

Aplikasi Content Based Image Retrieval

by Suhendro Irianto

Submission date: 16-Feb-2020 03:59PM (UTC+0800)

Submission ID: 1258149180

File name: APLIKASI_CONTENT_BASED_IMAGE_RETRIEVAL.pdf (2.52M)

Word count: 29170

Character count: 186470

- Aplikasi
- Content Based
- Image Retrieval

Hak cipta pada penulis
Hak penerbitan pada penerbit
Tidak boleh diproduksi sebagian atau seluruhnya dalam bentuk apapun
Tanpa izin tertulis dari pengarang dan/atau penerbit

Kutipan Pasal 72 :

Sanksi pelanggaran Undang-undang Hak Cipta (UU No. 10 Tahun 2012)

5

1. Barang siapa dengan sengaja dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam Pasal 2 ayat (1) atau Pasal (49) ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling singkat 1 (satu) bulan dan/atau denda paling sedikit Rp. 1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan atau denda paling banyak Rp. 5.000.000.000,00 (lima miliar rupiah)
2. Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu Ciptaan atau hasil barang hasil pelanggaran Hak Cipta atau Hak Terkait sebagaimana dimaksud ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp. 500.000.000,00 (lima ratus juta rupiah)



Aplikasi Content Based Image Retrieval

**Dona Yulawati
Agus Rahardi
Nisar
Suhendro Yusuf Irianto
Wasilah**

Perpustakaan Nasional RI: Katalog Dalam Terbitan (KDT)

APLIKASI CONTENT BASED IMAGE RETRIEVAL

**Dona Yuliatwati
Agus Rahardi
Nisar
Suhendro Yusuf Irianto
Wasilah**

Rancang Sampul & Penata Isi
Aura Creative

ISBN: 978-602-51690-8-3

Cetakan November 2018
XIV+ 141 hlm. ; 15.5 x 23 cm

**Penerbit
Darmajaya (DJ) Press**

Alamat :
Kampus IIB DARMAJAYA
Jl. Zainal Abidin Pagar Alam No 93,
Bandar Lampung 35142, INDONESIA

Hak Cipta dilindungi Undang-Undang
All Rigths Reserved.

5

Dilarang mengutip atau memperbanyak sebagian
atau seluruh isi buku ini tanpa izin tertulis dari penerbit

PRAKATA

Content Based Image Retrieval merupakan salah satu cabang dari bidang image processing dan computer vision. Image processing adalah bidang termasuk metode-metode untuk pengolahan, analisa dan pemahaman tentang citra digital dan high dimensional data dari dunia nyata untuk menghasilkan angka atau simbol-simbol informasi untuk pengambil keputusan. Buku ini ditujukan untuk mahasiswa yang mengambil mata kuliah pengolahan citra dan computer vision. Buku ini berisi teori-teori dan dari hasil penelitian yang dilakukan oleh penulis. Buku ini digunakan bagi siapa yang ingin mempelajari bidang pengolahan citra digital atau *image processing*. Aplikasi Content Based Image Retrieval menyajikan hasil-hasil penelitian yang berkaitan dengan *image retrieval*.

Buku ini terdiri dari 11 bab, menyediakan konsep dan teori serta definisi sampai aplikasi pengolahan citra dalam kehidupan sehari-hari. Pada bab pertama materi yang diberikan content based image retrieval, untuk bab 2 jenis-jenis citra digital, bab 3 mesin pencarian, bab 4 penerapan pengolahan citra, bab 5 dasar pengolahan citra, bab 6 perbaikan citra (*image enhancement*), bab 7 analisa citra, bab 8 kompresi data citra, bab 9 pengenalan pola, bab 10 aplikasi search engine berbasis cbir, bab 11 digital watermarking.

60

Penulis menyadari masih banyak kekurangan dalam penyusunan buku ini. Saran dan kritik akan sangat membantu dalam memperbaiki penyusunan buku ini

Bandar lampung, Agustus 2018

Penulis

DAFTAR ISI

KATA PENGANTAR.....	v
DAFTAR ISI	vi
DAFTAR GAMBAR	xiii
BAB 1. CONTENT BASED IMAGE RETRIEVAL.....	1
1.1 Fitur Citra Dalam CBIR1	3
1.1 Warna.....	3
1.1.2 Tekstur.....	3
1.1.3 Fitur Bentuk.....	5
Query-by-Example	5
1.2 Database Citra Digital	6
1.3 Matching Image Query	7
1.3.1 Matching Berdasarkan Fitur Warna	7
1.3.2 Matching Berbasis Fitur Tekstur.....	8
1.3.3 Matching Berbasis Fitur Bentuk.....	8
1.4 Matching Citra JPEG Berbasis Content	10
1.5 CBIR Dalam Compressed Domain	12
Tugas mandiri.....	12

BAB 2. JENIS-JENIS CITRA DIGITAL	13
2.1 Jenis atau Tipe Citra Digital.....	13
2.2 Citra dan Pengolahan Citra	15
2.3 Konversi Antar Citra	17
2.3.1 rgb2gray.....	17
2.3.2 im2bw	18
Tugas Mandiri.....	18
BAB 3. MESIN PENCARIAN.....	19
3.1 Arsitektur Mesin Pencari.....	19
3.2 Indeks Pencarian.....	20
3.3 Mesin Query	22
3.4 Antarmuka Pencarian	23
3.5 Google	23
3.6 Pencarian Gambar	25
3.7 Pencarian Gambar Berdasarkan Text.....	27
3.8 Pencarian Gambar Berbasis konten.....	28
Tugas Mandiri.....	31
BAB 4. Penerapan Pengolahan Citra.....	32
4.1 Image Processing, Image Analysis, Image Understanding, dan Computer Vision.....	32
4.2 Penerapan Pengolahan Citra Digital	33
4.2.1 Bidang Biomedis.....	34
4.2.2 Bidang Penginderaan Jarak Jauh (Remote Sensing)	34
4.2.3 Bidang Biometrika	34

4.2.4 Bidang Fotografi.....	34
4.2.5 Bidang Desain Visual.....	35
4.2.6 Identifikasi Objek-objek pada Suatu Citra	35
4.2.7 Bidang Volumetrik.....	35
4.2.8 Meneliti Proses Dinamis	35
4.2.9 OCR.....	36
4.2.10 Bidang Pemantapan Citra dan Video	36
4.2.11 Image Retrieval atau Image Querying	36
4.3 Komponen dalam Sistem Pengolahan Citra	
Digital.....	36
Tugas Mandiri.....	37
BAB 5. DASAR PENGOLAHAN CITRA.....	38
5.1 Pengertian Citra Digital	38
5.2 Cahaya (Light), Luminance, Brightness, dan	
Contrast.....	39
5.3 Pembentukan Citra Digital.....	41
5.3.1 Akuisisi Citra	41
5.3.2 Sampling.....	42
5.3.3 Kuantisasi.....	42
5.4 Pixel dan Voxel	42
5.5 Relasi Ketetanggaan.....	43
5.6 Resolusi Citra	43
5.6.1 Resolusi Pixel	44
5.6.2 Resolusi Spasial	44
5.6.3 Resolusi Spektrum.....	44

5.6.4 Resolusi Temporal	44
5.6.5 Resolusi Radiometrik	44
5.7 Format File Citra	45
Tugas Mandiri.....	47
BAB 6. PERBAIKAN CITRA (IMAGE ENHANCEMENT)	48
6.1 Operasi Titik.....	48
6.1.1 Penyesuaian Kecerahan (Brightness Adjustment)	49
6.1.2 Negasi (Negation).....	50
6.1.3 Koreksi Gamma (Gamma Correction)	50
6.1.4 Perenggangan Kontras (Constrast Stretching).....	51
6.1.5 Pengirisan Intensitas (Intensity Slicing)	52
6.1.6 Pemisahan Bit (Bit Extraction).....	53
6.1.7 Pemampatan Rentangan (Range Compression)	54
6.1.8 Ekualisasi Histogram (Histogram Equalization)	54
6.1.9 Ekualisasi Histogram Adaptif (Adaptif Histogram Equalization).....	55
6.1.10 Penajaman Lokal	56
6.2 Operasi Geometri Citra	56
6.2.1 Translasi	56
6.2.1 Rotasi	57
6.2.2 Refleksi.....	57
6.2.3 Penskalaan	58
6.2.4 Transformasi Affine (Affine Transformation)	59
Tugas Mandiri.....	59

BAB 7. ANALISIS CITRA.....	60
7.1 Morphologi	60
7.1.1 Dilasi (Dilation).....	61
7.1.2 Erosi.....	62
7.1.3 Opening.....	63
7.1.4 Closing	64
7.1.5 Hit and Miss Transform	64
7.1.6 Thinning7.	65
1.7 Thickening	66
7.2 Edge Detection.....	66
7.2.1 Edge, Line, dan Point Detection.....	66
7.2.2 Deteksi Tepi Berdasarkan Turunan Pertama	68
7.2.3 Deteksi Tepi Berdasarkan Turunan Kedua	71
Tugas Mandiri.....	72
BAB 8. KOMPRESI DATA CITRA (IMAGE DATA- COMPRESSION).....	73
8.1 Pengantar Kompresi Data	73
8.2 Data Berlebihan (Data Redundancy).....	76
8.2.1 Coding Redundancy.....	77
8.2.2 Interpixel Redundancy.....	79
8.2.3 Psychovisual Redundancy	79
8.3 Model Kompresi Citra	80
Tugas Mandiri.....	80

BAB 9. PENGENALAN POLA	83
9.1 Pengertian Pengenalan Pola.....	83
9.2 Aplikasi Pengenalan Pola Untuk Mengenali Citra	
Sidik Jari	85
9.3 Aplikasi Pengenalan Pola Untuk Mengenali Citra	
Wajah.....	86
9.4 Aplikasi Pengenalan Pola Untuk Mengenali Citra	
Iris Mata	87
9.5 Aplikasi Pengenalan Pola Untuk Mengenali Citra	
Telapak Tangan.....	88
9.6 Aplikasi Pengenalan Pola Untuk Temu Balik Citra	88
9.7 Aplikasi Pengenalan Pola Untuk Pengenalan Tulisan	89
9.8 Aplikasi Pengenalancitra Pada Bidang Kedokteran.....	89
Tugas Mandiri.....	90
BAB 10. APLIKASI SEARCH ENGINE BERBASIS CBIR	91
10.1 Prototipe Search Engine Flora dan Fauna.....	91
10.2 Search engine	93
10.3 Content Based Image Retrieval	94
10.4 Matching Citra JPEG Berbasis Content	95
10.5 Fitur Bentuk (Shape) dan Struktur	98
10.6 Fitur Warna dan Fitur Tekstur.....	101
10.7 Desain CBIR untuk Search Engine.....	102
10.8 Tampilan Program dan Pembahasan	109
Tugas Mandiri.....	112

BAB 11. DIGITAL WATERMARKING	113
11.1 Citra Digital	114
11.2 Citra Warna	115
11.3 Format File Gambar	115
11.4 Operasi Berbasis Bingkai	116
11.5 Operator Aritmatik.....	116
11.6 Operator Aritmatik.....	117
11.7 Least Significant Bit Hiding (LSB).....	120
11.8 Metode Watermarking dan Content Based Image Retrieval.....	121
11.9 Aplikasi Watermarking.....	123
Tugas Mandiri.....	134
 DAFTAR PUSTAKA.....	 135

DAFTAR GAMBAR

Gambar 1.1	Contoh tekstur visual.....	3
Gambar 2.1	Gambar Image Binary	13
Gambar 2.2	Citra gray scale	14
36 Gambar 2.3	Citra berwarna (RGB).....	14
Gambar 2.4	Tahapan Pengolahan Citra Digital.....	15
Gambar 2.5	Contoh Image Segementation.....	16
Gambar 2.6	Contoh Image Analysis	16
Gambar 2.7	Contoh Image Recontruction.....	17
Gambar 3.1	Arsitekture Search Engine.....	20
Gambar 8.1	Source encoder dan Source decoder.....	81
Gambar 10.1	Arsitektur search engine	93
Gambar 10.2.	Diagram alur searching citra flora-fauna dengan query by example, searching citra berbasis warna dan tekstur, berbasis bentuk dan struktur	100
Gambar 10.3	Diagram Konteks	103
Gambar 10.4	DFD Level 1.....	104
Gambar 10.5	Arsitektur CBIR	104
Gambar 10.6	Flowchart Program.....	105
Gambar 10.7	Flowchart Program	106
Gambar 10.8	Flowchart Program	107

Gambar 10.9	Tampilan Add Image to Database.....	108
Gambar 10.10	Tampilan Muka Program.....	108
Gambar 10.11	Tampilan Menu Utama	109
Gambar 10.13	Pencarian Gambar Berdasarkan Warna dan Bentuk.....	111
Gambar 11.2	Proses <i>Verification</i> pada <i>Citra Digital</i>	119
Gambar 11.3	Proses penyisipan dan <i>ekstraksi watermarking</i>	122
Gambar 11.4	Sistem verifikasi TTD, karya seni dan artifact dengan teknik CBIR.....	122
Gambar 11.5	Verifikasi keaslian TTD dan artifact menggunakan watermarking dan Content Based Image Retrieval	123
Gambar 11.6	Tampilan Menu Awal	125
Gambar 11.7	Menu Steganografi.....	126
Gambar 11.8	Snapshot menyisipan/menyembunyikan logo kedalam dukomen	126
Gambar 11.9	Snapshot memunculkan kembali logo yang disembunyikan dalam dokument	127
Gambar 11.11.	Tampilan Ukuran File Citra Hidden	130
Gambar 11.12	Tampilan Ukuran File Steganografi	131

CONTENT BASED IMAGE RETRIEVAL

Kemajuan dalam teknologi digital semakin berkembang dengan banyak data dalam bentuk digital. Contoh salah satu dari data digital yaitu bentuk Image atau gambar, dengan makin banyaknya data dalam bentuk gambar maka dibutuhkan suatu aplikasi perangkat lunak untuk mengorganisir gambar tersebut. Menggunakan aplikasi lunak tertentu, dapat membantu user untuk menyimpan gambar sesuai dengan kategori yang diinginkan oleh user. Bila user membutuhkan gambar tersebut maka user dapat mencari satu persatu dari koleksi gambar yang telah disimpan dalam database dengan menggunakan pencarian gambar atau image retrieval yang disediakan oleh perangkat lunak.

Untuk fungsi image retrieval yang disediakan oleh aplikasi biasanya berbasis teks. Salah satu solusi dengan menggunakan image retrieval yang berbasis dari sebuah image atau image properties. Image retrieval yang berbasis pada image properties biasa disebut dengan nama *Content Based Image Retrieval* (CBIR). Image Properties yang menggunakan CBIR, image properties dari sebuah image diekstrak secara otomatis dengan menggunakan algoritma tertentu. Aplikasi content based image retrieval untuk saat ini telah banyak diterapkan pada bidang biomedis, kriminalitas, militer, komunikasi, budaya, pendidikan, hiburan dan pertanian.

Dalam CBIR untuk menentukan kemiripan suatu citra dalam suatu database dengan citra referensi adalah dengan menghitung *Euclidean distance*-nya. Menurut (Zhang,et.al., 2014) terdapat empat macam ukuran untuk menghitung kemiripan, yaitu berdasarkan kemiripan warna, bentuk, tekstur dan struktur. Isi (*content*) dari suatu citra terdiri dari fitur persepsi (warna, bentuk, tekstur, dan struktur), fitur semantik, peranan, pandangan, dan fitur impresi dan emosi.

Sementara itu untuk mengukur efektivitas dari pencarian gambar dengan menggunakan *precision* dan *Recall*, *precision* adalah jumlah citra yang mirip yang terpanggil dibagi dengan semua citra yang terpanggil (*retrieved*). Sedang *Recall* adalah jumlah citra yang terpanggil dibagi dengan jumlah citra yang dalam kategori, (Cho, 2004; Thamarai and Dharma,2014)

$$p = \frac{a}{z}$$

$$r = \frac{b}{y}$$

Dimana *p* adalah *precision* dan *r* adalah *recall*, sementara *a* adalah jumlah citra yang relevan atau mirip terpanggil, *z* adalah jumlah semua citra yang terpanggil, dan *y* jumlah citra relevan yang ada dalam database. Sedangkan untuk mengukur efisiensi teknik pencarian citra (*image retrieval*) dihitung dengan menggunakan persamaan berikut:

$$Error\ rate = \frac{\text{Jumlah citra yang tidak relevan yang terpanggil}}{\text{Total jumlah citra yang terpanggil}}$$

$$Retrieval - efficiency = \frac{\text{Jumlah citra relevan terpanggil}}{\text{Total jumlah citra yang terpanggil}}$$

Nilai *precision* atau *recall* sendiri tidak cukup memberikan informasi, karena nilai *recall* akan bisa dibuat 1 dengan memanggil semua citra dalam database. Demikian juga *precision* dapat dibuat tinggi dengan cara memanggil beberapa citra.

1.1 Fitur Citra Dalam CBIR

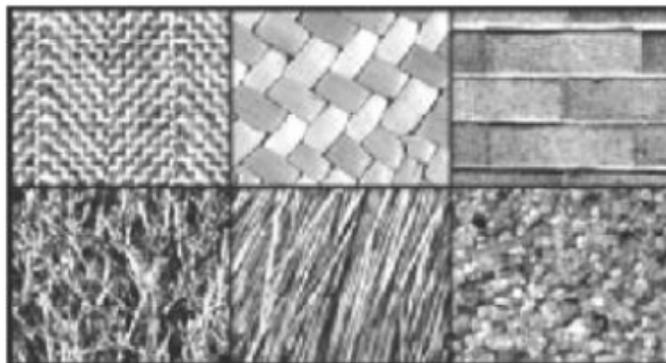
Dalam CBIR matching citra digital didasarkan fitur-fitur citra yang fitur tersebut antara lain: warna (*histograms, gridded layout, wavelets*), tekstur (*Laws, Gabor filters, local binary partition*), bentuk (*first segment the image then use statistical or structural shape similarity measures*), dan objek dan hubungan diantara objek tersebut.

1.1.1Warna

Fitur warna merupakan fitur yang paling banyak digunakan dalam pencarian citra (CBIR). Representasi dari fitur warna biasanya diwakili dengan histogram warna. Warna merupakan fitur low-level (low level feature), hal ini disebabkan karena warna sangat peka terhadap visual manusia dan berkaitan dengan model HVS.

1.1.2 Tekstur

Tekstur dapat merupakan sifat permukaan kasar atau halus, vertical atau horizontal. Tekstur dapat merupakan pattern dari suatu citra seperti keteraturan dan granualit. Sementara itu tekstur sendiri mempunyai fitur juga yaitu fitur statistic (entropy, homogeneity, kontras),⁴⁰ wavelets dan fractal. Tekstur dapat juga didefinisikan sebagai frekuensi perubahan rona pada citra atau pengulangan rona kelompok obyek yang terlalu kecil untuk dibedakan secara individual. Tekstur sering dinyatakan dengan kasar, halus dan belang-belang. Contoh pengenalan obyek berdasarkan tekstur.



