

PEMANFAATAN METODE K-NEAREST NEIGHBOR PADA KLASIFIKASI IMAGE BERDASARKAN POLA FITUR DAN TEKSTUR

Nurul Fuad

Dosen Program Studi Teknik Informatika, Fakultas Teknik, Universitas Islam Lamongan

Jl. Veteran No. 53 A Lamongan

Telp. (0322) 324706

E-mail: nurulfuad2@gmail.com

ABSTRACT

Content-based image search can use Content Based Image Retrieval (CBIR). CBIR works by measuring the similarity of query images with all the images in the database so that the query cost is directly proportional to the number of images in the database. Limiting the range of image search by way of classification is one way to reduce the query cost on CBIR. Application of K-Nearest Neighbor method aims to classify the image as well as to measure the level of accuracy and time of classification. In this study built a software that can extract the color and texture features of an image by using the Color Histogram method and the Edge Histogram Descriptor. The results of the feature extraction process are then used by the software in the learning process and classification by the K-Nearest Neighbor method. The software is built with structured analysis and design methods then implemented using VB.net programming language

The final result of classification is then tested with parameter level accuracy and classification time. The test results show that the combination of color and texture features provides a higher level of accuracy than classification based on features and textures but requires longer classification time.

Keywords: Image classification, feature extraction, k-nearest neighbor

PENDAHULUAN

Pencarian image dengan menggunakan *keyword* yang sering memberikan hasil yang ambigu mendorong lahirnya teknologi *Content Based Image Retrieval* (CBIR). CBIR dapat membantu proses pencarian gambar berdasarkan *content* gambar tersebut. CBIR bekerja dengan cara menerima masukan *query* berupa gambar dan membandingkan gambar *query* dengan semua gambar yang ada di dalam database kemudian menampilkan hasil *query* secara terurut berdasarkan ukuran kemiripan antara gambar *query* dengan gambar pada database[8].

Sayangnya, jumlah gambar dalam database yang sangat banyak akan meningkatkan *query cost* pada CBIR. Salah satu solusinya adalah membatasi *range* pencarian dengan melakukan klasifikasi gambar. Klasifikasi gambar tidak hanya meningkatkan akurasi, tetapi juga meningkatkan kecepatan *query* pada *image retrieval*[12].

Untuk melakukan klasifikasi terhadap image, harus ditetapkan terlebih dahulu ciri atau fitur yang digunakan sebagai pembeda antara gambar yang satu dengan gambar yang lain. Salah satu metode yang dapat digunakan untuk

mengklasifikasi gambar berdasarkan fitur yang dimilikinya adalah *K-Nearest Neighbor*.

Dalam penelitian penelitian ini, yang menjadi pokok pembahasan adalah bagaimana menerapkan *K-Nearest Neighbor* untuk klasifikasi gambar *landscape* serta bagaimana tingkat akurasi dan waktu klasifikasi dari *K-Nearest Neighbor classifier* tersebut. Agar penelitian tetap terfokus maka fitur yang akan diteliti adalah warna dan tekstur, metode ekstraksi fitur warna yang akan digunakan adalah *Color Histogram* dan untuk mengekstrak fitur tekstur digunakan metode *Edge Histogram Descriptor*.

Pengertian Klasifikasi adalah proses mencari sekumpulan model atau fungsi yang mendeskripsikan dan membedakan kelas-kelas data untuk tujuan agar fungsi tersebut dapat digunakan untuk memprediksi label objek yang label kelasnya tidak diketahui. Model yang dihasilkan adalah berdasarkan hasil analisis terhadap sekumpulan data *learning* [4].

Fitur dari sebuah gambar adalah karakteristik atau atribut dari sebuah gambar yang dapat membedakannya dari gambar yang lain [11].

Pada referensi [5] disebutkan bahwa jika diinginkan sebuah sistem yang dapat membedakan objek-objek dengan tipe yang berbeda, maka pertama-tama harus ditetapkan karakteristik objek yang dapat diukur sebagai parameter yang mendeskripsikan objek. Karakteristik inilah yang disebut dengan fitur.

