yang ada. Tahap ini merupakan tahap yang paling penting dimana bentuk awal aplikasi yang akan diimplementasikan dan didefinisikan. Pada tahapan ini dilakukan desain model data, desain proses-proses yang ada, dan desain antarmuka aplikasi.

4. Pengimplementasian

Tahap pengimplementasian ini, aplikasi telah mulai dibuat secara menyeluruh. Berbekal pedoman pada tahap-tahap sebelumnya yaitu desain sistem, bahasa pemrograman yang digunakan, perancangan kode program dan bagaimana penerapan pada teknologi yang akan digunakan untuk membuat sistem.

5. Uji Coba dan Evaluasi

Implementasi yang sudah selesai, akan dievaluasi dengan melakukan skenario berdasarkan alur yang telah dibuat. Jika terdapat kekurangan ataupun kesalahan dapat diperbaiki demi kelayakan keberhasilan aplikasi sesuai dengan tujuan pembuatan tugas akhir ini.

6. Penyusunan Buku Tugas Akhir

Tahap ini dilakukan untuk pembuatan laporan dari semua dasar teori dan metode yang digunakan serta hasil-hasil yang diperoleh selama pengerjaan tugas akhir. Laporan tugas akhir ini akan dibagi menjadi beberapa bab sebagai berikut:

- Bab I, pendahuluan, berisi latar belakang, permasalahan, tujuan, batasan permasalahan, metodologi, dan sistematika penulisan.
- Bab II, landasan teori, akan dibahas dasar ilmu yang mendukung pembahasan tugas akhir ini.
- Bab III, desain aplikasi.
- Bab IV, implementasi dari aplikasi yang telah dibuat, akan dilakukan pembuatan aplikasi yang dibangun dengan komponen-komponen yang telah ada yang sesuai dengan permasalahan dan batasan yang telah dijabarkan pada bab pertama.
- Bab V, uji coba dan analisa hasil, akan dilakukan uji coba berdasarkan parameter-parameter yang ditetapkan, dan kemudian dilakukan analisis terhadap hasil uji coba tersebut.
- Bab VI, penutup, berisi simpulan yang dapat diambil dari tugas akhir ini beserta saran untuk pengembangan lanjut.

7. JADWAL KEGIATAN

Tugas akhir ini diharapkan bisa dikerjakan menurut jadwal sebagai berikut:

No.	Tahapan	Bulan ke-											
		1			2			3			4		
1.	Studi Literatur												
2.	Perancangan Perangkat Lunak												
3.	Implementasi Perangkat Lunak												
4.	Uji Coba dan Evaluasi												
5.	Penyusunan Laporan Penelitian												

Keterangan: Bulan ke-1 dimulai pada Maret 2012.

8. DAFTAR PUSTAKA

- H. Kasuga, H. Yamamoto, M. Okamoto, Color quantization using the fast k-means algorithm, *Systems and Computers in Japan* 31 (8) (2000) 33–40.
- Y.-L. Huang, R.-F. Chang, A fast finite-state algorithm for generating RGB palettes of color quantized images, *Journal of Information Science and Engineering* 20 (4) (2004) 771–782.
- Y.-C. Hu, M.-G. Lee, K-means based color palette design scheme with the use of stable flags, *Journal of Electronic Imaging* 16 (3)

- (2007) 033003.
4. Y.-C. Hu, B.-H. Su, Accelerated k-means clustering algorithm for colour image quantization, Imaging Science Journal 56 (1) (2008) 29–40.
 5. Z. Xiang, Color image quantization by minimizing the maximum intercluster distance, ACM Transactions on Graphics 16 (3) (1997) 260–276.

LEMBAR PENGESAHAN

Surabaya, 20 Februari 2012
Mengetahui/Menyetujui

Dosen pembimbing I

Yudhi Purwananto, S.Kom., M.Kom.
NIP. 19700714 199703 1 002

Dosen pembimbing II

Rully Soelaiman, S.Kom., M.Kom.
NIP. 19700219 199402 1 001