Gambar 1.1 Contoh tekstur visual

Tekstur citra memberikan informasi tentang susunan spasial dari warna atau intensitas citra atau region dalam citra. Tekstur citra dapat sengaja dibuat atau secara alami terjadi pada saat citra diambil oleh kamera. Tekstur citra adalah satu cara yang dapat digunakan membantu dalam proses segmentasi citra, analisa citra dan klasifikasi citra. Untuk menganalisa tekstur suatu citra dapat dilakukan dengan dua cara yaitu dengan pendekatan statistic dan pendekatan struktur.

Pendekatan statistik adalah pendekatan tesktur citra diman teksture citra dianggap sebagai pengukuran kauntitatif dan penyusunan intensitas region. Secara umum pendekatan ini lebih udah dikerjakan dalam perhitungan dan secara luas digunakan karena tekstur alami dibuat dari suatu pattern dari elemen-elemen yang tidak teratur. Sementara itu pendekatan struktur merupakan kumpulan dari texel-texel (tekstur pixel) dari pola yang teratur atau pola yang berulang. Oleh karena itu pendekatan ini akan sangat ideal untuk analisa tekstur buatan (*artificial texture*). Untuk men-generate suatu tekstur dapat digunakan *Co-occurrenceMatrices* yang menggunakan hubungan spasial dari kesamaan gray. Fitur numeric dihitung dari *co-occurrencematrix* yang digunakan dan mewakili perbandingan dan klasifikasi tekstur. Berikut adalah subset dari fitur standard yang diturunkan dari normalisasi *co-occurrence matrix*.

$$\text{Angular 2nd Moment} = \sum_i \sum_j p[i, j]^2$$

$$\text{Contrast} = \sum_{n=0}^{Ng-1} n^2 \{ \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} p[i, j] \}, \text{ where } |i - j| = n$$

$$\text{Correlation} = \frac{\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (i, j) p[i, j] - \mu_x \mu_y}{\sigma_x \sigma_y}$$

$$\text{Entropy} = - \sum_i \sum_j p[i, j] \log(p[i, j])$$

Dimana $p[i, j]$ adalah ke $[i, j]$ masukan dalam a gray-tone spatial dependence matrix, dan Ng jumlah beda gray-levels dalam kuantisasi citra. Salah satu kekurangan *co-occurrence matrix* adalah fitur yang telah diesktrak dan tidak harus berhubungan dengan presepsi visual citra.

1.1.3 Fitur Bentuk

Bentuk tidak harus kepada bentuk citra secara utuh tetapi bentuk berdasarkan citra secara utuh tetapi bentuk berdasarkan citra daerah tertentu yang sedang dicari. Ekstraksi bentuk dari suatu objek memiliki peranan yang penting dalam bidang visualisasi informasi. Oleh karena itu fitur bentuk sangat berguna dalam CBIR. Penggunaan bentuk dari suatu objek sampai sekarang masih menjadi masalah untuk CBIR yang efisien. Penggunaan bentuk (*shape*) suatu objek merupakan tantangan yang sulit dalam bidang CBIR. Bentuk yang suatu objek memegang peranan penting dalam pencarian atau searching citra yang mirip. Penggunaan fitur bentuk dalam bidang image CBIR kurang berkembang dibandingkan fitur warna dan tekstur hal ini karena fitur bentuk memiliki kompleksitas yang tinggi dibanding fitur warna dan tekstur. Khususnya region-region citra yang diwakili oleh objek yang ditemukan untuk menjelaskan bentuk dari objek tersebut.

Query-by-Example

Seperti diketahui bahwa suatu citra mempunyai beberapa karakteristik atau atribut yang dapat digunakan untuk keperluan *retrieval*, atribut tersebut antara lain :

- Pada Warna, tekstur dan bentuk
- Susunan atau adanya tipe objek tertentu (misalnya meja dikeliling kursi)
- Hadirnya atau adanya kejadian-kejadian tertentu seperti pertandingan sepak bola
- Kehadiran seseorang, lokasi atau peristiwa-peristiwa tertentu seperti Presiden sedang berpidato.
- Meta data seperti siapa yang mengambil foto dimana dan kapan.

Query dapat dibagi menjadi beberapa level, dimana semakin tinggi level semakin meningkat kompleksitasnya.

- **Level 1** merupakan tipe retrieval dengan menggunakan fitur *primitive* seperti warna, *texture*, *shape* atau *spatial location* dari elemen citra. Contoh query ini adalah “cari foto atau objek yang tipis dan gelap yang ada pada kira atas. “Cari gambar yang mirip dengan gambar ini”
- **Level 2** merupakan query dengan menggunakan fitur *derived* (biasanya dikenal sebagai fitur *logical*). Dalam level ini melibatkan beberapa *logical inference* tentang identitas *about the identity of the objects* yang diwakili dalam citra. Level ini dapat dibedakan menjadi dua kategori:
 - a. *Retrieval* objek menurut jenis misalnya, cari gambar bus way.
 - b. *Retrieval* objek atau orang misalnya, cari gambar tugu tani.
- **Level 3** merupakan retrieval dengan menggunakan atribut *abstract*, atribut abstrak mencakup *significant amount of high-level reasoning* tentang arti dan tujuan objek yang ada. Level ini dapat dibagi menjadi :
 - a. *Retrieval* kejadian atau aktifitas contoh “Cari orang sedang berlari”
 - b. *Retrieval* gambar wajah dengan ekspresi emosi “Cari gambar wajah tegang”

1.2 Database Citra Digital

Terdapat banyak contoh database citra digital Menurut Gonsale *et.al.*, (2007) citra (*image*) adalah gambar pada bidang dwimatra atau dua dimensi. Citra juga dapat diartikan sebagai kumpulan titik-titik dengan intensitas warna tertentu yang membentuk suatu kesatuan dan mempunyai pengertian artistik. Citra sebagai salah satu komponen multimedia yang memegang peranan sangat penting sebagai salah satu bentuk informasi visual. Lebih lanjut Gonsale *et.al.*, (2007) menyatakan citra sebagai keluaran dari suatu sistem perekam

data dapat bersifat analog, berupasinyal-sinyal video seperti gambar pada monitor televisi atau bersifat digital yang dapat langsung disimpan pada suatu media magnetik. Citra ada dua macam yaitu citrakontinu dan citra diskrit. Citra Kontinu dihasilkan dari sistem optik yang menerimasinyal analog, contohnya mata manusia, kamera analog.

2
Suatu Citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x dan y, dan nilai Amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut citra digital. Nilai Pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut sebagai *picture element* atau *image elements* atau *pels* atau *pixels*, Russ (2010).

1.3 Matching Image Query

1.3.1 Matching Berdasarkan Fitur Warna

Untuk membanding dua image atau citra dalam content based image retrieval pada umumnya menggunakan contoh sebuah gambar (*image*) dan citra dalam database dengan cara menghitung *distance* (jarak) antara dua citra. Pengukuran jarak (*distance*) dua citra artinya mencari kesamaan atau *similarity* antara dua citra dengan mempertimbangkan bermacam-macam fitur seperti fitur warna, tekstur, bentuk dan lain-lain. Jika jarak antara dua citra hasilnya adalah nol (0) berarti kedua citra tersebut persis sama dengan memperlihatkan fitur-fitur tersebut.

Perhitungan jarak untuk dalam CBIR berdasarkan kesamaan warna dikerjakan dengan menghitung histogram warna untuk setiap pixel yang diidentifikasi pada nilai-nilai tertentu. Penelitian saat ini mengenai proposi warna dengan menggunakan spatial relationship antara region warna.

Informasi warna sangat penting buat citra yang biasa digunakan dalam content-based image indexing. Definisi fitur warna sangat jelas secara luas digunakan dalam CBIR. Dimana kita harus melakukan de-compression dari citra yang telah dipadatkan (JPEG) kemudian

diekstrak informasi warnanya. Tetapi dalam hal CBIR compressed domain, ekstraksi fitur warna dilakukan dari koefisien DCT (koefisien DCT adalah koefisien yang didapat melalui partial decoding citra terpadatkan).

1.3.2 Matching Berbasis Fitur Tekstur

Pengukuran fitur tekstur digunakan untuk menggambarkan visual pattern didefinisikan, tekstur dipresentasikan dengan pexels yang disimpan dalam suatu himpunan tergantung pada berapa banyak tekstur terdeteksi pada suatu citra. Himpunan atau grup ini tidak hanya mendefinisikan tekstur tetapi juga mendefinisikan dimana tekstur terletak dalam citra atau image. Tekstur merupakan konsep yang sulit dalam kompleks, untuk mengidentifikasi.

Teksture merupakan konsep yang sulit untuk direpresentasikan. Mengidentifikasi tekstur dalam suatu citra terutama dapat dicapai dengan memodelkan tekstur sebagai variasi 2D greylevel. Kecerahan atau brightness dari sepasang piksel dan dihitung derajat kontras, regularity, kekasaran (coarseness) dan direction yang kemudian dapat diestimasi (Tamura, Mori & Yamawaki, 1978). Namun masalah yang ada adalah dalam mengidentifikasi pattern dari variasi pixel dan pixel yang berkaitan dengan pixel-pixel tersebut dengan class khusus tekstur seperti silky atau rough.

1.3.3 Matching Berbasis Fitur Bentuk

Yang dimaksud dengan fitur bentuk bukan bentuk dari citra tetapi merupakan bentuk dari suatu bregion tertentu yang sedang dicari. Fitur bentuk menentukan proses apa yang akan diterapkan segmentation atau edge detection ke dalam citra. Metode lain seperti (Tushabe dan Wikinson, 2008) metode filter bentuk untuk mengidentifikasi bentuk yang akurat, memerlukan campur tangan manusia karena metode-metode seperti segmentasi sangat sulit dikerjakan secara otomatis.

Fitur warna dikerjakan dengan menggunakan jarak histogram warna yang mirip, perhitungan jarak histogram warna dapat dihitung dengan menggunakan rumus :

$$D_{hist}(I,Q) = \frac{1}{2} \sum_{i=1}^n |h(I) - h(Q)| A_i$$

Dimana $h(I)$ adalah histogram database citra dengan K -bin, sedangkan $h(Q)$ adalah histogram citra query. Sementara A merupakan matrik $K \times K$. Berikut adalah langkah-langkah untuk menghitung fitur tekstur, jarak fitur tekstur dikerjakan dengan tahapan sebagai berikut :

- Retrieve citra yang ada dalam region yang mempunyai tekstur yang sama atau mirip dengan region disekitarnya.
- Kemudian *Gridded* (just like gridded color, but use texture).
- *Histogram-based* (e.g. compare the LBP histogram).

Menghitung Jarak Fitur Bentuk

Fitur bentuk jauh lebih kompleks dibandingkan dengan warna dan tekstur, bentuk memerlukan identifikasi dari region untuk membandingkan citra. Terdapat beberapa cara untuk mengukur kemiripan dari *pattern recognition* untuk mengukur *distance*, antara lain :

- *Global Shape Properties Projection Matching*.
- *In projection matching*, yaitu proyeksi horizontal dan vertikal dari suatu histogram.
- *Boundary matching* dengan menggunakan Fourier Descriptions DAN sudut dan sisi.

Sedangkan jarak antara query bentuk dan bentuk citra mempunyai dua komponen :

- Energy yang diperlukan untuk mengubah query bentuk kedalam satu dari bentuk citra yang paling cocok.
- Pengukuran bagaimana *deformed query* yang cocok dengan citra.

Beberapa langkah untuk matching citra, yaitu :

- Segmentasi citra kedalam beberapa region
- Cari properties dan hubungan diantara mereka
- Buat graph yang merepresentasikan region dan tepi yang berhubungan diantara keduanya.
- Gunakan matching untuk membandingkan dengan citra tiger sebagai suatu graph Z.

1.4 Matching Citra JPEG Berbasis Content

Dalam penelitian ini, citra yang digunakan adalah citra dengan format JPEG (citra dipadatkan). Citra JPEG, citra dibagi-bagi kedalam blok-blok (matrik 8x8), peindeksan (indexing) citra dilakukan dengan menggabungkan N blok kedalam satu gabungan blok dan setiap blok bertindak sebagai satuan ruang dalam gabungan blok. Kunci indeks (indexing key) akan mempunyai 64 elemen dan setiap elemen mempunyai satu lokasi. Pengindeksan kunci dapat dihitung dengan menggunakan formula berikut.

$$h_i = \frac{\sum_{i=1}^N (DCT_i)^2}{N}$$

Dimana h_i adalah elemen ke i dari vektor, dan *indexing key* dapat dijelaskan sebagai berikut: $H_1 = \{h_0, h_1, \dots, h_{63}\}$, sedangkan untuk menghitung Euclidean distance dapat dihitung dengan formula,

$$d(h_q, h_k) = \left(\sum_{i=1}^N \frac{|h_i^q - h_i^k|}{64} \right), d(h_q, h_k) \text{ adalah jarak antara citra referensi}$$

dan citra dalam database.

Menurut (Yanuar dan Harjoko, 2014) fitur isi citra mencakup i). bentuk, warna, tekstur, dan spasial, ii). Semantic, iii). Impresi, emosi, dan arti yang berkaitan dengan fitur-fitur yang ada. Sementara itu (Aisyah et al., 2015) menyatakan ada empat cara atau teknik dalam mengukur kemiripan citra, yaitu berdasarkan kesamaan bentuk, warna dan kesamaan hubungan objek.

Beberapa aplikasi CBIR dengan menggunakan histogram warna, layout warna, tekstur local, koefisien wavelet, dan bentuk adalah aplikasi yang diperkenalkan oleh Smeulders (Smeulders, 2000 dan Inoune 2005). Secara umum algoritma matching dapat dijelaskan sebagai berikut, misalnya terdapat dua histogram I dan M yang mana tiap histogram mempunyai storage (bin) sebanyak n, maka perpotongan dari kedua histogram dapat didefinisikan sebagai berikut:

$$H(I,M) = \frac{\sum_{j=1}^n \min(I_j, M_j)}{\sum_{j=1}^n M_j}$$

Dikarenakan kekurang dan kelebihan image retrieval berbasis teks dan konten, maka beberapa peneliti telah mencoba menggabungkan keduanya untuk pencarian citra. Terdapat banyak pendekatan telah digunakan untuk menggabungkan teks atau keyword dan teknik berbasis konten untuk pengambilan gambar. Metode sederhana yang digunakan untuk melakukan teks dan pengambilan konten berbasis secara terpisah dan menggabungkan hasil pengambilan telah diperkenalkan oleh para penulis (Besançon et al 2005; Jones et al 2005).

Wang (Wang et al 2006.) telah memperkenalkan pencarian citra berbasis anotasi, dalam sistem ini perlu keyword sebagai awal untuk mempercepat pencarian dengan memanfaatkan teknologi pencarian berbasis teks. Sayangnya keyword tidak selalu ada, maka jika keyword tidak ada kata sistem tidak akan efisiensi. Selanjutnya, sistem ini cenderung bias oleh karena tidak ada keyword yang digunakan sebagai awal dari pencarian suatu citra.

Untuk koefisien AC yang mempunyai nilai nol, maka fitur elemen dianggap sebagai nol sehingga satu vektor dari 64 elemen yang ada akan mewakili fitur tekstur blok pixel. Untuk mencirikan fitur tekstur dari seluruh citra, vektor seperti ini kemudian digunakan sebagai blok untuk digunakan sebagai indexing key. Jika misalnya terdapat sebanyak N blok koefisien DCT dalam suatu citra, maka indexing key dapat dicari dengan menggunakan persamaan berikut:

$$H = \sum_{i=1}^{64} C_i$$

Dimana $C_i = \frac{\sum_{k=1}^N (\text{category})_k}{N}$ adalah kategori ke i diantara ke 64 koefisien DCT.

Banyak pendekatan telah digunakan untuk menggabungkan teks atau *keyword* dan teknik berbasis konten untuk pengambilan gambar. Metode sederhana yang digunakan dengan teks dan pengambilan konten berbasis secara terpisah dan menggabungkan hasil citra yang terpanggil telah diperkenalkan oleh beberapa ahli (Besançon et al 2005; Jones et al 2005). Untuk mengukur kemiripan antara citra query dengan citra di dalam database dikerjakan dengan menghitung jarak Euclidean dengan menggunakan persamaan sebagai berikut.

$$d(Q, I) = \frac{\sum_i (Q_i - I_i)^2}{M}, \text{ dimana } Q \text{ dan } I \text{ merupakan citra}$$

query dan I adalah citra dalam, sedangkan M singkatan dari jumlah *content descriptor*.

1.5 CBIR Dalam Compressed Domain

Berdasarkan standar kompresi JPEG, proses dekompresi yang lengkap dapat diringkas sebagai: i. Decoding entropi melalui tabel Huffman, ii, dequantization untuk mendapatkan koefisien DCT, dan iii, IDCT untuk merekonstruksi blok piksel. Untuk setiap sistem CBIR yang bekerja dalam domain piksel, ini adalah syarat operasi atau operasi pendahuluan sebelum ekstraksi kunci dan *matching distance*.

Tugas mandiri

1. Jelaskan Pengertian dari content based image retrieval
2. Berikan contoh Algoritma terkait dengan Content Based Image Retrieval.
3. Sebutkan langkah-langkah untuk matching citra.

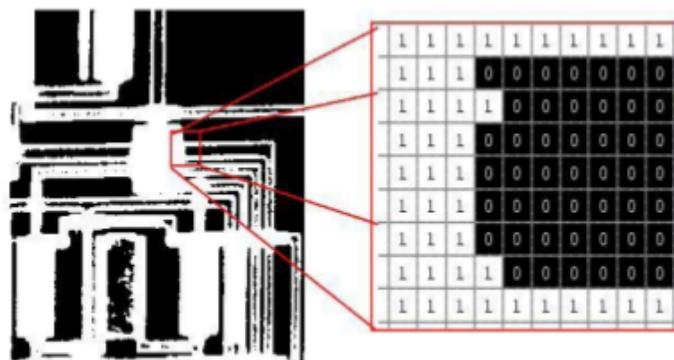
JENIS-JENIS CITRA DIGITAL

2.1 Jenis atau Tipe Citra Digital

44

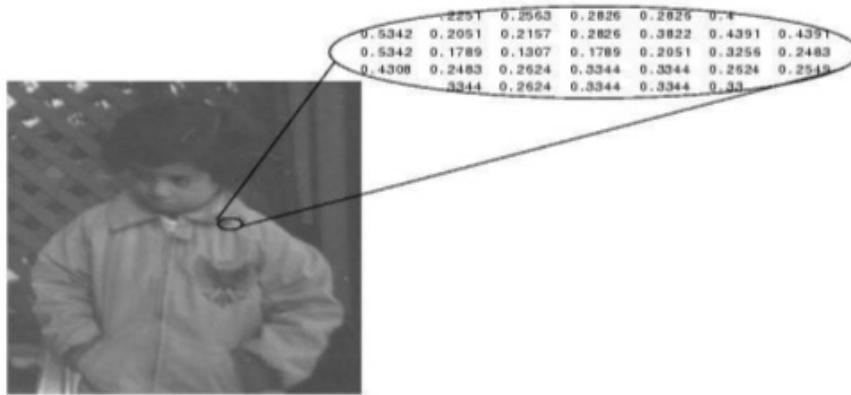
Tingkat ketajaman/resolusi warna pada citra digital tergantung pada jumlah “bit” yang digunakan oleh komputer dalam merepresentasikan setiap pixel. Dalam merepresentasikan citra setiap pixel sering digunakan adalah tipe 8-bit citra (citra 256 colors (0 untuk hitam – 255 untuk putih). Dalam matlab dikenal 4 jenis citra digital yaitu : citra binary, citra indexed, citra grayscale dan citra truecolor.