Dari beberapa definisi di atas maka ekstraksi fitur dapat didefinisikan sebagai proses mengekstrak karakteristik deskriptif dari sebuah gambar dengan menggunakan metode tertentu.

Fitur yang dipilih harus mampu mendeskripsikan objek yang diwakilinya dengan baik. Adapun kriteria fitur yang baik menurut [5] adalah sebagai berikut:

1) **Discrimination**

Fitur diharapkan mampu memberikan perbedaan nilai yang signifikan untuk objek-objek yang tidak berada dalam satu kelas.

2) **Reliability**

Fitur diharapkan mampu memberikan perbedaan nilai yang minimal untuk objek-objek yang berada dalam satu kelas yang sama.

3) **Independence**

Jika digunakan beberapa fitur, sebaiknya fitur-fitur tersebut tidak saling berkorelasi satu sama lain, artinya nilai fitur yang satu tidak mempengaruhi nilai fitur yang lain.

4) **Small Number**

Fitur diharapkan memiliki dimensionalitas yang kecil sehingga jika direpresentasikan sebagai vektor, maka vektor yang dihasilkan akan memiliki jumlah elemen yang sesedikit mungkin.

Warna merupakan fitur yang paling ekspresif dibandingkan dengan fitur visual yang lain [1]. Warna juga merupakan fitur yang paling banyak digunakan dalam *image retrieval* [8]. Salah satu metode yang dapat digunakan untuk mengekstrak fitur warna dari sebuah gambar adalah dengan menggunakan metode *Color Histogram*.

Histogram dari sebuah gambar digital dengan *gray level* dalam rentang [0, L-1] merupakan fungsi

$$h(r_k) = n_k \tag{1}$$

di mana r_k adalah *gray level* ke- k dan n_k adalah jumlah *pixel* dalam gambar yang memiliki *gray level* ke- k [7]. Umumnya histogram mengalami proses normalisasi dengan cara membagi nilai

tiap *bin* dengan jumlah seluruh *pixel* yang ada dalam gambar yang diwakili oleh variabel n . dengan demikian, histogram ternormalisasi dari sebuah gambar dapat dinyatakan dengan persamaan berikut:

$$P(r_k) = n_k / n \tag{2}$$

Ruang warna merepresentasikan warna dalam sistem koordinat sehingga tiap warna direpresentasikan sebagai titik dalam ruang warna tersebut [7]. Dalam ruang warna RGB tiap warna direpresentasikan oleh sebuah titik dalam ruang vektor 3 dimensi dengan sumbu R, G dan B yang menyatakan derajat warna merah, hijau dan biru yang menyusun warna tersebut.

Keuntungan menggunakan ruang warna RGB adalah umumnya gambar digital tersedia dalam format bitmap yang menggunakan ruang warna RGB. Oleh karena itu untuk membuat histogram dari gambar digital tidak harus melewati proses konversi ke ruang warna RGB terlebih dahulu.

Fitur warna yang diekstrak diharapkan memiliki dimensionalitas yang seminimal mungkin. Oleh karena itu, histogram dibentuk setelah melalui proses kuantisasi untuk mereduksi jumlah *gray level*.

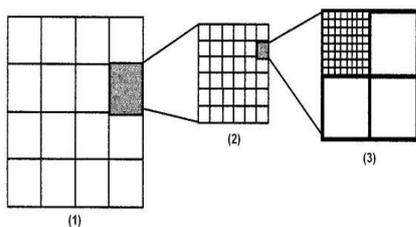
Dalam *Uniform Quantization*, tiap sumbu dalam ruang warna dikuantisasi secara terpisah. Tiap sumbu dibagi-bagi menjadi beberapa segmen yang ukurannya sama [10]. *Uniform Quantization* dinyatakan dengan persamaan sebagai berikut [6]:



$$\tag{3}$$

Salah satu keuntungan menggunakan metode *Uniform Quantization* yaitu waktu yang dibutuhkan untuk proses kuantisasi adalah yang paling cepat dibandingkan dengan menggunakan metode kuantisasi yang lain. [10].