- a. Citra Binar dikenal pula sebagai **citra binary bilevel image**. Dimana matriknya berisi nilai 0 dan 1, mewakili warna putih. Contoh citra biner dapat dilihat pada gambar.



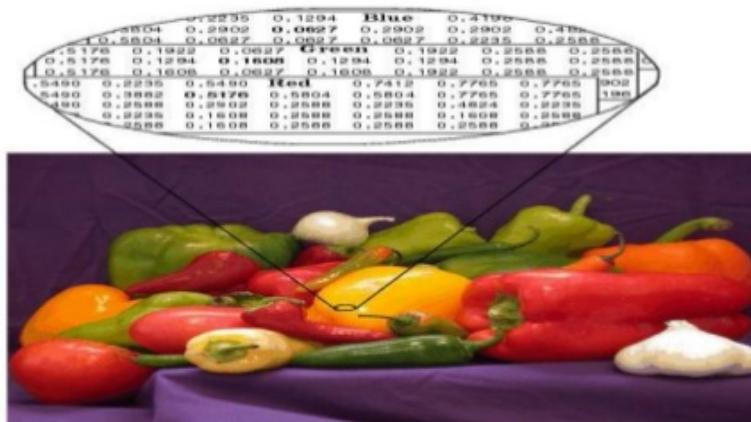
Gambar 2.1 Gambar Image Binary

- b. Citra grayscale dikenal sebagai citra: intensity, gray scale atau gray level : klas array untuk uint8, uint16, int16, single or double mempunyai nilai pixel tergantung nilai Intensitynya. Untuk array single atau double, mempunyai kisaran nilai [0, 1], untuk uint8, mempunyai kisaran nilai dari [0, 255]. Untuk uint16, kisaran nilai adalah dari [0, 65535]. Untuk int16, kisaran nilai dari [-32768, 32767].



Gambar 2.2 Citra gray scale

- c. Citra truecolor dikenal sebagai citra RGB : array m-x-n-x 3 array klas uint8, uint16, single atau double mempunyai nilai menurut nilai intensitasnya. Untuk array single atau double, mempunyai kisaran nilai dari [0, 1]. Untuk unit 8, kisaran nilai dari [0, 255]. untuk uint16, kisaran nilai dari [0, 65535].

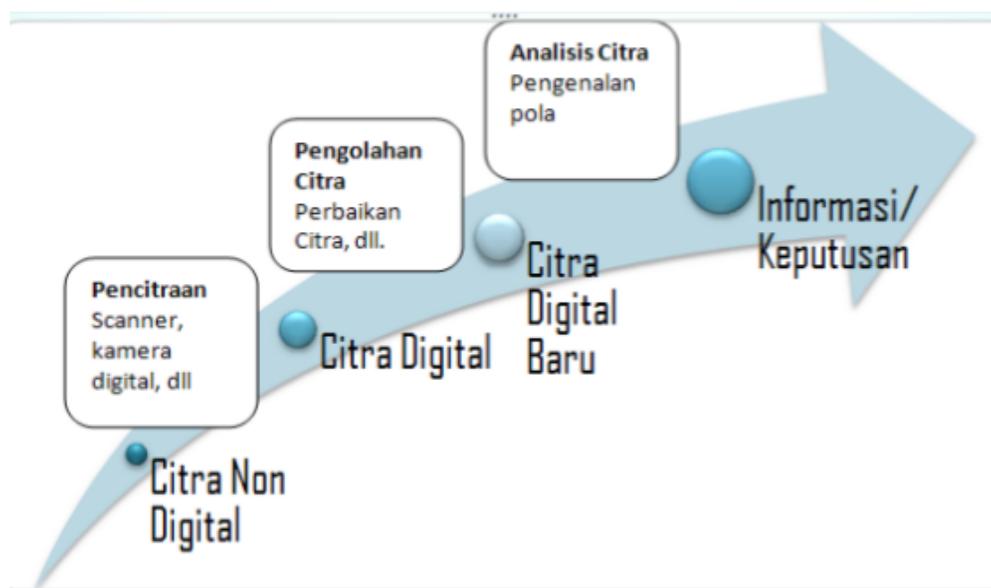


Gambar 2.3 Citra berwarna (RGB)

Untuk menentukan warna pixel pada (2,3), maka kita harus melihat triplet RGB yang disimpan pada (2,3,1:3). Untuk Red, melihat (2,3,1) -> 0.5176. Untuk Green melihat(2,3,2) -> 0.1608. Untuk Blue melihat (2,3,3) -> 0.0627. Maka warna pixel (2,3) adalah 0.5176, 0.1608 dan 0.0627.

2.2 Citra dan Pengolahan Citra

Pengolahan citra (image processing) : pemrosesan citra, khususnya menggunakan komputer, menjadi citra yang kualitasnya lebih baik. Input berupa citra dan outputnya berupa citra yang kualitasnya lebih baik dari citra input. Tahapan Pengolahan Citra Digital adapun sebagai berikut:



Gambar 2.4 Tahapan Pengolahan Citra Digital

Adapun operasi pengolahan Citra Digital, sebagai berikut :

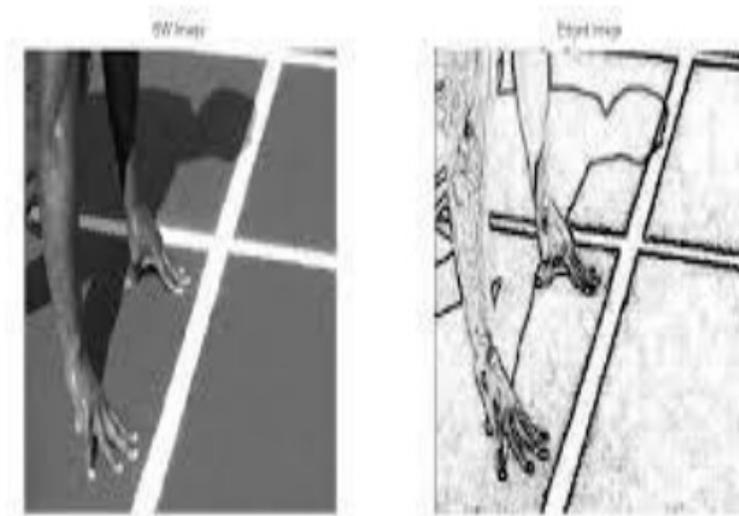
- Image Enhancement (Perbaikan Kualitas Citra) yaitu menonjolkan beberapa aspek informasi yang terkandung dalam citra. Contoh perbaikan kontras, perbaikan tepi objek, penajaman, pemberian warna semu dan lain-lain.
- Image Restoration (Pemugaran Citra) yaitu menghilangkan cacat citra. Contoh reduksi noise.

- Image Compression (Pemampatan Citra) yaitu ukuran citra yang dibuat lebih kecil dari ukuran sebenarnya. Contoh file citra BMP 258 Kb diubah menjadi JPEG 49 Kb.
- Image Segmentation (Pengelompokan Citra) yaitu memisahkan objek dari 2 unsur yang berbeda. Contoh memisahkan objek minyak dengan air, pendeteksian tanda tangan dan lainnya.



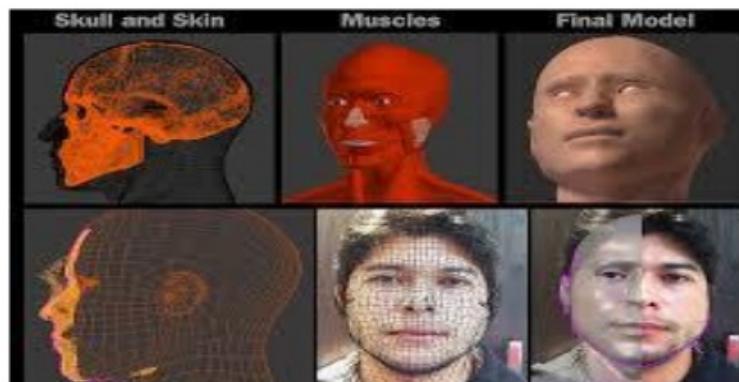
Gambar 2.5 Contoh Image Segmentation

- Image Analysis (Analisis Citra) yaitu menghasilkan deskripsi dari citra (ciri tertentu yang untuk mengidentifikasi). Contoh deteksi tepi.



Gambar 2.6 Contoh Image Analysis

- Image Reconstruction (Rekonstruksi Citra) yaitu membentuk ulang objek dari beberapa citra hasil proyeksi.



Gambar 2.7 Contoh Image Reconstruction

2.3 Konversi Antar Citra

Pada saat mengkonversi jenis citra satu ke yang lain, sering menghasilkan citra yang sedikit berbeda dengan citra asli. Terdapat beberapa fungsi atau command untuk konversi citra dalam Matlab diantaranya: `gray2ind`, `im2bw`, `ind2gray`, `rgb2ind` and `mat2gray`.

2.3.1 `rgb2gray`

Fungsi ini digunakan untuk mengubah citra berwarna (RGB) menjadi citra gray scale. `rgb2gray` digunakan untuk menghilangkan komponen Hue dan Saturation dan tetap menyimpan komponen Luminance. Contoh penggunaan command ini adalah:

`I= imread('monas.jpg');`..... membaca citra yang bernama `monas.jpg`

`J= rgb2gray(I);`..... Mengubah citra berwarna `monas` ke citra grayscale

`Figure, imshow(I), figure, imshow(J);` menampilkan citra asli dan citra hasil

2.3.2 im2bw

Mengubah citra berwarna ke citra binary image. Im2bw menghasilkan citra biner dari citra indek atau citra RGB, command ini mengubah citra ke dalam format grayscale (jika belum berupa citra intensitas) dan menggunakan *thresholding* untuk mengubah citra grayscale ke citra biner. Output citra biner BW mempunyai nilai 1 (putih) dan 0 (hitam) untuk semua piksel. Contoh penggunaan command ini dapat ditulis dibawah ini.

load pohon..... memanggil citra yang bernama pohon

BW = im2bw(X, map, 0.4)..... mengubah citra intensitas menjadi citra hitam putih

Imshow(X,map), figure, imshow(BW)....menampilkan citra asal dan citra hasil

Tugas Mandiri

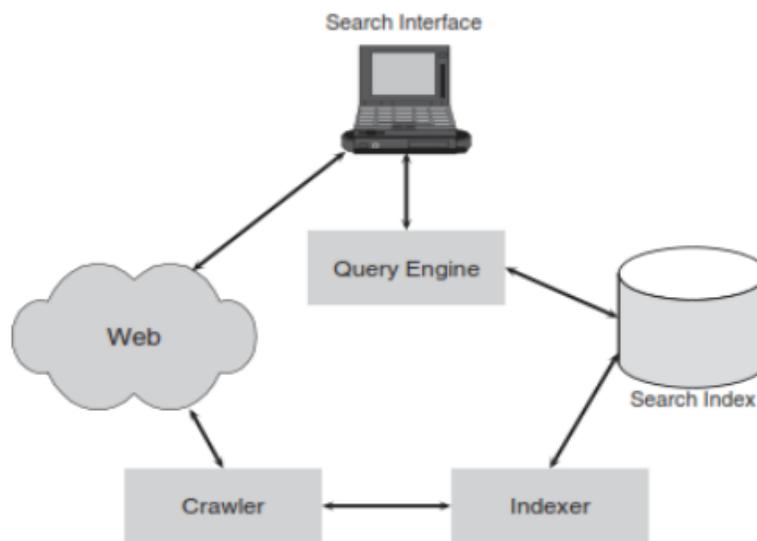
1. Jelaskan Kembali jenis-jenis dari citra.
2. Buatlah tahapan dari pengolahan citra.
3. Sebutkan jenis-jenis macam Konversi Antar Citra.
4. Sebutkan operasi pengolahan Citra Digital.

MESIN PENCARIAN

3.1 ARSITEKTUR MESIN PENCARI

Untuk Membangun mesin pencarian, komponen apa yang diperlukan dan bagaimana saling menghubungkan. Menurut Sergey Brin dan Larry 'Anatomi pencarian web hipertekstik berskala mesin besar'(1998), Yakin bahwa halaman dengan paling banyak link menuju halaman tersebut dari halaman-halaman web relevan lainnya merupakan halaman-halaman yang paling relevan, Page dan Brin memutuskan untuk mencoba tesis mereka sebagai bagian dari studi mereka - ini menjadi fondasi bagi mesin pencari mereka. Mereka secara resmi membentuk perusahaan mereka Google Inc. pada 7 September 1998.

Komponen utama mesin pencarian adalah perayap, indeks pencarian, mesin pencarian, dan antarmuka pencarian.



Gambar 3.1 Arsitektur Search Engine

Web crawler adalah perangkat lunak untuk melintasi halaman web, mendownload untuk pengindeksan dan mengikuti hyperlink yang direferensikan pada halaman yang diunduh. Web crawler juga dikenal sebagai spider, penyihir atau robot perangkat lunak. Komponen kedua adalah pengindeks yang bertanggung jawab untuk membuat indeks pencarian dari halaman web itu diterima dari penjelajah.

3.2 Indeks Pencarian

Indeks pencarian adalah repositori data yang berisi semua informasi yang dibutuhkan mesin pencari untuk mencocokkan dan mengambil halaman web. Jenis struktur data yang digunakan untuk mengatur indeks dikenal sebagai file inverted. Ini sangat mirip dengan indeks di belakang buku. Ini berisi semua kata-kata yang muncul di halaman web yang dijelajahi, terdaftar dalam urutan abjad (ini disebut indeks file), dan untuk setiap kata memiliki daftar referensi ke halaman web di mana kata itu muncul (ini disebut posting daftar). Pada tahun 1998, Brin dan Page melaporkan indeks pencarian Google mengandung 14 juta kata, jadi saat ini pasti jauh lebih besar dari itu, meskipun jelas jauh lebih kecil dari jumlah halaman web yang dilaporkan, yang saat ini lebih dari 600 miliar. (Google melaporkan bahwa setelah membuang kata-kata yang muncul kurang dari 200 kali, ada sekitar 13,6 juta kata unik dalam indeks pencarian Google) Seringkali, banyak informasi disimpan untuk setiap entri dalam indeks seperti jumlah dokumen dalam daftar posting untuk entri, yaitu jumlah web halaman yang mengandung kata kunci, dan untuk

setiap entri individu dalam file postingan juga dapat menyimpan jumlah kemunculan kata kunci di halaman web dan posisi setiap kejadian di dalam halaman. Jenis informasi ini berguna untuk menentukan relevansi konten. Indeks pencarian juga akan menyimpan informasi yang berkaitan dengan hyperlink di database tautan terpisah, yang memungkinkan mesin pencari untuk melakukan hyperlink analisis, yang digunakan sebagai bagian dari proses peringkat halaman web. Tautannya database juga dapat diatur sebagai file yang dibalik sedemikian rupa sehingga file indeksnya diisi oleh URL dan daftar posting untuk setiap entri URL, yang disebut sumber URL, berisi semua URL tujuan yang membentuk tautan antara sumber ini dan URL tujuan. Database tautan untuk Web dapat digunakan untuk merekonstruksi struktur web dan memiliki cakupan yang baik, file indeksnya harus berisimiliaran entri. Ketika memasukkan daftar posting dalam perhitungan ukuran dari database tautan, maka jumlah entri dalam database akan menjadi urutan besarnya lebih tinggi. Kompresi basis data tautan demikian penting masalah untuk mesin pencari, yang perlu melakukan analisis hyperlink yang efisien. Randall et al telah mengembangkan teknik kompresi untuk database tautan, yang memanfaatkan struktur Web. Teknik mereka berdasarkan pada pengamatan bahwa sebagian besar halaman web cenderung tertaut ke halaman lain di web yang samasitus, dan banyak halaman web di situs web yang sama cenderung tertaut ke kumpulan umum halaman. Menyisir pengamatan ini dengan metode kompresi terkenal, telah berhasil mengurangi persyaratan ruang hingga enam bit per hyperlink. Teks yang dilekatkan ke hyperlink, disebut tautan (atau anchor) teks, itu diklik oleh pengguna yang mengikuti tautan, dianggap sebagai bagian dari web halaman itu referensi. Jadi ketika sebuah kata seperti "catur" muncul di beberapa teks tautan, maka daftar posting untuk kata itu akan berisi entri untuk URL tujuan dari tautan.

3.3 Mesin Query

Mesin query adalah jantung algoritmik dari mesin pencari. Kerja dalam organisasi mesin query komersial adalah rahasia yang dijaga dengan baik, karena mesin telusur benar paranoid, takut situs web yang ingin meningkatkan peringkat mereka secara tidak berhati-hati memanfaatkan algoritme yang digunakan mesin telusur untuk memberi peringkat halaman hasil. Mesin pencari melihat manipulasi seperti itu sebagai spam, karena itu mengerikan efek pada kualitas hasil yang disajikan kepada pengguna. (Spam biasanya terkait dengan e-mail yang tidak diminta juga dikenal sebagai junk e-mail, meskipun katasпам awalnya berasal dari ham yang dibumbui dan mengacu pada produk daging kaleng.) Tidaklah mudah untuk membedakan antara spam mesin telusur dan organik optimisasi mesin pencari, di mana desain halaman web yang baik dan sehat mengarah mereka terlihat pada hasil teratas mesin telusur untuk query yang terkait halaman-halaman. Mengenai spam mesin telusur harus diselesaikan di pengadilan. Mesin query menyediakan antarmuka antara indeks pencarian, pengguna, dan Web. Mesin kueri memproses kueri pengguna dalam dua langkah. Dalam langkah pertama, mesin kueri diambil dari informasi indeks pencarian halaman web yang berpotensi relevan yang cocok dengan kata kunci dalam permintaan pengguna, dan pada langkah kedua peringkat hasil dihasilkan, dari yang paling relevan ke bawah.