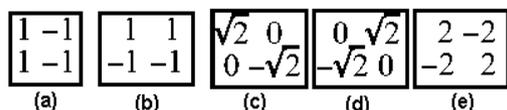
Edge Histogram Descriptor (EHD) menggambarkan distribusi spasial *edge*. Distribusi *edge* merupakan karakteristik tekstur yang berguna dalam proses *image matching* bahkan dengan kondisi di mana gambar tidak memiliki tekstur yang homogen [1]. Langkah-langkah untuk mengekstrak fitur tekstur dengan menggunakan metode EHD adalah sebagai berikut [3]:



Gambar 1. Ilustrasi Ekstraksi Fitur Tekstur dengan Menggunakan EHD

- Bagi gambar menjadi 4x4 region yang sama besar
- Bagi tiap region menjadi *sub-block* dengan ukuran yang sama
- Tiap *sub-block* dibagi menjadi 2x2 partisi
- Untuk setiap *sub-block*, nilai dalam tiap partisi dirata-ratakan sehingga tiap *sub-block* dapat diperlakukan sebagai gambar 2x2 *pixel*
- Terapkan *edge detector* pada tiap *sub-block*. Sebuah *sub-block* dinyatakan sebagai *edge block* jika hasil operasi *sub-block* dengan *edge detector* melebihi nilai *edge threshold* yang telah ditetapkan sebelumnya.
- Buat histogram yang menggambarkan distribusi *edge block*. Histogram berukuran 240 bin, diperoleh dari 16 region x 5 *edge orientation* x 3 komponen warna (R,G,B)

Adapun *edge detector* standar yang digunakan pada MPEG-7 ISO/IEC terdiri dari 5 operator, masing-masing untuk mendeteksi *edge* dengan orientasi vertikal, horisontal, 45 derajat, 135 derajat dan *isotropic* atau tak berarah.

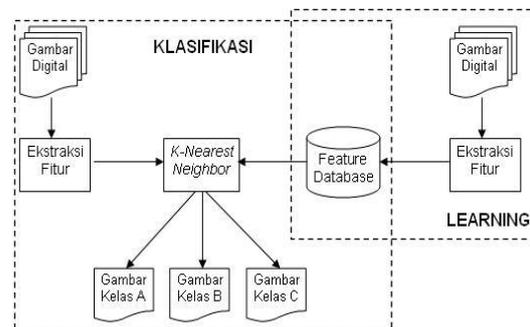


Gambar 2. Edge detector standar MPEG-7 ISO/IEC

Beberapa parameter EHD dapat mengalami penyesuaian tergantung pada kebutuhan. Salah satu contohnya adalah dalam [2] gambar tidak dibagi menjadi 4x4 region, *edge detector* yang digunakan adalah *Sobel operator*, sedangkan nilai *edge threshold* ditetapkan sama dengan 100. Pada penelitian ini, proses ekstraksi fitur dengan metode EHD dilakukan terhadap gambar yang dibagi menjadi 4x4, 3x3 dan 2x2 region. *Edge detector* yang digunakan

adalah *edge detector* standar MPEG-7 ISO/IEC dengan nilai *edge threshold* = 100.

METODE PENELITIAN



Gambar 3. Framework Sistem

Perangkat lunak akan terdiri dari 2 buah proses utama yang saling berkaitan, yaitu proses *learning* dan proses klasifikasi.

Input untuk proses *learning* adalah kumpulan *learning image* yang sudah diketahui label kelasnya. Adapun output yang dihasilkan adalah fitur gambar yang disimpan dalam sebuah *feature database*.

Input untuk proses klasifikasi adalah gambar yang akan diklasifikasikan dan fitur *learning image* yang tersimpan dalam *feature database*. Adapun outputnya adalah label kelas gambar yang diinputkan.

Dengan demikian jelaslah bahwa proses *learning* harus dilakukan sebelum proses klasifikasi sebab output dari proses *learning* menjadi salah satu input yang dibutuhkan dalam proses klasifikasi.

Pada proses *learning*, input perangkat lunak adalah kumpulan *learning image* yang telah diketahui label kelasnya. Semua *learning image* akan diekstrak fiturnya dengan menggunakan metode *Color Histogram* dan *Edge Histogram Descriptor*. Hasil dari proses ekstraksi fitur akan disimpan dalam sebuah *feature database*.

Berikut ini adalah langkah-langkah yang dilakukan dalam proses *learning*:

- 1) Membaca data bitmap *learning image*
- 2) Melakukan ekstraksi fitur wana dengan metode *Color Histogram*
- 3) Melakukan ekstraksi fitur tekstur dengan metode *Edge Histogram Descriptor*

Pada proses klasifikasi, input perangkat lunak adalah *testing image* yang belum diketahui label kelasnya. *Testing image* akan

mengalami proses ekstraksi fitur. Hasil dari proses ekstraksi fitur akan dibandingkan dengan fitur yang tersimpan dalam *feature database* untuk menentukan label kelas *testing image*.

Berikut ini adalah langkah-langkah yang dilakukan dalam proses klasifikasi:

- 1) Membaca data bitmap *testing image*
- 2) Melakukan ekstraksi fitur warna dengan metode Color Histogram
- 3) Melakukan ekstraksi fitur tekstur dengan metode Edge Histogram Descriptor
- 4) Mengukur perbedaan antara hasil ekstraksi fitur *testing image* dengan

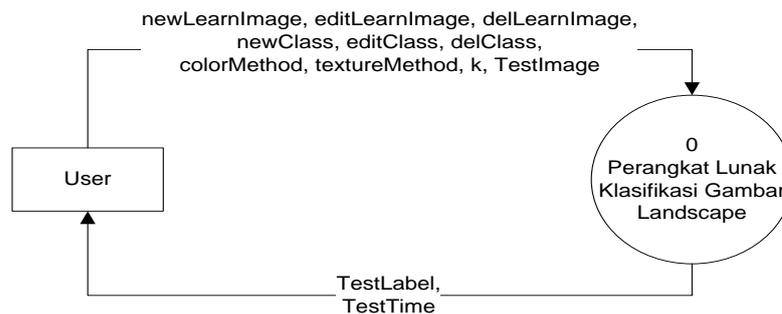
fitur learning image yang tersimpan dalam *feature database*

- 5) Melakukan voting untuk menentukan label kelas image

Perancangan perangkat lunak dilakukan dengan metode terstruktur. Adapun model-model yang dihasilkan adalah sebagai berikut:

1. Diagram Konteks
2. *Data Flow Diagram* (DFD)
3. Spesifikasi Proses
4. Kamus Data.

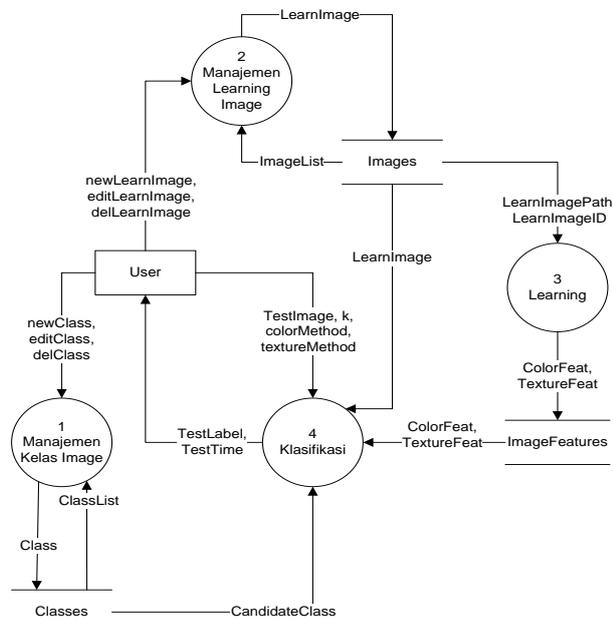
Berikut ini adalah diagram konteks dan *data flow diagram* dari sistem yang telah dibangun:



Gambar 4. Diagram Konteks

Diagram konteks hanya menggambarkan input dan output sistem secara keseluruhan. Adapun proses-proses inti yang ada di dalam