Memutuskan cara menentukan peringkat web halaman berputar pada pemahaman kita tentang konsep apa yang "relevan" untuk pengguna, diberi pertanyaan. Masalah dengan relevansi adalah apa yang relevan untuk satu pengguna mungkin tidak relevan dengan yang lain. Singkatnya, relevansi adalah, untuk sebagian besar, pribadi dan tergantung pada konteks dan tugas yang ada dalam pikiran pengguna. Mesin pencari mengambil pandangan yang sangat pragmatis tentang relevansi dan terus-menerus menyesuaikan dan meningkatkan algoritma peringkat mereka dengan memeriksa bagaimana peselancar mencari Web, sebagai contoh, dengan mempelajari log permintaan terbaru.

3.4 Antarmuka Pencarian

Setelah query diproses, mesin query mengirim daftar hasil ke pencarian antarmuka, yang menampilkan hasil di layar pengguna. Antarmuka pengguna menyediakan tampilan dan nuansa dari mesin pencari, memungkinkan pengguna untuk mengajukan pertanyaan, telusuri daftar hasil, dan klik halaman web yang dipilih untuk penelusuran lebih lanjut. Dari sudut pandang kegunaan, penting bahwa pengguna dapat membedakan antara yang disponsorikan, yang merupakan iklan, dan hasil organik, yang diberi peringkat berdasarkan query mesin. Sementara sebagian besar mesin pencari web telah memutuskan untuk pindah tampilan portal web menuju tampilan yang lebih sederhana dan lebih bersih yang dipelopori oleh Google, beberapa dari mereka, terutama Yahoo dan MSN, mempertahankan tampilan portal rumah mereka halaman, menawarkan pengguna mereka berbagai layanan dengan harapan mengubah mereka menjadi pelanggan, terlepas dari penggunaan layanan pencarian inti mereka.

3.5 Google

Google berawal dari proyek penelitian dua mahasiswa Ph.D. Universitas Stanford, Larry Page dan Sergey Brin pada awal 1996 yang mengembangkan teori bahwa sebuah mesin pencari yang berdasarkan analisis matematika hubungan antara situs-situs web akan memberikan hasil yang lebih baik daripada dengan menggunakan teknik-teknik pencarian dasar yang digunakan pada saat itu. Sistem ini pada awalnya dinamakan BackRub karena menggunakan backlink untuk memperkirakan seberapa penting sebuah situs. Google menjadi populer di antara pengguna Internet karena desainnya yang sederhana dan 'bersih' serta hasil pencariannya yang relevan. Iklan dijual berdasarkan kata kunci (keyword) sehingga mereka menjadi lebih relevan bagi para pengguna, dan iklan-iklan tersebut diharuskan menggunakan teks saja agar desain halaman tetap rapi dan loading halaman tetap cepat. Konsep penjualan iklan berdasarkan kata kunci diawali oleh Overture yang dulunya bernama GoTo.com.

Pada saat kebanyakan perusahaan dotcom lainnya bangkrut, Google secara diam-diam semakin memperkuat pengaruhnya dan mendapatkan. Pada September 2001, mekanisme pemeringkatan Google (PageRank) diberikan hak paten Amerika. Hak paten tersebut diberikan secara resmi kepada Leland Stanford University dan mencantumkan nama Lawrence Page sebagai sang pencipta. Pada Februari 2003, Google membeli Pyra Labs, pemilik Blogger, sebuah situs web pionir dan pemimpin hosting weblog. Akuisisi ini tampak tidak konsisten dengan misi umum Google, namun langkah ini membuat Google dapat menggunakan informasi dari posting-posting blog untuk memperbaiki kecepatan dan relevansi artikel-artikel di Google News. Pada masa puncak kejayaannya pada awal 2004, Google mengurus hampir 80 persen dari seluruh permintaan pencarian di Internet melalui situs webnya dan klien-klien seperti Yahoo!, AOL dan CNN. Share Google turun sejak Yahoo! melepaskan teknologi pencarian Google pada Februari 2004 agar dapat menggunakan hasil pencarian independen.

Google tak hanya unik dari asal katanya. Google pun memiliki latar belakang sejarah yang unik. Google lahir dari sebuah pertemuan dua pemuda yang terjadi secara tidak sengaja pada tahun 1995 lalu. Larry Page, alumnus Universitas Michigan, yang sedang menikmati kunjungan akhir pekan, tanpa sengaja dipertemukan dengan Sergey Brin, salah seorang murid yang mendapat tugas mengantarkan Larry. Dalam pertemuan tanpa sengaja tadi, dua pendiri Google tersebut sering terlibat diskusi panjang. Keduanya memiliki pendapat dan pandangan yang berbeda sehingga sering terlibat perdebatan. Namun, perbedaan pemikiran mereka justru menghasilkan sebuah pendekatan unik dalam menyelesaikan salah satu tantangan terbesar pada dunia komputer. Yakni, masalah bagaimana memperoleh kembali data dari set data masif. Pada Januari 1996, Larry dan Sergey mulai melakukan kolaborasi dalam pembuatan search engine yang diberi nama BackRub. Setahun kemudian pendekatan unik mereka tentang analisis jaringan mengangkat reputasi BackRub. Kabar mengenai teknik baru mesin pencari langsung menyebar ke penjuru kampus. Larry dan Sergey terus menyempurnakan teknologi Google sepanjang awal 1998. Keduanya juga mulai mencari investor untuk

mengembangkan kecanggihan teknologi Google. Misi Google adalah, “untuk mengumpulkan informasi dunia dan menjadikannya dapat diakses secara universal dan berguna.” Filosofi Google meliputi slogan seperti “Don’t be evil”, dan “Kerja harusnya menantang dan tantangan itu harusnya menyenangkan”.

Penjelajah web Google menjelajahi Web yang mencakup lebih dari satu triliun halaman web di upaya spidering terus menerus. Secara berkala, Google memperbarui indeks pencariannya, yang pada akhir 2009 didistribusikan di lebih dari 500.000 server di beberapa cluster dan pada pertengahan tahun 2008, secara geografis tersebar di 36 pusat data Google menyimpan lusinan salinan indeks pencariannya di seluruh klaster dan selama menari Google, yang merupakan periode waktu antara awal dan akhir dari pembaruan indeks, beberapa server pasti akan memiliki versi lama indeks, dan dengan demikian hasil pencarian akan bervariasi tergantung pada server dari mana hasilnya diambil. Biasanya, tarian Google membutuhkan waktu beberapa hari, dan periode ini dianggap sebagai waktu terbaik untuk situs web untuk memperbarui konten web mereka halaman. Begitu tarian selesai, Google memulai perayapan baru dalam ayunan penuh, dan jika perayapan baru tidak memanen laman web yang diperbarui, webmaster akan harus menunggu sebulan lagi sebelum Google memiliki salinan baru mereka situs. Periode tari Google juga merupakan waktu ketika Google memperkenalkan tweak ke mesin kueri, sehingga para webmaster ingin mengetahui peringkat terbaru situs mereka untuk mengoptimalkannya sebelum berikutnya.

3.6 Pencarian Gambar

Pencarian gambar web itu penting, karena ada sejumlah besar informasi visual di halaman web yang mungkin ingin dicari oleh pengguna. Dalam beberapa kasus, gambar dapat bertindak sebagai pembeda untuk pencarian berbasis teks, misalnya jika Anda mencari perusahaan dan yang dapat Anda ingat adalah bahwa logo mereka memiliki gambar bidak catur. Dalam kasus lain hasil pencarian bisa berupa gambar, misalnya, jika Anda ingin melihat gambar Senat House di London. Beberapa gambar dalam buku ini diperoleh dari web publik melalui proses pencarian yang bijaksana, tetapi tidak ada

pencarian yang melibatkan pencarian berbasis konten, menggunakan fitur yang terdapat dalam gambar.

Beberapa aplikasi pencarian gambar adalah penyaringan materi yang tidak pantas dan ilegal (materi semacam itu tidak selalu mudah dideteksi oleh teks di halaman web), perjalanan dan pariwisata (gambar dan peta tempat yang akan kita kunjungi), pendidikan dan pelatihan (gambar untuk mengilustrasikan ide), hiburan (untuk bersenang-senang), e-commerce (kami ingin melihat apa yang kami rencanakan untuk beli), desain (seperti membangun rencana), sejarah dan seni (kami ingin melihat artefak atau lukisan), mode (kami ingin melihat apa yang trendi), dan pengambilan gambar domain-spesifik (seperti merek dagang, sidik jari, dan perangko).

Dibandingkan dengan pencarian web berbasis teks, pencarian gambar berbasis konten masih banyak di laboratorium penelitian. Selain pencarian gambar, ada masalah yang berkaitan dengan pencarian konten multimedia umum, apa pun itu, dan masalah spesifik yang perlu ditangani untuk masing-masing jenis media tertentu.

Mencari dan mengambil model 3D adalah masalah yang menantang. Ini melibatkan penggambaran bentuk-bentuk objek yang memungkinkan kesamaan antara objek yang akan dideteksi. Permintaan model 3D dapat melibatkan teks, tetapi juga sketsa model yang akan dicocokkan, misalnya kerangka mobil dapat ditarik dan sistem akan mengembalikan objek mirip mobil dari repositori. Demonstrasi mesin pencari 3D, yang sedang dikembangkan oleh Princeton Shape Retrieval and Analysis Group, dapat ditemukan di <http://shape.cs.princeton.edu/search.html>.

Pencarian konten audio adalah area yang berkembang secara paralel dengan pengenalan suara. Namun, pengambilan informasi audio jauh lebih luas dalam lingkupnya daripada pengenalan suara. Aplikasi penting dari retrieval audio adalah mampu mengenali, membandingkan, dan mengklasifikasikan objek musik di Web.

Satu aplikasi konkret pencarian audio, dikomersilkan oleh Shazam Entertainment (www.shazam.com), adalah pengakuan lagu melalui telepon seluler [668]. Pengguna memanggil ke layanan, dan kemudian

mengarahkan telepon ke sumber musik untuk jangka waktu sekitar 10 - 15 detik. Dalam beberapa detik tambahan, layanan ini akan mengidentifikasi lintasan dari basis data yang berisi lebih dari 8 juta sidik jari jejak (sejak 2009), menggunakan perangkat lunak pengenalan musik yang dikembangkan oleh Avery Wang, sementara ia adalah mahasiswa PhD di bidang Teknik Elektro di Stanford University.

sekarang berkonsentrasi pada pencarian gambar. Meskipun mesin pencari web berbasis teks dirancang untuk menemukan halaman web yang relevan, mereka tidak memiliki kemampuan built-in untuk menemukan gambar dalam halaman-halaman ini. Untuk mengatasi situasi ini, mesin pencari web utama sekarang menyediakan pencarian gambar sebagai layanan terpisah.

Dalam konteks Web, pencarian gambar, adalah masalah menemukan gambar yang relevan yang terdapat di halaman web. Untuk menjawab pertanyaan, mesin pencari gambar dapat menggunakan isyarat tekstual dalam halaman yang berisi gambar, dan konten yang disematkan dalam gambar itu sendiri. Tantangan besar pencarian gambar adalah untuk dapat mengambil gambar dengan konten secara andal. Di bawah ini kita akan membahas bagaimana hal ini dapat dilakukan, tetapi pertama-tama kita akan melihat bagaimana pencarian gambar berbasis teks dilakukan oleh mesin pencari utama.

3.7 Pencarian Gambar Berdasarkan Text

⁴⁵ Google (<http://images.google.com>), Yahoo (<http://images.yahoo.com>), dan Microsoft Bing (<http://images.bing.com>) menyediakan pencarian gambar sebagai bagian dari layanan mereka, sementara Picsearch (www.picsearch.com) adalah mesin pencari gambar independen, dikelola dari Stockholm, yang dikhususkan untuk pengambilan video, gambar, dan audio. Pada awal 2010, Picsearch menyediakan akses ke lebih dari 3 miliar gambar, dan mesin pencari utama web dengan mekanisme perayapannya yang kuat akan mengindeks angka yang jauh lebih besar.

Di semua mesin pencari gambar ini, query ditentukan secara tekstual; tidak ada input visual, seperti pengguna yang menyediakan gambar contoh, adalah mungkin. Pada saat ini (awal 2010), selain menggunakan informasi tekstual yang tersedia seperti yang dijelaskan di bawah ini,

mesin pencari gambar sudah menggunakan beberapa fitur tingkat rendah dari konten gambar dalam proses pencarian.

Bagaimana gambar dapat digolongkan menggunakan informasi tekstual? Ketika crawler mesin pencari mengunduh halaman web HTML, juga mengunduh gambar yang dikandungnya, yang ditentukan dalam tag IMG. Bersama dengan gambar, penjelajah mengekstrak namafile gambar, teks yang menggambarkan gambar dari bidang ALT, judul gambar, URL halaman web yang berisi gambar, dan teks apa pun yang mengelilingi tag IMG. Kata kunci yang diperoleh dari potongan teks ini kemudian disimpan dalam indeks gambar, dan digunakan untuk mencari gambar menggunakan metode pengambilan teks TF - IDF standar.

3.8 Pencarian Gambar Berbasis konten

Terlepas dari permintaan teks standar, pengguna dapat menetapkan gambar sebagai masukan ke mesin pencari. Seringkali sulit untuk menemukan gambar, yang mirip dengan yang kita cari, sehingga permintaan awal mungkin tekstual. Alternatif visual yang menarik, selain menampilkan gambar, yang disediakan oleh beberapa mesin pencari gambar adalah memungkinkan pengguna untuk membuat sketsa fitur utama dari gambar yang mereka cari, atau untuk memberikan ikon perwakilan. Setelah hasil awal dikembalikan, pengguna mungkin ingin menegaskan kembali permintaan mereka melalui umpan balik relevansi, seperti yang dijelaskan di bawah ini.

Ketika gambar ditambahkan ke indeks mesin pencari, gambar disegmentasikan ke wilayah yang lebih kecil yang homogen sesuai dengan beberapa kriteria. Fitur tingkat rendah, terutama, warna, bentuk, dan tekstur, diekstraksi dari gambar dan disimpan dalam indeks. Karena jumlah dimensi dari fitur vector mungkin tinggi, teknik reduksi dimensi, seperti pengelompokan, akan sering digunakan sebelum fitur disimpan. Dalam prakteknya informasi

disimpantentang gambar hanya sebagian deskripsi dari objek yang diwakili di dalamnya. Selain itu, fitur tingkat rendah tidak dapat menggambarkan tingkat tinggi seperti objek yang menjadi mobil, seseorang, atau pantai liburan. Lebih buruk lagi adalah kenyataan bahwa manusia cenderung menafsirkan gambar dalam hal fitur semantik tingkat tinggi. Masalah yang paling menantang ini disebut jeda semantik. Salah satu cara untuk mengatasi kesenjangan semantik adalah dengan menambahkan anotasi teks ke gambar, tetapi dalam konteks pencarian web, ini bukanlah solusi terukur. telah dibahas bagaimana mesin pencari saat ini menggunakan isyarat tekstual dalam pencarian gambar web; isyarat tekstual ini memberikan bentuk anotasi tingkat tinggi. Pengorganisasian gambar berdasarkan kategori dalam struktur direktori, dengan cara yang mirip dengan direktori web, adalah metode lain untuk membantu pengguna mengidentifikasi gambar melalui konsep semantik; tetapi seperti halnya dengan direktori halaman web, ini juga bukan proposisi terukur. Namun, direktori gambar yang dibuat dengan baik dapat memberikan banyak contoh gambar yang dapat digunakan pengguna untuk memulai pencarian gambar berbasis konten.

Untuk mengilustrasikan proses pengambilan, asumsikan bahwa kueri dispesifikasikan sebagai gambar. Citra masukan ini disegmentasikan dan fitur-fiturnya diekstraksi. Kemudian pencarian indeks dilakukan untuk menemukan gambar yang paling mirip dengan input, khususnya, ke tetangga terdekatnya ditemukan dan diberi peringkat berdasarkan kesamaan. Ini disajikan kepada pengguna, yang kemudian dapat mempertegas kueri melalui umpan balik relevansi.

Dalam upaya untuk menemukan ukuran kemiripan yang sesuai, para peneliti telah meminjam ide-ide dari teori psikologi penilaian kesamaan. Satu anggapan umum adalah bahwa kesamaan adalah fungsi yang positif dan monotonik yang tidak seimbang yang memenuhi aksioma jarak tertentu, tetapi pendekatan probabilistik yang berbeda terhadap kesamaan juga dimungkinkan.

Gagasan di balik umpan balik relevansi adalah bahwa pengguna, kata Archie, dapat menegaskan kembali permintaan awalnya sebagai berikut, setelah sistem pengambilan gambar mengembalikan kepadanya daftar peringkat gambar hasil, sebagai jawaban atas

permintaan awalnya. Archie kemudian memeriksa gambar yang dikembalikan dan menandai beberapa dari mereka sebagai "relevan" (contoh positif dari apa yang ingin dilihatnya, yaitu, lebih seperti ini) dan yang lain sebagai "tidak relevan" (contoh negatif dari apa yang dia tidak ingin lihat, yaitu, kurang seperti ini). Sistem merespons umpan balik ini dengan menyesuaikan query ke arah umpan baliknya. Penyesuaian melibatkan memodifikasi bobot fitur dalam query asli dan memperluas query dengan fitur baru dari gambar yang ditandai. Bobot fitur dari contoh positif meningkat dan bobot fitur dari contoh negatif menurun. Permintaan yang disesuaikan kemudian diterbitkan kembali ke mesin pencari gambar, dan hasil yang diharapkan adalah bahwa set baru gambar hasil telah "bergerak" ke arah apa yang ingin dilihat Archie, yaitu, lebih banyak gambar yang relevan dan kurang relevan. Proses umpan balik relevansi dapat diulang beberapa kali hingga Archie puas dengan hasilnya atau sampai set hasil telah stabil. Umpan balik relevansi sangat penting dalam konteks pencarian gambar, karena sering kali cara terbaik untuk merumuskan query adalah dengan memberikan contoh gambar sistem dari apa yang ingin Anda lihat, sebagai tambahan atau tanpa permintaan teks.

Umpan balik relevansi juga bisa kolaboratif, dalam arti bahwa umpan balik dapat disimpan dalam log file dan kemudian diagregasikan di banyak pengguna. Idennya di sini adalah untuk menyesuaikan bobot fitur semua pengguna dengan query yang serupa, sehingga berbagi umpan balik di antara pengguna yang berbeda. Eksperimen dengan iFind, gambar berbasis konten

mesin pencari yang dikembangkan di Microsoft Research China, telah menunjukkan bahwa bentuk umpan balik relevansi kolektif ini efektif dalam meningkatkan ketepatan mesin pencari gambar.

Variasi, yang disebut umpan balik pseudorelevansi, dapat digunakan untuk meningkatkan hasil query hanya berdasarkan pada isyarat teks dan tanpa interaksi pengguna. Begini Cara kerjanya. Archie mengajukan permintaan ke mesin pencari gambar seperti sebelumnya, dan set awal gambar hasil dikembalikan. Sekarang, alih-alih mengembalikan ini ke Archie, sistem reranks hasil awal menggunakan mesin pencari berbasis teks, sebagai berikut.