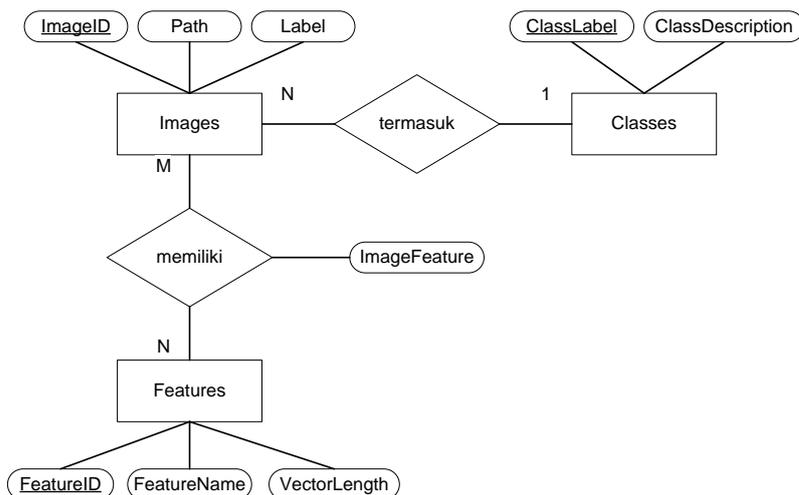
sistem serta aliran data yang terjadi antar proses digambarkan dalam *data flow diagram* berikut ini



Gambar 5. DFD Level 1

Adapun untuk keperluan penyimpanan data, maka diperlukan sebuah basis data. Rancangan

basis data dapat dilihat pada diagram Entity-Relationship berikut ini:



Gambar 6. Entity Relationship Diagram

Adapun hasil implementasi dari rancangan basis data pada diagram ER di atas dapat dilihat melalui struktur tabel berikut ini:

Tabel 1. Struktur tabel 'Classes'

No	Field	Type Data	Keterangan
1	ClassLabel	String	Primary key, label dari kelas
2	ClassDescription	String	Deskripsi kelas

Tabel 2. Struktur tabel 'Features'

No	Field	Type Data	Keterangan
1	FeatureID	Number	Primary key
2	FeatureName	String	Nama fitur
3	VectorLength	Number	Jumlah elemen vektor

Tabel 3. Struktur tabel 'Images'

No	Field	Type Data	Keterangan
1	ImageID	Number	Primary key Label dari <i>learning image</i> , foreign key yang mengacu ke
2	Label	String	

3	Path	String	field 'ClassLabel' pada tabel 'Classes' Path absolute ke file <i>learning image</i>
---	------	--------	---

Tabel 4. Struktur tabel 'ImageFeatures'

No	Field	Type Data	Keterangan
1	ImageID	Number	Foreign key yang mengacu ke field 'ImageID' pada tabel 'Images'
2	FeatureID	Number	Foreign key yang mengacu ke field 'FeatureID' pada tabel 'Feature'
3	ImageFeature	String	Fitur yang diekstrak dari image dan disimpan dalam bentuk string.

HASIL DAN PEMBAHASAN

Data yang digunakan untuk pengujian berasal dari *Best Photo Collection CD* yang terdiri dari 200 gambar yang terbagi dalam 4 kelas yaitu *canyon, mountain, sunset* dan *waterfall*.

Tabel 5. Data untuk pelatihan dan pengujian

Kelas	Jumlah data learning	Jumlah data testing
Canyon	25 gambar	25 gambar
Mountain	25 gambar	25 gambar
Sunset	25 gambar	25 gambar
Waterfall	25 gambar	25 gambar
Total	100 gambar	100 gambar

Parameter yang diuji adalah tingkat akurasi dan waktu klasifikasi. Tingkat akurasi dihitung dengan menghitung persentase gambar yang dapat diklasifikasikan dengan benar dalam satuan persen sedangkan waktu klasifikasi dihitung mulai proses pembacaan data bitmap sampai menghasilkan label kelas dan diukur dalam satuan detik per gambar.