Pertama Peneliti mempertimbangkan halaman web yang berisi hasil awal dan membangun vektor untuk setiap halaman yang menyimpan nilai TF - IDF dari setiap kata di halaman, setelah menghilangkan kata-kata penghenti dan mungkin membendung kata-kata yang tersisa; kami memanggil vektor ini, vektor gambar untuk query. Permintaan asli kemudian dikirim ke mesin pencari berbasis teks, yang mengembalikan satu set alternatif halaman hasil. Pada tahap ini, kita membangun satu vektor teks, menyimpan nilai TF - IDF dari semua kata dalam hasil, dengan mempertimbangkan semua hasil berbasis teks bersama seolah-olah mereka ada dalam satu halaman. (Kami mencatat bahwa dalam skema ini dimungkinkan untuk memberikan bobot yang lebih tinggi pada kata-kata yang muncul di halaman peringkat yang lebih tinggi.)

Reranking hasil awal sekarang dilakukan dengan menghitung kemiripan antara masing-masing vektor gambar dan vektor teks, dan memeringkatnya dari kesamaan tertinggi hingga terendah. Satu ukuran kesamaan yang dapat dihitung antara dua vektor adalah produk titik dari vektor. Ini disebut kesamaan vektor dan dihitung dengan mengalikan nilai TF - IDF dalam dua vektor kata demi kata dan menjumlahkan hasilnya; jika diinginkan ini bisa dinormalisasi untuk menghasilkan ukuran kesamaan antara 0 dan 1.

Eksperimen menggunakan umpan balik pseudorelevansi untuk rerank hasil dari pencarian gambar Google telah menunjukkan peningkatan yang substansial dalam ketepatan sistem pencarian gambar.

Tugas Mandiri

1. Jelaskan apa yang di maksud dengan mesin pencari.
2. Ceritakan sejarah lahirnya google sebagai mesin pencari.
3. Jelaskan apa yang di maksud dengan mesin query.

Penerapan Pengolahan Citra

4.1 *Image Processing, Image Analysis, Image Understanding, dan Computer Vision*

Keempat istilah di atas sering dijumpai dalam mempelajari pengolahan citra digital. Belum ada keterangan yang jelas tentang batasan pengolahan citra dengan aplikasi citra lainnya seperti analisis citra, deskripsi citra, dan visi komputer. Namun keempat istilah di atas seringkali dibedakan dari input dan outputnya.

Image processing memiliki input dan outputnya berupa citra. Sebagai contoh, suatu citra ditransformasi ke bentuk citra lainnya. *Image analysis* memiliki input berupa citra dengan outputnya bukan citra akan tetapi berupa hasil pengukuran terhadap citra tersebut. Sebagai contoh, suatu citra wajah dianalisis untuk mendapatkan fitur wajah seperti jarak kedua mata dan jarak mata dengan hidung. *Image understanding* memiliki input berupa citra dengan outputnya adalah deskripsi tingkat tinggi dari citra tersebut (output bukan berupa citra). Sebagai contoh, diberikan suatu input citra seseorang keluarannya deskripsi dari orang tersebut dapat berupa seperti : orang tersebut sedang menangis, sedih, senyum, atau tertawa lebar. *Computer vision* bertujuan untuk mengkomputerisasi penglihatan manusia atau dengan kata lain membuat citra digital dari citra sebenarnya (sesuai dengan penglihatan manusia).

Hal tersebut dapat disimpulkan input dari *computervision* adalah berupa citra penglihatan manusia sedangkan outputnya berupa citra digital.

1

Kategori Pengolahan Citra

Pengolahan citra dapat dibagi ke dalam tiga kategori yakni kategori rendah, menengah, dan tinggi. Kategori rendah melibatkan operasi-operasi sederhana seperti prapengolahan citra untuk mengurangi derau, pengaturan kontras, dan pengaturan ketajaman citra. Pengolahan kategori rendah ini memiliki input dan output berupa citra. Pengolahan kategori menengah melibatkan operasi-operasi sederhana seperti segmentasi dan klasifikasi citra. Proses pengolahan citra menengah ini melibatkan input berupa citra dan output berupa atribut (fitur) citra yang dipisahkan dari citra input. Pengolahan citra kategori tinggi melibatkan proses pengenalan dan deskripsi citra.

15

4.2 Penerapan Pengolahan Citra Digital

Pengolahan citra digital dimulai sekitar awal tahun 1920-an dari dunia pemberitaan media cetak, di mana sebuah citra dikirim melalui kabel bawah laut dari London menuju ke New York. Proses transmisi ini menghemat waktu pengiriman dari seminggu menjadi kurang dari tiga jam. Sebelum dikirim, citra terlebih dulu dikodekan dan setelah diterima citra direkonstruksi ulang. Contoh ini sebenarnya masih kurang tepat digunakan sebagai awal mula pengolahan citra digital karena dalam prosesnya belum menggunakan teknologi komputer. Pengolahan citra digital sebenarnya mulai dari sekitar tahun 1960-an ketika saat itu diluncurkan komputer yang mampu melakukan pengolahan citra. Komputer tersebut adalah pemicu cepatnya perkembangan teknologi pengolahan citra digital. Pada tahun 1964 terjadi proses pengolahan citra berupa perbaikan kualitas citra bukan dari distorsi di laboratorium Jet Propulsion. Citra bulan kemudian dikirim pesawat ulang alik *Ranger 7* (Gonzales, 2002). Pengolahan citra digital memiliki spektrum aplikasi yang sangat luas. Berikut adalah penjelasan singkat dari berbagai aplikasi pengolahan citra digital.

4.2.1 Bidang Biomedis (Biomedical)

25

Pengolahan citra digital mengalami kemajuan penting dalam bidang kedokteran ketika ditemukannya tomografi terkomputerisasi (*Computerized Tomography/CT*) pada tahun 1970-an dan kini teknologi tomografi tersebut sudah maju sangat pesat. Pengolahan citra digital dapat digunakan untuk deteksi tumor atau kanker rahim, identifikasi penyakit paru-paru, identifikasi penyakit hati, identifikasi penyakit tulang, segmentasi tulang dari otot yang lainnya, klasifikasi gigi, dan analisis citra mikroskopis. Beberapa dari kemajuan pada bidang kedokteran tersebut karena kemampuan pengolahan citra digital mampu menginterpretasikan sinar x (*x ray*). Kemajuan penting lainnya adalah aplikasi volumetrik 3D *Magnetic Resonance Imaging* (MRI) yang mampu mendapatkan pencitraan organ dalam tubuh manusia secara jelas dengan menggunakan scanner MRI.

4.2.2 Bidang Penginderaan Jarak Jauh (*Remote Sensing*)

38

Informasi penting dari sumber-sumber alam seperti pertanian, perairan, kelautan, mineral, hutan, dan geologi dapat diperoleh dengan melakukan analisis citra terhadap citra satelitnya. Pencemaran air laut, kerusakan wilayah, dan pencemaran atau polusi udara dapat dilakukan dengan menganalisis citra satelitnya. Aplikasi yang lain adalah identifikasi kapal laut yang melewati perbatasan wilayah laut suatu negara.

4.2.3 Bidang Biometrika

Teknologi pengamanan suatu sistem mengalami kemajuan pesat akibat dari pesatnya perkembangan pengolahan citra pada bidang biometrika. Sebagai contoh pemanfaatan sidik jari, iris, wajah, dan biometrika yang lainnya untuk sistem identifikasi seseorang.

33

4.2.4 Bidang Fotografi

Kemajuan di bidang fotografi memberi dampak pada bidang-bidang astronomi, *photogrammetry*, dan fisika partikel. Para astronom dapat melakukan pengukuran terhadap posisi dan jarak suatu bintang dari foto udara. Para fisikawan menggunakan citra dari gelembung hidrogen untuk melakukan penelitian dan telah mengantarkan kepada penemuan berbagai partikel dasar.

4.2.5 Bidang Desain Visual

Dunia arsitektur dapat membuat desain visual suatu bangunan sebelum melakukan pembanguna yang sesungguhnya. Desain visual akan sangat mempermudah para arsitek dalam memberikan penjelasan rinci terhadap suatu rancangannya.

4.2.6 Identifikasi Objek-objek pada Suatu Citra

Pengolahan citra digital mampu mengidentifikasi jenis atau banyaknya objek-objek pada suatu citra. Contoh aplikasinya adalah menghitung jumlah sel darah merah (eritrosit) yang rusak atau mengetahui kondisi sel darah, menghitung volume dari sampel citra gelembung yang diakibatkan air laut, menghitung jumlah gelembung pada citra gelembung sabun, dan menentukan penyebaran partikel pigmen pada citra kulit. Berikut adalah contoh pemanfaatan pengolahan citra untuk menghitung atau mengetahui sel darah merah yang rusak. Dengan teknik segmentasi citra dan *connected component labelling*.

4.2.7 Bidang Volumetrik

Salah satu kemajuan penting dalam dunia komputer grafis adalah bidang volumetrik yaitu kemampuan untuk merekonstruksi suatu citra 3 dimensi dari citra 2 dimensi.

4.2.8 Meneliti Proses Dinamis

Penelitian proses dinamis dapat dibantu dengan menggunakan rangkaian citra yang berurutan sesuai dengan perubahan yang terjadi. Aplikasi ini banyak dijumpai di berbagai ilmu pengetahuan seperti dalam ilmu botani. Ilmu botani mempelajari tentang pertumbuhan tanaman dan mekanisme untuk mengontrol pertumbuhan tersebut. Contoh lainnya datang dari disiplin ilmu oseanografi. Untuk meneliti proses mikro yang terjadi di permukaan laut dan daerah sekitarnya. Terkadang akan sangat sulit untuk melakukan perhitungan karena adanya gerakan gelombang. Untuk itu diperlukan adanya citra urutan yang dapat menggambarkan setiap perubahan yang terjadi. Para ahli akan dapat melakukan pengukuran berdasarkan perbedaan citra-citra tersebut. Contoh penggunaan deret citra lainnya adalah untuk mengetahui perbedaan

tingkat NO₂, pada permukaan bumi. Citra diambil dari satelit ini akan menunjukkan perbedaan dan akan menunjukkan letak sumber serta daerah yang paling banyak mengandung gas tersebut.

4.2.9 OCR

Salah satu aplikasi yang penting dalam dunia pengolahan citra adalah pengenalan objek (*object recognition*). Aplikasi yang paling banyak dijumpai adalah OCR (*Optical Character Recognition*). Aplikasi OCR sering digunakan untuk mengidentifikasi citra huruf untuk kemudian diubah ke dalam bentuk file tulisan.

4.2.10 Bidang Pemantapan Citra dan Video

Teknologi pemanfaatan citra dan video (*image and video compression*) berkembang sangat pesat sekarang ini. Teknologi ini mampu memanfaatkan citra dan video dengan rasio yang tinggi. Dampak kemajuan teknologi ini sangat bermanfaat dan begitu terasa pada era internet ini di mana bandwidth menjadi sesuatu yang mahal.

4.2.11 Image Retrieval atau Image Querying

Image retrieval atau *image querying* adalah aplikasi pengolahan citra yang dapat membantu pengguna mengambil atau mencari dengan cepat suatu citra pada suatu *database* citra berdasarkan *query* atau permintaan pengguna. *Database* pada umumnya memiliki ukuran dalam skala besar.

4.3 Komponen dalam Sistem Pengolahan Citra Digital

Komponen utama dalam sistem pengolahan citra digital terdiri atas beberapa komponen. Komputer dalam sistem pengolahan data citra digital merupakan komputer umum yang dapat berupa PC hingga super komputer. Software yang digunakan dapat terdiri atas *software* khusus pengolahan citra digital atau *software* bahasa pemrograman untuk mengimplementasikan algoritma pengolahan citra. *Storage* merupakan media penyimpanan citra digital, yang berguna untuk media penyimpanan selama proses dilakukan, tempat penyimpanan utama yang selalu terhubung dengan komputer untuk mempercepat pemanggilan kembali, dan media penyimpanan untuk *back up*. Sensor citra berguna untuk proses akuisisi citra (mendapatkan atau

membuat citra digital). *Display* untuk menampilkan citra digital. *Hardcopy* untuk mencetak kembali citra digital ke dalam berbagai bentuk seperti CD, kertas, film, dan lain sebagainya. *Network* atau jaringan berguna untuk transmisi citra dan biasanya membutuhkan *bandwidth* yang cukup besar.

Tugas Mandiri

1. Jelaskan penerapan pengolahan citra di bidang Biomedis (*Biomedical*).
2. Jelaskan apa yang anda ketahui mengenai *Image Retrieval* atau *Image Querying*.
3. Jelaskan apa yang dimaksud dengan OCR (*Optical Character Recognition*).

DASAR PENGOLAHAN CITRA

Setelah membaca bab ini maka pembaca akan memahami pengertian tentang citra digital., *light*, *luminance*, *brightness*, *contrast*, fenomena *mach bands*, fenomena kontras simultan, fenomena ilusi optikm proses pembentukan citra digital, sampling dan kuantisasi, pixel, *voxel*, dan relasi atarpixel, resolusi citra, jenis-jenis citra berdasarkan warna (*level intensitas*), berbagai sistem ruang warna, serta mengetahui berbagai format file citra.

5.1 Pengertian Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data 2 dimensi. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y, dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau

pixels. Istilah terakhir (pixel) paling sering digunakan pada citra digital.

5.2 Cahaya (Light), Luminance, Brightness, dan Contrast

17

Cahaya merupakan radiasi elektromagnetik yang menstimulasi sistem penglihatan manusia. Cahaya ditampilkan sebagai sebuah distribusi energi spektrum $L(\lambda)$ dimana λ adalah panjang gelombang cahaya yang dapat diterima oleh sistem penglihatan manusia. Panjang gelombang cahaya yang dapat diterima mata adalah 350 hingga 780 nm. Cahaya yang diterima oleh mata kita dapat dirumuskan sebagai berikut.

$$l(\lambda) = p(\lambda)l(\lambda)$$

Dengan $p(\lambda)$ merupakan tingkat refleksi (pantulan) atau tingkat transmisi (perambatan) dari objek dan $L(\lambda)$ adalah distribusi energi. Jangkauan iluminasi yang dapat diterima oleh sistem penglihatan manusia adalah 1 hingga 10^{10} . Retina mata manusia memiliki dua tipe penerima cahaya yang dinamakan *rod* dan *cone*. Jumlah *rod* yang terdapat dalam mata mencapai 100 juta. Bentuk *rod* relatif panjang dan tipis. *Rod* menghasilkan bentuk penglihatan *scotopic* yang merupakan respon visual dari iluminasi tingkat rendah (pangkat rendah). *Rod* berperan dalam mendapatkan citra keabuan. *Cone* memiliki jumlah yang lebih sedikit dari *rod*. Jumlah *cone* mencapai 6.5 juta dengan bentuk yang lebih pendek dan tebal dari *rod*. Sensitivitas *cone* yang lebih rendah bila dibandingkan dengan *rod*. *Cone* menghasilkan bentuk penglihatan *photopic* yakni respon penglihatan dari iluminasi tinggi (pangkat tinggi). Di antara daerah *scotopic* dan *photopic* *cone* dan *rod* berfungsi beriringan dan menghasilkan penglihatan *mesopic*. Ada tiga jenis *cone* yakni merah, hijau, dan biru yang masing-masing berfungsi untuk mendapatkan atau menangkap informasi warna. Ketiga jenis *cone* tersebut memiliki pigmen fotosensitif yang berbeda. *Cone* banyak terdapat di sekitar retina (*fovea*) dengan tingkat kepadatan mencapai 120 *cone* per derajat. Kepadatan *cone* menurun dari radius 1° di luar *fovea*. Letak *fovea* berada pada titik jatuhnya cahaya hasil proyeksi dari lensa mata. Saraf mata memiliki koneksi langsung ke setiap *cone* dan *rod*. Sinyal dari *cone* dan *rod* ditransmisikan melalui saraf mata. Hasil proyeksi

pada retina bagian kanan dari kedua mata akan ditransmisikan langsung ke otak kanan dan begitu juga pada hasil proyeksi bagian kiri yang akan ditransmisikan ke otak kiri. Hal ini memungkinkan kita tetap dapat melihat walaupun salah satu mata tidak berfungsi. *Luminance* atau intensitas dari suatu distribusi spasial objek dengan distribusi cahaya $I(x, y, \lambda)$ dapat dirumuskan sebagai berikut.

$$F(x,y) = \int_0^x l(x, y, \lambda)V(\lambda)d\lambda$$

Dengan $V(\lambda)$ adalah fungsi efisiensi relatif *luminance*. $V(\lambda)$ digambarkan berupa kurva dengan karakteristik yang ditentukan oleh penglihatan *scotopic* dan *photopic*. *Luminance* dari suatu objek tidak tergantung dari *luminance* latar sekelilingnya. Lain halnya dengan *brightness* suatu sebuah objek yang merupakan tingkat *luminance* yang diterima oleh mata dan sangat tergantung dari *luminance* latar sekitarnya. Hal ini menyebabkan sebuah objek dengan latar yang berbeda dapat memiliki tingkat *luminance* yang sama namun dengan tingkat *brightness* yang berbeda. Contoh di atas merupakan perbedaan antara *luminance* dengan *brightness*. Berikut adalah contoh fenomena yang berhubungan dengan persepsi penglihatan manusia. Dua fenomena pertama berhubungan dengan tingkat kecerahan intensitas yang diterima mata manusia sedangkan fenomena terakhir berhubungan dengan ilusi optis geometris.

Fenomena *MachBands* Sistem penglihatan cenderung menangkap nilai intensitas di sekitar batas berbeda.

Fenomena Kontras Simultan, fenomena kedua adalah kontras simultan. Fenomena ini berhubungan dengan fakta bahwa tingkat kecerahan tidak tergantung dari intensitas.

Ilusi Optis Geometris, fenomena ilusi optis adalah ketika mata menerima informasi yang salah tentang sebuah objek secara geometri.

5.3 Pembentukan Citra Digital

Pembentukan citra digital (diskrit) melalui beberapa tahapan, yaitu akuisisi citra, *sampling*, dan kuantisasi.

5.3.1 Akuisisi Citra

Proses akuisisi citra adalah pemetaan suatu pandangan (*scene*) menjadi citra kontinu dengan menggunakan sensor. Ada beberapa macam sensor untuk akuisisi citra, yaitu sensor tunggal (*single sensor*), sensor garis (*sensor strip*), dan sensor larik (*sensor array*).

Sensor Tunggal

Sensor tunggal yang paling familiar adalah *photodiode*. *Photodiode* terbentuk dari silikon yang memiliki tegangan keluaran yang sebanding dengan cahaya. Untuk menciptakan citra 2 dimensi dengan menggunakan sensor ini, harus ada proses pemindahan relatif di setiap sumbu x dan y antara sensor objek.

Negatif film merupakan penyimpanan hasil dari proses akuisisi data citra yang digulung pada dinding tabung. Tabung tersebut akan berotasi. Gerakan rotasi tabung ini merupakan perpindahan dalam sumbu y. Sensor tunggal ditempatkan di sebuah penyangga yang dapat membuatnya bergerak bebas sepanjang penyangga tersebut. Gerakan sensor di penyangga tersebut merupakan perpindahan dalam sumbu x. Metode ini tidak mahal tetapi memakan waktu yang cukup lama dalam melakukan akuisisi.

Sensor Garis

30
Sensor garis melakukan pencitraan satu arah. Sensor ini berupa deretan sensor yang disatukan dalam satu baris sehingga dapat melakukan akuisisi sumbu x secara bersamaan. Untuk mengakuisisi citra keseluruhan, sensor digerakan searah sumbu y. Sensor ini sering dijumpai dalam mesin *scanner*.

Sensor Larik

Sensor yang ketiga adalah sensor larik. Sensor jenis ini banyak sekali ditemukan dalam kamera digital. Sensor ini berbentuk larik 2 dimensi. Sensor larik yang terdapat pada amera digital disebut sensor CCD dengan ukuran sensor yang rata-rata mencapai 4.000 x 4.000 elemen.

Sensor akan menangkap setiap iluminasi yang dipantulkan oleh objek dan akan diproyeksikan ke dalam bidang citra. Secara bersamaan sensor larik akan menghasilkan keluaran yang setara dengan integrasi dari cahaya yang diterima setiap sensor.

5.3.2 Sampling

Tahap berikutnya untuk pembentukan citra digital setelah citra kontinu terbentuk adalah proses sampling.

Proses sampling adalah proses digitasi pada koordinat x, y . Seperti yang disebutkan di atas, hasil dari sensor masih berupa citra kontinu yang merupakan fungsi kontinu $f(x,y)$. Fungsi tersebut merupakan sinyal kontinu pada nilai x, y dan juga amplitudonya (intensitas). Nilai x dan y yang kontinu akan diubah menjadi bentuk diskrit.

5.3.3 Kuantisasi

Proses kuantisasi adalah proses perubahan nilai amplitudo kontinu menjadi nilai baru yang berupa nilai diskrit. Nilai amplitudo yang dikuantisasi adalah nilai-nilai pada koordinat diskrit hasil proses sampling.

16

5.4 Pixel dan Voxel

Setiap pixel mewakili tidak hanya satu titik dalam sebuah citra melainkan sebuah bagian berupa kotak yang merupakan bagian terkecil (sel). Nilai dari sebuah pixel haruslah dapat menunjukkan nilai rata-rata yang sama untuk seluruh bagian dari sel tersebut.

Pada citra 3D satuan atau bagian terkecilnya bukan lagi sebuah pixel melainkan sebuah voxel. Voxel adalah singkatan dari *volume element*. Posisi dalam voxel ditentukan dengan tiga buah variabel yaitu k yang menyatakan kedalaman (*depth*), m menyatakan posisi baris, dan n

yang menyatakan posisi kolom. Penggambarannya dapat dilakukan dengan sumbu kartesian.

5.5 Relasi Ketetanggaan

Suatu pixel p pada koordinat (x,y) memiliki empat pixel tetangga (2 dalam arah horisontal dan 2 arah vertikal) dengan koordinat sebagai berikut.

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1).....$$

Keempat pixel tersebut sering disebut dengan *4-neighbors* dari p dan dinyatakan dengan $N_4(p)$.

Selain dalam arah horizontal dan vertikal, pixel tetangga dari p juga berada pada arah diagonal dengan koordinat:

Dan dinyatakan dengan $N_D(p)$.

$N_4(p)$ dan $N_D(p)$ bersama-sama membentuk *8-neighbors* dari p dan dinyatakan dengan $N_8(p)$.

Pixel tetangga suatu pixel dapat berada di luar citra bilamana koordinat (x,y) berada pada batas (pinggir) citra.

5.6 Resolusi Citra

Resolusi citra merupakan tingkat detail suatu citra. Semakin tinggi resolusi citra maka akan semakin tinggi pula tingkat detail dari citra tersebut. Satuan dalam pengukuran resolusi citra dapat berupa ukuran fisik (jumlah garis per mm/jumlah garis per inchi) ataupun dapat juga berupa ukuran citra menyeluruh (jumlah garis per tinggi citra). Resolusi sebuah citra dapat diukur dengan berbagai cara sebagai berikut.

- a. Resolusi pixel
- b. Resolusi spasial
- c. Resolusi spektral
- d. Resolusi temporal
- e. Resolusi radiometrik

Berikut ini adalah penjelasan dari masing-masing resolusi tersebut.

5.6.1 Resolusi Pixel

Resolusi pixel merupakan perhitungan jumlah pixel dalam sebuah citra digital. Sebuah citra dengan tinggi N pixel dan lebar M pixel berarti memiliki resolusi sebesar $M \times N$. Resolusi pixel akan memberikan dua buah angka integer yang secara berurutan akan mewakili jumlah pixel lebar dan jumlah pixel tinggi dari citra tersebut. Pengertian lainnya dari resolusi pixel adalah merupakan hasil perkalian jumlah pixel lebar dan tingginya dan kemudian dibagi dengan 1 juta. Jenis resolusi pixel seperti ini sering kali dijumpai dalam kamera digital. Suatu citra yang memiliki lebar 2.048 pixel dan tinggi 1.536 = 3.145.728 pixel atau 3,1 mega pixel. Perhitungan lainnya menyatakan dalam satuan pixel per inchi. Satuan ini menyatakan banyaknya pixel yang ada sepanjang 1 inchi baris dalam citra.

5.6.2 Resolusi Spasial

Resolusi spasial menunjukkan seberapa dekat jarak setiap garis pada citra. Jarak tersebut tergantung dari sistem yang menciptakan citra tersebut. Resolusi spasial menghasilkan jumlah pixel per satuan panjang. Resolusi spasial dari sebuah monitor komputer adalah 72 hingga 100 garis per inchi atau dalam resolusi pixel 72 hingga 100 ppi.

5.6.3 Resolusi Spektrum

Sebuah citra digital membedakan intensitas ke dalam beberapa spektrum. Citra multi spektrum akan memberikan spektrum atau panjang gelombang yang lebih baik yang akan digunakan untuk menampilkan warna.

5.6.4 Resolusi Temporal

Resolusi temporal berkaitan dengan video. Suatu video merupakan kumpulan frame statis yang berupa citra yang berurutan dan ditampilkan secara cepat. Resolusi temporal memberikan jumlah frame yang dapat ditampilkan setiap detik dengan satuan *frame per second* (fps).

5.6.5 Resolusi Radiometrik

Resolusi ini memberikan nilai atau tingkat kehalusan citra yang dapat ditampilkan dalam satuan bit contoh citra 8 bit dan citra 256 bit.

Semakin tinggi resolusi radiometrik ini maka semakin baik perbedaan intensitas yang ditampilkan.

5.7 Format File Citra

36

Format file citra standar yang digunakan saat ini terdiri dari beberapa jenis. Format-format ini digunakan dalam menyimpan citra dalam sebuah file. Setiap format memiliki karakteristik masing-masing. Berikut adalah penjelasan beberapa format ukuran umum digunakan saat ini.

Bitmap (.bmp)

Format .bmp adalah format penyimpanan standar tanpa kompresi yang umum dapat digunakan untuk menyimpan citra biner hingga citra warna. Format ini terdiri dari beberapa jenis yang setiap jenisnya ditentukan dengan jumlah bit yang digunakan untuk menyimpan sebuah nilai pixel.

Tagged Image Format (.tif, .tiff)

Format .tif merupakan format penyimpanan citra yang dapat digunakan untuk menyimpan citra bitmap hingga citra dengan warna palet terkompresi. Format ini dapat digunakan untuk menyimpan citra yang tidak terkompresi dan juga citra terkompresi.

9

Portable Network Graphics (.png)

Format .png adalah format penyimpanan citra terkompresi. Format ini dapat digunakan pada citra *grayscale*, citra dengan palet warna, juga citra *fullcolor*. Format .png juga mampu menyimpan informasi hingga kanal alpha dengan penyimpanan sebesar 1 hingga 16 bit per kanal.

9

JPEG (.jpg)

.jpg adalah format yang sangat umum digunakan saat ini khususnya untuk transmisi citra. Format ini digunakan untuk menyimpan citra hasil kompresi dengan metode JPEG.

7

MPEG (.mpg)

Format ini digunakan di dunia internet dan diperuntukkan sebagai format penyimpanan citra bergerak (video). Format ini mendukung video dengan kompresi ber-rugi.

9

Graphics Interchange Format (.gif)

Format ini dapat digunakan pada citra warna dengan palet 8 bit. Penggunaan umumnya pada aplikasi web. Kualitas yang rendah menyebabkan format ini tidak terlalu populer di kalangan peneliti pengolahan citra digital.

7

RGB (.rgb)

Format ini merupakan format penyimpanan citra yang dibuat oleh *silicon graphics* untuk menyimpan citra berwarna.

RAS (.ras)

Format .ras digunakan untuk menyimpan citra dengan format RGB tanpa kompresi.

7

Postscript (.ps, .eps, .epfs)

Format ini diperkenalkan sebagai format untuk menyimpan citra buku elektronik. Dalam format ini, citra direpresentasikan ke dalam deret nilai desimal atau hexadesimal yang dikodekan ke dalam ASCII.

7

Portable Image File Format

Format ini memiliki beberapa bagian di antaranya adalah *portable bitmap*, *portable graymap*, *portable pixmap*, dan *portable network map* dengan format berturut-turut adalah .pbm, .pgm, .ppm dan .pnm. format ini baik digunakan untuk menyimpan dan membaca kembali data citra.

PPM

PPM terdiri dari dua bagian umum yaitu bagian pendahuluan dan bagian data citra. Bagian pendahuluan memiliki tiga bagian kecil yang pertama adalah pengenalan PPM yang dapat berupa *p3* (untuk citra ASCII) dan *p6* (untuk citra binari). Bagian pendahuluan yang kedua adalah ukuran panjang dan lebar citra. Bagian ketiga dari

pendahuluan adalah nilai maksimum dari komponen warna. Keistimewaannya adalah dalam data citra dapat disimpan komentar dengan memberikan tanda '#' sebelum komentar.

PGM

Format ini hampir mirip dengan format PPM hanya saja format ini menyimpan informasi *grayscale* (satu nilai per pixel). Pengenal yang digunakan adalah *p2* dan *p5*.

PBM

PBM digunakan untuk menyimpan citra biner. Hampir sama dengan PPM dan PGM, Format PBM ini memiliki pendahuluan, hanya saja pendahuluannya tidak memiliki bagian ketiga (penjelasan nilai maksimum pixel). Penggenap yang digunakan adalah *p1*.

Tugas Mandiri

1. Apa yang di maksud dengan sensor garis.
2. Jelaskan apa yang dimaksud dengan resolusi citra.
3. Jelaskan apa itu resolusi pixel.
4. Jelaskan perbedaan resolusi pixel dengan resolusi spektrum.

PERBAIKAN CITRA (*IMAGE ENHANCEMENT*)

Perbaikan citra bertujuan meningkatkan kualitas tampilan citra untuk pandangan manusia atau untuk mengkonversi suatu citra agar memiliki format yang lebih baik sehingga citra tersebut menjadi lebih mudah untuk diolah dengan mesin (komputer).

28

Perbaikan terhadap suatu citra dapat dilakukan dengan operasi titik (*point operation*), operasi spasial (*spatial operation*), operasi geometri (*geometric operation*), dan operasi aritmatik (*arithmetic operation*). Penjelasan terhadap berbagai metode perbaikan citra berikut dikelompokkan berdasarkan keempat operasi tersebut.

6.1 Operasi Titik

Histogram

Histogram citra sangat berkaitan dengan berbagai teknik pengolahan citra, terutama metode-metode yang tergolong dalam operasi titik. Oleh karena itu, pada bagian awal dari operasi titik akan dijelaskan terlebih dulu tentang konsep histogram.

Histogram citra menunjuk pada histogram dari nilai intensitas pixel. Histogram menampilkan banyaknya pixel dalam suatu citra yang

dikelompokkan berdasarkan level nilai intensitas pixel yang berbeda. Pada citra *grayscale* 8 bit, terdapat 256 level nilai intensitas yang berbeda maka pada histogram akan ditampilkan secara grafik distribusi masing-masing 256 level nilai piksel tersebut.

64

Histogram citra yang ditampilkan dalam grafik 2D, dengan sumbu x menyatakan nilai intensitas piksel dan sumbu y menyatakan frekuensi (banyaknya kemunculan) suatu nilai intensitas pixel.

Proses pembuatan histogram dapat dilakukan dengan memeriksa setiap nilai pixel pada citra, kemudian hitung banyaknya nilai pixel tersebut dan disimpan di memori.

Histogram juga bisa diterapkan untuk citra berwarna, dengan cara memisahkan terlebih dulu 3 komponen warna *red*, *green*, dan *blue*, kemudian setiap komponen warna dibuat histogramnya. Histogram citra berwarna juga dapat ditampilkan dalam grafik 3D dengan salah satu sumbu menyatakan komponen ruang warna.

6.1.1 Penyesuaian Kecerahan (*Brightness Adjustment*)

Penyesuaian kecerahan (*brightness*) intensitas pixel merupakan operasi pixel yang paling sederhana. Tingkat kecerahan suatu citra dapat dilihat dari histogramnya. Semua pixel biasanya terkonsentrasi pada salah satu sisi histogram dengan rentangan *gray level* tertentu. Semakin dinaikkan tingkat *brightness* suatu citra maka konsentrasi nilai pixel pada histogram akan bergeser ke sisi kanan, demikian juga sebaliknya, semakin diturunkan maka konsentrasi nilai pixel pada histogram akan bergeser ke sisi kiri.

Penyesuaian tingkat *brightness* suatu citra dapat dinyatakan sebagai :

$$U' = U + c$$

dengan U' dan U berturut-turut menyatakan citra setelah dan sebelum penyesuaian kecerahan sedangkan c adalah suatu konstanta yang merupakan factor penyesuaian. Proses penyesuaian di atas dilakukan dengan menambahkan (mengurangkan) nilai setiap pixel dengan suatu konstanta. Apabila nilai pixel setelah penyesuaian melebihi nilai maksimum intensitas yang mungkin untuk citra *grayscale*, nilai maksimum intensitas adalah 255 maka nilai pixel

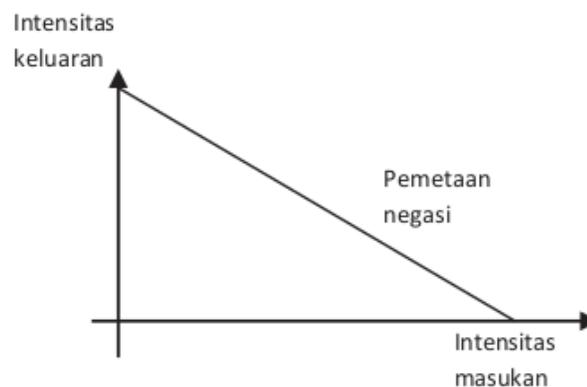
tersebut akan dijadikan 255. Demikian pula sebaliknya, bila nilai pixel hasil penyesuaian lebih kecil dari 0 (nol) maka nilai pixel tersebut dijadikan 0.

6.1.2 Negasi (Negation)

Negasi adalah proses pemetaan nilai pixel suatu citra, yaitu pada citra biner, pixel hitam dijadikan putih dan putih dijadikan hitam. Sedangkan pada citra grayscale atau berwarna, nilai maksimum pixel dikurangi dengan nilai pixel yang sedang diproses. Secara matematis, negasi dapat dinyatakan sebagai berikut.

$$U' = 2^l - U$$

Dengan U' dan U berturut-turut menyatakan citra setelah dan sebelum proses negasi dan l menyatakan nilai bit dari gray level.



6.1.3 Koreksi Gamma (Gamma Correction)

Brightness suatu citra juga dapat diperbaiki dengan menggunakan koreksi gamma. Bentuk umum dari transformasi gamma adalah :

$$U' = U^{\frac{1}{\gamma}}$$

Dengan U' dan U berturut-turut adalah citra setelah dan sebelum mengalami koreksi gamma, sedangkan γ adalah faktor koreksi gamma, dengan kisaran nilai $0 < \gamma < 1$. Bila $\gamma = 1$ maka hasil transformasi akan sama dengan citra masukan. Semakin kecil faktor koreksi maka citra output akan semakin terang dan semakin tinggi faktor koreksi maka citra output akan mendekati citra asli.

6.1.4 Perenggangan Kontras (Constrast Stretching)

3 Kontras suatu citra adalah distribusi pixel terang dan gelap, Citra *grayscale* dengan kontras rendah maka akan terlihat terlalu gelap, terlalu terang, atau terlalu abu-abu. Histogram citra dengan kontras rendah, semua pixels akan terkonsentrasi pada sisi kiri, sisi kanan atau di tengah. Semua pixel akan berkelompok secara rapat pada suatu sisi tertentu dan menggunakan sebagian kecil dari semua kemungkinan nilai pixel.

Citra dengan kontras tinggi memiliki daerah gelap dan terang yang luas. Histogram citra dengan kontras tinggi memiliki dua puncak besar. Satu puncak terkonsentrasi pada sisi kiri dan yang satunya terkonsentrasi pada sisi kanan histogram.

Citra dengan kontras yang bagus menampilkan rentangan nilai pixel yang lebar. Histogramnya relatif menunjukkan distribusi nilai pixel yang seragam, tidak memiliki puncak utama, atau tidak memiliki lembah.

Perenggangan kontras adalah teknik yang sangat berguna untuk memperbaiki kontras citra terutama citra yang memiliki kontras rendah. Teknik ini bekerja dengan baik pada citra yang memiliki distribusi Gaussian atau mendekati distribusi Gaussian.

3 Pada perenggangan kontras, setiap pixel pada citra U ditransformasi dengan menggunakan fungsi berikut.

$$3 \quad o(i, j) = \frac{u(i, j) - c}{d - c} (L - 1)$$

Dengan $o(i, j)$ dan $u(i, j)$ berturut-turut pixel sesudah dan sebelum ditransformasi pada koordinat (i, j) , c dan d berturut-turut menyatakan nilai maksimum dan minimum dari pixel pada citra input, dan L menyatakan nilai *grayscale* maksimum. Bila nilai pixel lebih kecil dari 0 maka akan dijadikan 0 dan bila lebih besar dari $(L-1)$ maka akan dijadikan $(L-1)$.

Permasalahan yang muncul dengan menggunakan formula di atas adalah bila nilai maksimum (d) pixel terlalu tinggi dan nilai minimum (c) pixel terlalu rendah, akan tetapi kehadiran pixel dengan nilai maksimum atau nilai minimum sangat sedikit (atau mungkin hanya 1)

maka proses penskalaan yaitu pembagi $(d - c)$ pada formula di atas menjadi kurang representatif. Untuk menyelesaikan permasalahan ini dapat dilakukan dengan memilih nilai c dan d berdasarkan persentase tertentu, misalnya c diberi nilai dengan nilai pixel di mana terdapat sekitar $p\%$ pixel lebih kecil dari c , demikian juga untuk nilai d diberi nilai pixel di mana terdapat sekitar $q\%$ pixel lebih besar dari d .