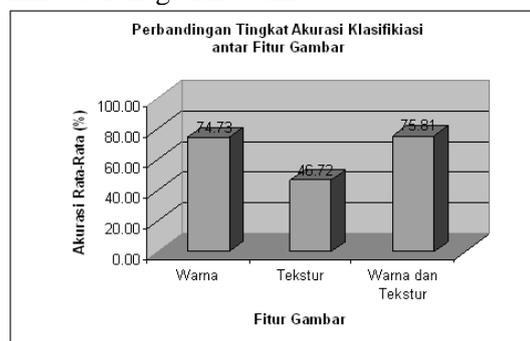
Total jumlah percobaan yang dilakukan adalah 375 percobaan yang terdiri dari:

- 1) 75 percobaan klasifikasi berdasarkan fitur warna yang diekstrak dengan metode *Color Histogram* 256, 4 dan 32 bin untuk nilai *k* masing-masing 1 sampai 25.
- 2) 75 percobaan klasifikasi berdasarkan fitur tekstur yang diekstrak dengan metode *Edge Histogram Descriptor* 4x4, 3x3 dan 2x2 region untuk nilai *k* masing-masing 1 sampai 25.
- 3) 225 percobaan klasifikasi berdasarkan fitur warna dan tekstur yang diekstrak dengan 9 kombinasi metode ekstraksi fitur warna dan tekstur untuk nilai *k* masing-masing 1 sampai 25.

Variabel Klasifikasi			Jumlah Percobaan
Fitur	Metode Ekstraksi Fitur	Nilai k	
Warna	Color Histogram 256 bin	1 sampai 25	25
	Color Histogram 64 bin	1 sampai 25	25
	Color Histogram 32 bin	1 sampai 25	25
Tekstur	Edge Histogram 4x4 Region	1 sampai 25	25
	Edge Histogram 3x3 Region	1 sampai 25	25
	Edge Histogram 2x2 Region	1 sampai 25	25
Warna dan Tekstur	Color Histogram 256 bin + Edge Histogram 4x4 Region	1 sampai 25	25
	Color Histogram 256 bin + Edge Histogram 3x3 Region	1 sampai 25	25
	Color Histogram 256 bin + Edge Histogram 2x2 Region	1 sampai 25	25
	Color Histogram 64 bin + Edge Histogram 4x4 Region	1 sampai 25	25
	Color Histogram 64 bin + Edge Histogram 3x3 Region	1 sampai 25	25
	Color Histogram 64 bin + Edge Histogram 2x2 Region	1 sampai 25	25
	Color Histogram 32 bin + Edge Histogram 4x4 Region	1 sampai 25	25
	Color Histogram 32 bin + Edge Histogram 3x3 Region	1 sampai 25	25
	Color Histogram 32 bin + Edge Histogram 2x2 Region	1 sampai 25	25
Total Jumlah Percobaan			375

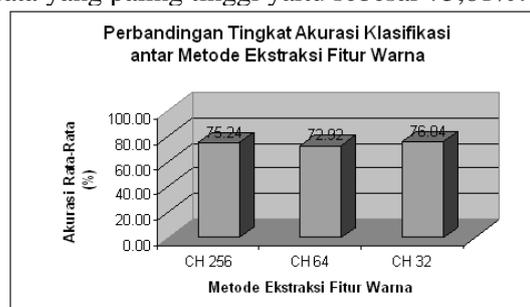
Gambar 7. Skenario Pengujian

Dalam tiap percobaan digunakan 100 gambar testing. Adapun hasil pengujian dapat dilihat dalam grafik berikut:



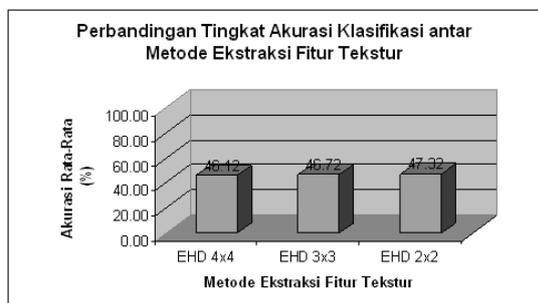
Gambar 8. Grafik Perbandingan Tingkat Akurasi Klasifikasi Antar Fitur

Fitur gambar yang digunakan dalam klasifikasi akan mempengaruhi tingkat akurasi klasifikasi. Klasifikasi dengan berdasarkan fitur warna dan tekstur memiliki tingkat akurasi rata-rata yang paling tinggi yaitu sebesar 75,81%.