Sudah tentu nilai p dapat sama atau berbeda dengan q . Nilai p dan q dapat ditentukan dengan membuat histogram citra input dan berdasarkan sebaran nilai pixel pada histogram, nilai p dan q yang representatif dapat dipilih.

$$o(i, j) = \begin{cases} 0 & \text{untuk } p_{low} \leq u(i, j) \leq p_{high} \\ \frac{u(i, j) - c}{d - c} & \text{untuk } u(i, j) > p_{high} \\ 255 & \text{untuk } u(i, j) \leq p_{low} \end{cases}$$

Dengan p_{low} dan p_{high} berturut-turut menyatakan nilai pixel yang berkaitan dengan nilai $p\%$ dan $q\%$.

6.1.5 Pengirisan Intensitas (Intensity Slicing)

Pengirisan intensitas berguna ketika ingin menonjolkan intensitas (*gray level*) tertentu saja pada citra. Ada dua pendekatan yang dapat digunakan untuk melakukan pengirisan intensitas sebagai berikut. Pertama, memberi nilai tinggi pada rentangan nilai intensitas yang ingin ditonjolkan dan nilai intensitas yang lainnya diberi nilai rendah. Kedua, memberi nilai tinggi pada rentangan nilai intensitas yang ingin ditonjolkan dan nilai intensitas yang lainnya tetap dipertahankan. Pendekatan pertama dipilih bila latar belakang citra diabaikan, sedangkan pendekatan kedua untuk mempertahankan latar belakang citra. Secara matematis, kedua pendekatan di atas dapat dinyatakan sebagai berikut.

Pendekatan pertama : ... $o = \begin{cases} L, \\ 0, \end{cases}$

Pendekatan kedua :..... $o = \begin{cases} L, \\ 0, \end{cases}$

untuk $a \leq u \leq b$

untuk yang lainnya

untuk $a \leq u \leq b$

untuk yang lainnya

Dengan a dan b berturut-turut menyatakan nilai batas bawah dan atas intensitas yang mau ditonjolkan.

6.1.6 Pemisahan Bit (Bit Extraction)

Suatu pixel merupakan kombinasi dari beberapa *significant bit*, misal untuk pixel dengan nilai maksimum 255 memiliki 8 *significant bit*, yaitu dari bit ke-0 sampai ke-7. Sesuai dengan namanya, metode pemisahan bit memisahkan beberapa bit dari pixel untuk menghasilkan nilai pixel yang baru. Setiap pixel pada citra dapat dinyatakan dengan komposisi bit sebagai berikut.

$$u = k_1 2^{B-1} + k_2 2^{B-2} + \dots + k_n 2^{B-n} + \dots + k_{B-1} 2^1 + k_B \dots$$

Nilai pixel baru dapat diperoleh dengan:

$$o = \begin{cases} L, \\ 0, \end{cases}$$

untuk $k_n = 1$

untuk yang lainnya

dengan :

$$k_n = 1_n - 2i_{n-1}$$

dan

$$i_n = \text{int} \left[\frac{u}{2^{B-n}} \right], \text{int} [x] = \text{bagian integer dari } x$$

6.1.7 Pemampatan Rentangan (*Range Compression*)

Rentangan dinamis pixel suatu citra terkadang sangat besar, seperti pada citra hasil transformasi Foutier, di mana hanya sedikit pixel yang terlihat. Untuk mengatasi permasalahan ini dapat dilakukan dengan memampatkan rentangan yang besar tadi dengan metode pemampatan rentangan atau sering juga disebut Metode Transformasi Logaritmik, yang dapat dinyatakan sebagai berikut.

$$o = c \log_{10}(1 + |u|)$$

Dengan o dan u berturut-turut menyatakan nilai pixel setelah dan sebelum diolah, dan c merupakan faktor penskalaan yang dinyatakan sebagai berikut.

$$o = \frac{255}{\log(1+|R|)}$$

Dengan R merupakan nilai maksimum magnitude pada citra input. Penghitungan nilai c dengan cara di atas akan dapat menjaga nilai maksimum pixel dari citra output adalah 255.

6.1.8 Ekualisasi Histogram (*Histogram Equalization*)

Ekualisasi histogram merupakan salah satu bagian penting dari beberapa aplikasi pengolahan citra. Tujuan dari teknik ini adalah untuk menghasilkan histogram citra yang seragam. Teknik ini dapat dilakukan pada keseluruhan citra atau pada beberapa bagian citra saja.

Histogram hasil proses ekualisasi tidak akan seragam atau sama untuk seluruh intensitas. Teknik ini hanya melakukan distribusi ulang terhadap distribusi intensitas dari histogram awal. Jika histogram awal memiliki beberapa puncak dan lembah maka histogram hasil ekualisasi akan tetap memiliki puncak dan lembah. Akan tetapi puncak dan lembah tersebut akan digeser. Histogram hasil ekualisasi akan lebih disebar (spreading).

Distribusi ulang terhadap histogram awal dilakukan dengan memetakan setiap nilai pixel pada histogram awal menjadi nilai pixel baru dengan cara sebagai berikut.

$$n(g) = \max \left(0, \text{round} \left[(L - 1) * \frac{c(g)}{N} \right] - 1 \right)$$

24

Dengan $n(g)$ adalah nilai pixel baru, N menyatakan banyaknya pixel pada citra (bila citra berukuran 8×8 maka N adalah 64), g menyatakan nilai gray level awal yang nilainya dari $1 \dots L-1$ (L menyatakan nilai gray level maksimum). Sedangkan $c(g)$ menyatakan banyaknya pixel yang memiliki nilai sama dengan g atau kurang yang secara matematis dapat dinyatakan sebagai:

$$c(g) = \sum_{i=1}^g h(i), \quad g = 1, 2, \dots, L-1$$

dengan $h(i)$ menyatakan histogram awal.

Berikut contoh penyebaran histogram dengan proses ekualisasi histogram.

Tabel 6.1 Proses Ekualisasi Histogram

g	$h(g)$	$c(g)$	$n(g)$
0	6	6	0
1	28	34	0
2	34	68	1
3	94	162	4
4	40	202	5
5	48	250	7
6	2	252	7
7	4	256	7

$L = 8 \quad N = 256$

6.1.9 Ekualisasi Histogram Adaptif (*Adaptif Histogram Equalization*)

Pada dasarnya ekualisasi histogram adaptif sama dengan ekualisasi histogram di atas. Hanya saja pada ekualisasi histogram adaptif, citra dibagi menjadi blok-blok (*sub-image*) dengan ukuran $n \times n$ kemudian pada setiap blok dilakukan proses ekualisasi histogram. Ukuran blok

(n) dapat bervariasi dan setiap ukuran blok akan membenkan hasil yang berbeda. Setiap blok dapat saling tumpang tindih beberapa pixel dengan blok lainnya.

6.1.10 Penajaman Lokal

Perenggangan kontras dan ekualisasi histogram yang dibahas sebelumnya merupakan proses yang bersifat global, artinya nilai pixel baru diperoleh berdasarkan penyebaran *gray level* pada keseluruhan citra. Proses penajaman pada citra juga dapat dilakukan secara lokal, yaitu nilai pixel baru dihitung pada suatu window tertentu pada citra kemudian bergerak pixel demi pixel. Nilai pixel baru yang dihitung adalah pixel pada pusat window.

6.2 Operasi Geometri Citra

Operasi geometri adalah proses perubahan hubungan spasial antara setiap pixel pada sebuah citra. Operasi geometri memetakan kembali pixel citra input dari posisi awal (x_1, y_1) ke posisi baru (x_2, y_2) pada citra output. Proses yang tergolong ke dalam di antaranya adalah translasi, penskalaan, rotasi, refleksi, *warping* dan *Affine Transformation*.

6.2.1 Translasi

Operasi translasi adalah memindahkan setiap elemen pixel citra input ke posisi baru pada citra output di mana dimensi dari kedua citra (citra masukan dan citra output) pada umumnya adalah sama. Posisi baru dari suatu pixel ditentukan dari nilai variabel translasi (p, q) .

Secara umum operasi translasi melakukan perubahan dengan cara menambahkan koordinat awal dengan nilai variabel translasi.

$$x_2 = x_1 + p$$

$$y_2 = y_1 + p$$

Jika ukuran citra output diset sama dengan citra input maka bila terdapat posisi hasil yang berada di luar batas citra output, pixel tersebut tidak dipetakan. Untuk posisi citra output yang tidak memiliki nilai pixel diset dengan nilai 0 atau warna hitam.

6.2.1 Rotasi

Rotasi merupakan suatu transformasi geometri memindahkan nilai-nilai pixel dari posisi awal menuju ke posisi akhir yang ditentukan melalui nilai variabel rotasi sebesar θ° terhadap sudut 0° atau garis horizontal dari citra.

Proses rotasi dapat dilakukan dengan rumus sebagai berikut.

$$x_2 = \cos(\theta) \times (x_1 - x_0) - \sin(\theta) \times (y_1 - y_0) + x_0$$

$$y_2 = \sin(\theta) \times (x_1 - x_0) + \cos(\theta) \times (y_1 - y_0) + y_0$$

Di mana (x_0, y_0) adalah koordinat titik pusat dari citra input dan (θ) adalah sumbu putar. Sumbu putar pada umumnya memiliki arah putar searah jarum jam dengan garis horizontal. Seperti halnya operasi translasi, hasil perhitungan posisi hasil rotasi dapat memberikan nilai di luar batas citra output (apabila ukuran citra output sama dengan citra input). Untuk kasus seperti itu, ada beberapa implementasi yang membiarkan nilai pixel tersebut tanpa dipetakan ulang dan ada yang memetakan ke citra output sehingga menyebabkan ukuran citra membesar.

6.2.2 Refleksi

Refleksi atau pencerminan adalah proses pengolahan citra secara geometri dengan memindahkan nilai-nilai pixel pada posisi awal (x_1, y_1) menuju ke posisi baru di (x_2, y_2) pada citra output sesuai dengan posisi pencerminan. Posisi pencerminan ada tiga jenis yaitu pencerminan terhadap sumbu x, pencerminan terhadap sumbu y, dan pencerminan terhadap sumbu x dan y.

Pencerminan terhadap sumbu x di posisi x_0 dapat digambarkan dengan rumus berikut ini,

$$x_2 = -x_1 + (2 \times x_0)$$

$$y_2 = y_1$$

Sedangkan untuk pencerminan terhadap sumbu y di posisi y_0 adalah:

$$x_2 = x_1$$

$$y_2 = -y_1 + (2 \times y_0)$$

Dan untuk pencerminan yang dilakukan terhadap kedua sumbu baik x dan y di posisi (x_0, y_0) adalah:

$$x_2 = -x_1 + (2 \times x_0)$$

$$y_2 = -y_1 + (2 \times y_0)$$

41

6.2.3 Penskalaan

Penskalaan adalah sebuah operasi geometri yang memberikan efek memperbesar atau memperkecil ukuran citra input sesuai dengan variabel penskalaan citranya. Ukuran baru hasil penskalaan didapat melalui perkalian antara ukuran citra input dengan variabel penskalaan. Proses penskalaan dapat dilakukan dengan rumus:

$$P_o = S_p \times P_i$$

$$L_o = S_i \times L_i$$

Di mana (P_i, L_i) adalah ukuran citra input, (P_o, L_o) adalah ukuran citra output, dan (S_p, S_i) adalah variabel penskalaan yang diinginkan. Jika variabel penskalaan bernilai lebih besar dari 1 maka hasil penskalaannya akan memperbesar ukuran citra, sebaliknya apabila variabel penskalaannya lebih kecil dari 1 maka hasilnya akan memperkecil ukuran citra.

Metode Penskalaan untuk Memperkecil Ukuran Citra

Proses penskalaan dengan memperkecil citra dapat dilakukan dengan menggunakan metode subsampling atau interpolation.

Subsampling

Metode ini memilih salah satu nilai dari suatu daerah citra untuk mewakili daerah tersebut. Pemilihan nilai dilakukan secara acak. Metode ini sangat sederhana dan cepat namun hasilnya kurang bagus, apalagi untuk proses dengan daerah sampling yang besar.

Interpolasi

Metode kedua untuk memperkecil skala citra adalah interpolation. Metode ini menggunakan nilai rata-rata suatu region untuk mewakili region tersebut.

Metode Penskalaan dengan Memperbesar Ukuran Citra

Proses pembesaran dapat dilakukan dengan metode pengulangan pixel (pixel replication) dan interpolasi.

Pixel Replication

Pixel replication akan mengulang nilai-nilai pixel pada daerah atau kelompok yang telah ditentukan sesuai dengan faktor penskalaannya.

Interpolasi

Interpolasi pada proses perbesaran citra bekerja seperti halnya dalam proses penurunan skala citra namun kali ini digunakan untuk mengisi nilai pixel hasil perbesaran. Nilai pixel ini dihitung dari rata-rata pixel tetangganya.

6.2.4 Transformasi Affine (*Affine Transformation*)

Metode transformasi *Affine* ini menggabungkan proses dasar seperti translasi, rotasi, dan penskalaan ke dalam satu proses untuk mendapatkan citra output yang diinginkan.

Tugas Mandiri

1. Jelaskan tujuan dari perbaikan dari citra.
2. Jelaskan dari 4 perbaikan citra berikut ini operasi titik (*point operation*), operasi spasial (*spatial operation*), operasi geometri (*geometric operation*), dan operasi aritmatik (*arithmetic operation*).
3. Jelaskan tujuan dari Ekualisasi Histogram (*Histogram Equalization*)

ANALISIS CITRA

Setelah membaca bab ini, pembaca akan memahami berbagai teknik analisis citra meliputi berbagai operasi morfologi, operasi deteksi tepi dan deteksi garis, segmentasi, analisis bentuk, dan berbagai metode untuk pemisahan fitur pada citra.

7.1 Morphologi

Kata morfologi (*morphology*) secara sederhana dapat diartikan sebagai bentuk dan struktur suatu objek atau dalam deskripsi lainnya disebutkan bahwa morfologi adalah susunan dan hubungan antar bagian objek. Morfologi di dunia digital dapat diartikan sebuah cara untuk mendeskripsikan ataupun menganalisa bentuk dari objek digital.

Pada penjelasan sebelumnya, suatu citra dipandang sebagai fungsi amplitudo 2D dengan koordinat *real* atau diskrit. Pada morfologi, suatu citra dinyatakan sebagai himpunan koordinat diskrit (atau kontinu). Dalam hal ini, himpunan tersebut dengan point atau pixel objek pada citra. Karena objek dianggap sebagai suatu himpunan maka operasi-operasi himpunan seperti gabungan (*union*), irisan (*intersection*), dan komplemen (*complement*) dapat dilakukan.

Structuring Elements (SE)

Operasi morfologi menggunakan dua input himpunan yaitu suatu citra (pada umumnya citra biner) dan suatu kernel. Khusus dalam morfologi, istilah kernel biasa disebut dengan *structuring elements* (struktur pembentuk struktur). SE merupakan suatu matriks dan pada umumnya kecil.

Elemen dari SE dapat bernilai 1, 0, dan *don't care*. Nilai *don't care* biasanya ditandai dengan nilai elemen dikosongkan atau diberi tanda silang.

Ada 2 operasi dasar morfologi yaitu dilasi dan erosi. Kedua operasi dasar tersebut menjadi basis untuk membuat berbagai operasi morfologi yang sangat berguna untuk pengolahan citra digital, seperti *opening, closing, hit and miss transform, thinning* dan *thickening*.

7.1.1 Dilasi (Dilation)

Bila suatu objek (citra input) dinyatakan dengan A dan SE dinyatakan dengan B serta B_x menyatakan translasi B sedemikian sehingga pusat B terletak pada x. Operasi dilasi A dengan B dapat dinyatakan sebagai berikut.

$$D(A,B) = A \circ B = \{x : B_x \cap A \neq \emptyset\} \quad /$$

Dengan \emptyset menyatakan himpunan kosong.

Proses dilasi dilakukan dengan membandingkan setiap pixel citra input dengan nilai pusat SE dengan cara melapiskan (*superimpose*) SE dengan citra sehingga pusat SE tepat dengan posisi pixel citra yang diproses. Jika paling sedikit ada 1 pixel pada SE sama dengan nilai pixel objek (*foreground*) citra maka pixel input diset nilainya dengan nilai pixel *foreground* dan bila semua pixel yang berhubungan adalah *background* maka input pixel diberi nilai pixel *background*. Proses serupa dilanjutkan dengan menggerakkan (translasi) SE pixel pada citra input.

Semakin besar ukuran SE maka semakin besar perubahan yang terjadi. SE berukuran kecil juga dapat memberikan hasil yang sama dengan SE berukuran besar dengan cara melakukan dilasi berulang kali.

Efek dilasi terhadap citra biner adalah memperbesar batas dari objek yang ada sehingga objek terlihat semakin besar dan lubang-lubang yang terdapat di tengah objek akan tampak mengecil.

Dilasi pada Citra Grayscale

Proses dilasi dapat juga dilakukan pada citra *grayscale* dengan menggunakan rumus yang agak beda dengan rumus dilasi di atas, yaitu :

$$D_G(A,B) = \max_{[j,k] \in B} \{a[m-j, n-k] + b[j,k]\}$$

Rumus di atas menyatakan untuk input koordinat $[m,n]$, setiap elemen SE dijumlahkan dengan nilai pixel pada koordinat bersesuaian dari bagian citra yang sedang diproses dan nilai pixel hasil adalah nilai maksimum dari masing-masing penjumlahan. Elemen SE pada dilasi citra *grayscale* dapat bernilai *grayscale* pula.

Proses dilasi pada citra *grayscale* akan menyebabkan bagian terang yang dikelilingi oleh bagian yang lebih gelap akan melebar sedangkan bagian gelap yang dikelilingi oleh bagian yang lebih terang akan mengecil. Dengan kata lain proses dilasi terhadap citra *grayscale* akan menghasilkan citra yang lebih terang karena sebagian dari pixel yang memiliki intensitas keabuan yang rendah diubah menjadi tinggi.

7.1.2 Erosi

Sama seperti dilasi, proses erosi dilakukan dengan membandingkan setiap pixel citra input dengan nilai pusat SE dengan cara melapiskan SE dengan citra sehingga pusat SE tepat dengan posisi pixel citra yang diproses. Jika semua pixel pada SE tepat sama dengan semua nilai pixel objek (*foreground*) citra maka pixel input diset nilainya dengan nilai pixel *foreground*, bila tidak maka input pixel diberi nilai pixel *background*. Proses serupa dilanjutkan dengan menggerakkan SE pixel demi pixel pada citra input.