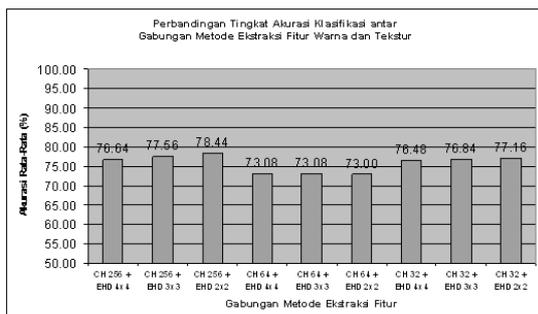
Gambar 9. Grafik Perbandingan Tingkat Akurasi Klasifikasi Aantar Metode Ekstraksi Fitur Warna

Metode ekstraksi fitur warna berpengaruh terhadap tingkat akurasi klasifikasi. Adapun metode ekstraksi fitur warna yang memberikan tingkat akurasi klasifikasi tertinggi adalah *Color Histogram 32 bin (CH 32)* yaitu sebesar 76,04 %.



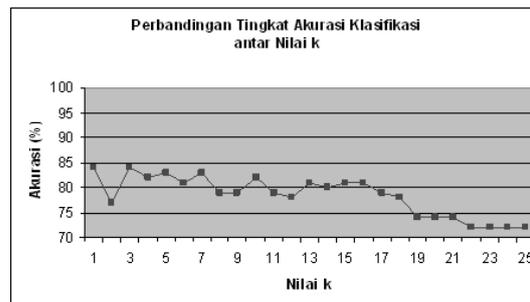
Gambar 10. Grafik Perbandingan Tingkat Akurasi Kaslifkasi Antar Metode Ekstraksi Fitur

Metode ekstraksi fitur tekstur berpengaruh terhadap tingkat akurasi klasifikasi. Adapun metode ekstraksi fitur tekstur yang memberikan tingkat akurasi klasifikasi tertinggi adalah *Edge Histogram Descriptor 2x2 Region (EHD 2x2)* yaitu sebesar 47,32 %.



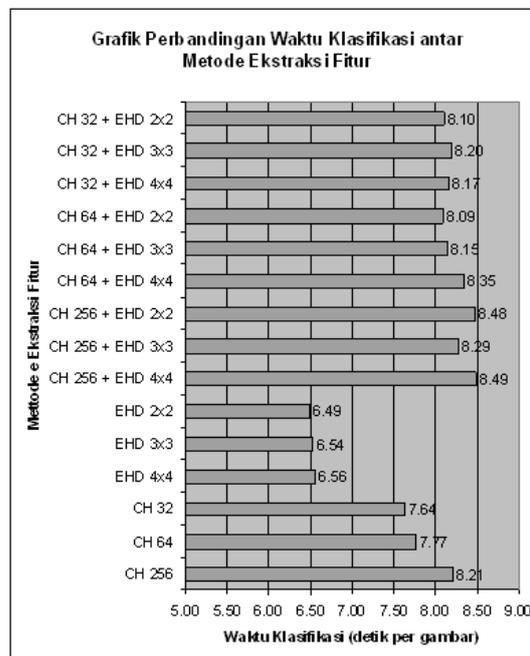
Gambar 11. Grafik Perbandingan Tingkat Akurasi Klasifikasi Antar Gabungan Metode Ekstraksi Fitur Warna dan Tekstur

Pada grafik di atas terlihat bahwa gabungan metode ekstraksi fitur warna dan tekstur yang memberikan tingkat akurasi klasifikasi paling tinggi adalah *Color Histogram 256 bin (CH 256)* dan *Edge Histogram Descriptor 2x2 Region (EHD 2x2)*. Gabungan kedua metode ini memberikan tingkat akurasi rata-rata sebesar 78,44%.



Gambar 12. Grafik Perbandingan Tingkat Akurasi Klasifikasi Antar Nilai K

Grafik di atas menunjukkan bahwa tingkat akurasi klasifikasi berfluktuasi tergantung nilai *k* yang diinputkan namun cenderung menurun seiring naiknya nilai *k*. Untuk klasifikasi berdasarkan fitur warna dengan metode *ColorHistogram 256 bin* dan fitur tekstur dengan metode *Edge histogram 2x2 Region*, tingkat akurasi paling tinggi diperoleh dengan nilai *k=1* dan *k=3* yaitu sebesar 84 %.



Gambar 13. Grafik perbandingan waktu Klasifikasi

Waktu klasifikasi tersingkat adalah klasifikasi berdasarkan fitur tekstur yang diekstrak dengan metode *Edge Histogram Descriptor 2x2 Region (EHD 2x2)* yaitu selama 6.49 detik per gambar sedangkan yang membutuhkan waktu paling lama adalah klasifikasi berdasarkan warna dan tekstur yang diekstrak dengan metode *Color Histogram 256 bin* dan *Edge Histogram Descriptor 4x4 Region*

(CH 256 + EHD 4x4) yaitu selama 8.49 detik per gambar.

KESIMPULAN

Dari hasil akhir sistem ini serta dari hasil uji coba yang telah dilakukan, dapat ditarik kesimpulan bahwa fitur gambar, metode ekstraksi fitur dan jumlah neighbor yang melakukan voting mempengaruhi tingkat akurasi klasifikasi gambar dengan metode *K-Nearest Neighbor*.

Pola Fitur yang memberikan tingkat akurasi paling tinggi adalah fitur warna dan tekstur sedangkan metode ekstraksi fitur yang memberikan tingkat akurasi tertinggi adalah kombinasi *Color Histogram 256 bin* dan *Edge Histogram Descriptor 2x2*. Tingkat akurasi klasifikasi berfluktuasi tergantung jumlah *neighbor* yang melakukan *voting* namun cenderung menurun seiring bertambahnya jumlah *neighbor* yang melakukan *voting*.

Adapun klasifikasi yang membutuhkan waktu paling singkat adalah klasifikasi dengan fitur tekstur, disusul kemudian oleh klasifikasi dengan fitur warna dan yang membutuhkan waktu paling lama adalah klasifikasi berdasarkan fitur warna dan tekstur.

Dalam tiap percobaan digunakan 100 gambar testing. Adapun hasil pengujian dapat dilihat dalam grafik berikut:

Fitur gambar yang digunakan dalam klasifikasi akan mempengaruhi tingkat akurasi klasifikasi. Klasifikasi dengan berdasarkan fitur warna dan tekstur memiliki tingkat akurasi rata-rata yang paling tinggi yaitu sebesar 75,81%.

Metode ekstraksi fitur warna berpengaruh terhadap tingkat akurasi klasifikasi. Adapun metode ekstraksi fitur warna yang memberikan

tingkat akurasi klasifikasi tertinggi adalah *Color Histogram 32 bin* (CH 32) yaitu sebesar 76,04 %.

REFERENSI

- [1] CBIR:Features, <http://www.ee.columbia.edu/~x1x/course/s/vis-hw3/page2.html>, didownload pada April 2007
- [2] CBIR: Texture Features, 2007, www.cs.auckland.ac.nz/compsci708s1c/, didownload pada tanggal 11 April 2007
- [3] Jiawei Han, Micheline Kamber, 2002, "Data Mining Concept and Techniques", Academic Press
- [4] Kenneth R. Castleman, 1996, "Digital Image Processing", Prentice Hall
- [5] Maher A. Sid Ahmed, 1995, "Image Processing: Theory, Algorithm and Architecture", McGrawHill
- [6] Rafael C. Gonzales, Richard E. Woods, 2002, "Digital Image Processing", Pentice Hall
- [7] Sundaram RMD, "Image Mining, Intricacies and Innovations", <http://www.amrita.edu/cde/>, didownload pada tanggal 11 April 2007
- [8] Teknomo, Kardi. K-Nearest Neighbors Tutorial, 2006, <http://people.revoledu.com/kardi/tutorial/KNN/>, didownload pada tanggal 11 Desember 2006
- [9] Uniform Quantization, 2007, <http://www.cs.wpi.edu/~matt/courses/cs563/>, didownload pada tanggal 11 April 2007