Erosi pada Citra Grayscale

Proses erosi dapat diperluas pada citra *grayscale* dengan menggunakan rumus berikut.

$$E_G(A,B) = \max_{[j,k] \in B} \{a[m+j, n+k] - b[j,k]\}$$

Rumus di atas menyatakan untuk input koordinat $[m,n]$ setiap nilai pixel citra dikurangi dengan nilai elemen SE pada koordinat bersesuaian dari bagian citra yang sedang diproses dan nilai pixel hasil adalah nilai minimum dari masing-masing pengurangan.

Proses erosi pada citra *grayscale* secara umum akan menyebabkan citra keluaran lebih gelap. Proses ini akan memperkecil wilayah terang dikelilingi oleh wilayah gelap serta akan memperbesar wilayah gelap dikelilingi wilayah terang.

Proses erosi menyebabkan citra keluaran tampak lebih gelap karena beberapa pixel citra yang memiliki intensitas keabuan yang tinggi diubah menjadi pixel dengan nilai intensitas keabuan yang rendah.

7.1.3 Opening

Operasi erosi sangat berguna dalam menghilangkan objek-objek kecil yang terdapat dalam citra, namun operasi ini memiliki kelemahan yakni terjadinya penurunan ukuran objek lainnya. Untuk mengatasi hal ini kita dapat melakukan operasi dilasi setelah melakukan operasi erosi dengan menggunakan SE yang sama. Kombinasi dari kedua proses ini dinamakan operasi opening. Secara matematis, proses opening dapat dinyatakan dengan:

$$O(A,B) = A \circ B = D(E(A,B),B)$$

Opening pada Citra Grayscale

Proses opening pada citra *grayscale* dapat dinyatakan sebagai berikut.

$$O_G(A,B) = D_G(E_G(A,B),B)$$

Operasi opening pada citra *grayscale* ini akan memberikan efek penurunan intensitas bagian citra terang yang berukuran lebih kecil

dari SE. Sedangkan untuk bagian citra terang yang lebih besar dari SE tidak akan berubah.

7.1.4 Closing

Sama seperti *opening*, operasi *closing* merupakan penggabungan antara operasi erosi dan dilasi. Hanya saja operasi dilasi dilakukan terlebih dahulu kemudian baru diikuti dengan operasi erosi. Operasi *closing* dinyatakan sebagai berikut.

$$C(A,B) = A \cdot B = E(D(A, - B), -B)$$

Hasil operasi *closing* hampir mirip seperti hasil operasi dilasi yakni memperbesar batas luar dari objek *foreground* dan juga menutup lubang kecil yang terletak di tengah objek, namun hasil operasi *closing* tidak sebesar hasil dilasi. Hasil dilasi akan menyebabkan pembengkakan bentuk keseluruhan objek. Efek ini dapat dikurangi dengan menerapkan proses erosi setelah proses dilasi tersebut.

Closing pada citra Grayscale

Proses *closing* pada citra *grayscale* dapat dinyatakan sebagai berikut :

$$C_G(A,B) = E_G(DG(A,\hat{B}),B)$$

7.1.5 Hit and Miss Transform

Transformasi *hit-and-miss* merupakan metode dasar morfologi untuk mendeteksi bentuk. Sama seperti proses morfologi lainnya, proses *hit and miss* dapat dilakukan dengan membandingkan setiap pixel citra input dengan nilai pusat SE dengan cara melapiskan SE dengan citra sehingga pusat SE tepat dengan posisi pixel citra yang diproses. Jika semua pixel pada SE tepat sama dengan semua nilai pixel citra maka pixel input diset nilainya dengan nilai pixel *foreground*, bila tidak maka input pixel diberi nilai pixel *background*. Proses serupa dilanjutkan dengan menggerakkan SE pixel demi pixel pada citra input.

Gambar 5.1 menunjukkan 4 contoh kernel yang dapat digunakan mendeteksi suatu pixel sudut pada citra biner dengan orientasi sudut berbeda melalui operasi *hit and miss*. Tiga kernel terakhir adalah versi transformasi dari kernel pertama.

1			1			0	0	0	0
0	1	1	1	1	0	1	1	0	0
0	0		0	0		1			1

Gambar 5.1Empat struktur kernel dengan orientasi sudut poin yang berbeda

Setelah mendapatkan citra hasil Hit and Miss untuk masing-masing kernel, kemudian dilanjutkan dengan operasi OR pada semua citra untuk secara bersamaan untuk mendapatkan hasil akhir yang menunjukkan semua orientasi sudut.

42

7.1.6 Thinning

Thinning merupakan suatu operasi morfologi, terkadang seperti erosi atau opening. Thinning mengubah bentuk asli citra biner menjadi citra yang menampilkan batas-batas objek/foreground hanya setebal satu pixel. Sepintas, thinning mempunyai kemiripan dengan deteksi tepi dalam hal output dari citra yang dihasilkan. Kedua proses tersebut sama-sama menampilkan batas obyek pada citra. Namun, tetap saja ada perbedaan antara thinning dengan deteksi tepi dari sisi cara kerjanya sebagai berikut.

4

- Deteksi tepi : merubah gray level atau intensitas citra menjadi citra yang menampilkan batas-batas/boundaries obyek berdasarkan kekontrasan warna antarpixel.
- Thinning : mereduksi pixel pada obyek biner menjadi pixel yang bernilai sama dengan nilai pixel pada background. Keluaran berupa citra biner dengan informasi berupa batas-batas obyek berdasarkan pixel dengan ketebalan satu pixel.

Tujuan thinning adalah untuk menghilangkan pixel objek (foregroundobject) pada citra biner. Thinning biasanya digunakan pada proses skeletonisasi.

Operasi *thinning* dapat dilakukan dengan memanfaatkan operator *Hit and Miss*. Sebagai berikut.

$$\text{Thin}(i,j) = I - \text{HitMiss}(i, j)$$

Dengan operasi pengurangan di atas menyatakan pengurangan logis (*logical subtraction*) yang didefinisikan sebagai berikut

$$X - Y = X \wedge \text{NOT } Y$$

7.1.7 Thickening

Thickening digunakan untuk memperluas daerah dari suatu objek, terkadang seperti operasi dilasi dan closing. Mirip seperti operasi *thinning*, operasi *thickening* dapat dilakukan dengan memanfaatkan operasi *hit and miss*. Operasi ini dilakukan untuk menambahkan beberapa wilayah objek dalam citra biner. Seperti operasi morfologi lainnya, hasil dari *thickening* sangat ditentukan oleh kernel yang digunakan. Hubungan antara operasi *thickening* dengan *hit and miss* dapat digambarkan dengan rumus di bawah ini

$$\text{Thicken}(I,J) = I \vee \text{HitMiss}(I,J)$$

Operasi *thickening* sering digunakan untuk menentukan aproksimasi *convex hull* dari suatu objek.

7.2 Edge Detection

Tepian dari suatu citra mengandung informasi penting dari citra bersangkutan. Tepian citra dapat merepresentasikan objek-objek yang terkandung dalam citra tersebut, bentuk, dan ukurannya serta terkadang juga informasi tentang teksturnya. Tepian citra adalah posisi dimana intensitas pixel dari citra berubah dari nilai rendah ke nilai tinggi atau sebaliknya. Deteksi tepi umumnya adalah langkah awal melakukan segmentasi citra.

7.2.1 Edge, Line, dan Point Detection

Arah dari tepian citra bermacam-macam. Ada yang lurus dan ada yang seperti kurva. Terdapat berbagai metode deteksi tepi yang dapat digunakan untuk mendeteksi berbagai macam jenis tepian. Setiap teknik memiliki keunggulan masing-masing. Satu teknik deteksi tepi mungkin dapat bekerja sangat baik dalam suatu aplikasi

tertentu namun sebaliknya belum tentu dapat bekerja secara maksimal dalam aplikasinya.

Tepian citra dapat dilihat melalui perubahan intensitas pixel pada suatu area. Berdasarkan perbedaan perubahan intensitas tersebut, tepian dapat dibagi menjadi 4 jenis, antara lain sebagai berikut.

a. Step

Tepian jenis *step* merupakan tepian citra yang terbentuk dari perubahan intensitas citra secara signifikan dari tinggi ke rendah ataupun sebaliknya.

b. Ramp

Tepian jenis ini terbentuk dari perubahan intensitas citra secara perlahan. Perubahan secara perlahan dapat dilihat pada bentuk kurva yang semakin tinggi dengan perubahan kontinu.

c. Line

Tepian jenis ini ditandai dengan perubahan intensitas secara drastis dari intensitas rendah-tinggi-rendah atau sebaliknya.

d. Step-line

Tepian *step-line* merupakan gabungan dari tepian jenis *step* dan *line*. Tepian jenis ini ditandai dengan peningkatan intensitas yang tajam dalam interval dan kemudian ditandai dengan penurunan yang tidak signifikan, sehingga perubahan intensitas selanjutnya berlangsung stabil.

7.2.2 Deteksi Tepi Berdasarkan Turunan Pertama

Terdepat berbagai operator deteksi tepi yang telah dikembangkan berdasarkan turunan pertama (*first order dervative*), diantaranya operator Robert, Operator Sobel, operator Prewitt, operator Krisch, dan operator Canny. Konsep dasar dari perhitungan deteksi tepi menggunakan turunan pertama adalah dengan memanfaatkan perbedaan nilai suatu pixel dengan pixel tetangganya, seperti persamaan berikut ini.

$$\text{delta}_y = f(x,y) - f(x,y+1)$$

$$\text{delta}_x = f(x,y) - f(x-1,y)$$

delta_y menyatakan perbedaan nilai pixel dalam arah y, sedangkan delta_x menyatakan perbedaan nilai pixel dalam arah x. Dalam proses konvolusi, delta_y diwakili dengan matrix 2 x 1 seperti berikut.

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Sedangkan delta_x diwakili dengan matrik 1 x 2 berikut

$$[-1 \quad 1]$$

delta_y dan delta_x dapat digunakan untuk menghitung nilai gradien sebagai berikut.

$$\text{Gradient} = \tan^{-1}\left(\frac{\text{delta}_y}{\text{delta}_x}\right)$$

Berikut adalah penjelasan dari masing-masing operator deteksi tepi sederhana dan memiliki tingkat komputasi yang cepat.

Operator Robert

Operator Robert merupakan suatu teknik deteksi tepi sederhana dan memiliki tingkat komputasi yang cepat. Pada umumnya operator ini digunakan untuk citra *grayscale*. Operator Robert dapat digambarkan dengan dua matriks berukuran 2 x 2 seperti berikut.

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Matriks di atas dapat merespon tepian maksimal hingga 45° . Kedua matriks di atas dapat diterapkan pada citra secara terpisah, baik untuk mendapatkan tepian horisontal maupun tepian vertikal. Kedua matriks ini juga dapat dikombinasikan dengan salah satu dari cara berikut untuk mendapatkan hasil gradien dari tepi, yaitu:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

$$|G| = |G_x| + |G_y|$$

Operator Sobel

Operator Sobel lebih sensitif terhadap tepian diagonal daripada tepian vertikal dan horisontal. Operator ini terbentuk dari matriks berukuran 3×3 seperti berikut ini.

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Sama halnya dengan operator Robert, operator Sobel juga dapat diterapkan secara terpisah untuk mendapatkan gradien horisontal dan gradien vertikalnya. Untuk mendapatkan gradien gabungan dapat digunakan dengan menggunakan persamaan berikut.

$$|G| = \sqrt{G_x^2 + G_y^2}$$

$$|G| = |G_x| + |G_y|$$

Operator Prewitt

Operator Prewitt merupakan kebalikan dari operator Sobel. Operator ini lebih sensitif terhadap tepian horisontal dan vertikal daripada tepian diagonal. Operator ini terbentuk dari matriks berukuran 3×3 seperti berikut ini.

$$G_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ 1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Operator Krisch

Operator ini mirip dengan operator Prewitt. Tapis Krisch dapat dipetakan ke delapan arah mata angin sehingga operator ini dapat menentukan gradien dari kedelapan arah tersebut. Operator Krisch terbentuk dari matriks 3 x 3 seperti berikut ini.

$$G_x = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Deteksi Tepi Isotropik

Deteksi isotropik dibentuk dari matriks 3 x 3 pada bagian horizontal maupun bagian vertikal. Matriks tersebut dapat dijabarkan sebagai berikut.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$$

46

Operator Canny

Deteksi tepi Canny dapat mendeteksi tepian yang sebenarnya dengan tingkat kesalahan minimum. Dengan kata lain, operator Canny didesain untuk menghasilkan citra tepian yang optimal. Berikut adalah langkah-langkah dalam melakukan deteksi tepi Canny.

1. Menghilangkan derau yang ada pada citra dengan mengimplementasikan tapis Gaussian. Proses ini akan menghasilkan citra yang tampak sedikit buram. Hal ini dimaksudkan untuk mendapatkan tepian citra yang sebenarnya. Bila tidak dilakukan maka garis-garis halus juga akan dideteksi sebagai tepian. Berikut ini adalah salah satu contoh tapis Gaussian dengan $\sigma = 1.4$.

$$\frac{1}{15} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

13

2. Melakukan deteksi tepi dengan salah satu operator deteksi tepi seperti Roberts, Prewitt, atau Sobel dengan melakukan pencarian secara horisontal (G_x) dan secara vertikal (G_y).

13

3. Menentukan arah tepian yang ditemukan dengan menggunakan rumus sebagai berikut.

$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

Selanjutnya membagi ke dalam 4 warna sehingga garis dengan arah berbeda dan memiliki warna yang berbeda. Pembagiannya adalah 0-22,5 dan 157,5- 180 derajat berwarna kuning, 22,5-67,5 berwarna hijau, dan derajat 67,5-157,5 berwarna merah.

4. Memperkecil garis yang muncul dengan menerapkan *nonmaximumsuppression* sehingga menghasilkan garis tepian yang lebih ramping.
5. Langkah terakhir adalah binerisasi dengan menerapkan dua buah *thresholding*.

7.2.3 Deteksi Tepi Berdasarkan Turunan Kedua

Pengembangan selanjutnya dari deteksi tepi mengarah pada turunan kedua (*second order detection*). Setiap operator deteksi tepi yang telah dijelaskan sebelumnya, akan menampilkan area yang terdeteksi sebagai tepian. Tepian yang dihasilkan masih berupa garis yang tebal. Idealnya, suatu operator harus dapat menampilkan pusat tepian. Jika suatu deteksi tepi menghasilkan tepian yang lebar maka sangat sulit untuk menentukan pusat dari tepian tersebut. Oleh karena itu perlu dilakukan proses *thinning* untuk mengurangi lebar tepian menjadi 1 pixel. Deteksi tepi dari turunan kedua dapat menghasilkan tepian citra yang lebih baik karena menghasilkan tepian yang lebih tipis. Salah satu operator dari turunan kedua adalah *Laplacian of Gaussian* (LOG).

Laplacian of Gaussian (LOG)

Laplacian of Gaussian adalah salah satu operator deteksi tepi yang dikembangkan dari turunan kedua. Operator Laplacian of Gaussian sangat berbeda dengan operator yang lainnya, karena operator Laplacian berbentuk omny directional (tidak honisontal tidak vertikal). Operator ini akan menangkap tepian dari semua arah dan menghasilkan tepian yang lebih tajam dari operator yang lainnya. Laplacian of Gaussian terbentuk dari proses Gaussian yang diikuti operasi laplace. Hasilnya tidak terlalu terpengaruh oleh derau karena fungsi Gaussian adalah mengurangi derau. Laplacian mask meminimalisasi kemungkinan kesalah deteksi tepi.

Fungsi dari Laplacian of Gaussian adalah sebagai berikut.

$$\text{LOG}(x,y) = \frac{1}{\pi\sigma^4} \left[1 - \frac{x^2+y^2}{2\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Fungsi di atas merupakan fungsi untuk membentuk tapis dari Laplacian of Gaussian. Salah satu contoh operator LOG dalam matriks 3 x 3 berikut.

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Tugas Mandiri

1. Jelaskan mengenai *Opening* dan *Closing*, sertakan buatn operasi dari *opening* dan *closing*.
2. Apa kegunaan dari fungsi *Thickening*.
3. Jelaskan *Laplacian of Gaussian* (LOG).

KOMPRESI DATA CITRA (*IMAGE DATA COMPRESSION*)

Setelah membaca bab ini, pembaca akan memahami pengertian tentang kompresi, konsep, dan jenis data berlebihan meliputi data berlebihan *coding*, *interpixel*, dan *psychovisual*, model sistem kompresi data, kriteria untuk mengukur hasil kompresi, berbagai teknik pengkodean meliputi Shannon Fano, Huffman, Zero Compression, RLE, *Difference Coding*, *Arithmetic Coding*, LZW, kompresi berbasis transformasi, dan teknik kompresi dengan Wavelet.

8.1 Pengantar Kompresi Data

22

Proses kompresi merupakan proses mereduksi ukuran suatu data untuk menghasilkan representasi digital yang padat atau mampat (*compact*) namun tetap dapat mewakili kuantitas informasi yang terkandung pada data tersebut. Pada citra, video, dan audio, kompresi mengarah pada minimisasi jumlah *bit rate* untuk representasi digital. Pada beberapa literatur, istilah kompresi sering disebut juga *source coding*, *data compression*, *bandwidth compression*, dan *signal compression*.

Data dan informasi adalah dua hal yang berbeda. Pada data terkandung suatu informasi. Namun tidak semua bagian data terkait dengan informasi tersebut atau pada suatu data terdapat bagian-bagian data yang berulang untuk mewakili informasi yang sama. Bagian data yang tidak terkait atau bagian data yang berulang tersebut disebut dengan data berlebihan (*redundancy data*). Tujuan daripada kompresi data tiada lain adalah untuk mengurangi data berlebihan tersebut sehingga ukuran data menjadi lebih kecil dan lebih ringan dalam proses transmisi.

Berikut dijelaskan beberapa contoh redundansi data dan manfaat proses kompresi.

Contoh pertama adalah transmisi citra pada mesin faksimili. Secara umum pada mesin faksimili, dokumen di-scan dan didigitalisasi. Suatu kertas berukuran 8.5 x 11 inch di-scan dengan resolusi 200 dpi akan menghasilkan 3.74 Mbits. Untuk mengirim (mentransmisi) data tersebut dengan menggunakan modem 14.4 kbits/second akan dibutuhkan waktu sekitar 5.62 menit. Melalui proses kompresi, waktu transmisi tersebut hanya menjadi 17 detik. Di sini proses kompresi berguna untuk menurunkan biaya transmisi (*cost transmission*).

Contoh kedua adalah aplikasi penyimpanan video pada media CD-ROM. Suatu video bergerak dengan 30 *frame per second* (fps) dan resolusi 720 x 480 pixel, akan menghasilkan ukuran video sekitar 20.736 Mbytes per detik. CD-ROM yang berukuran 650 Mbytes hanya dapat menyimpan video dengan durasi sekitar 31 detik. Dengan proses kompresi, CD-ROM akhirnya bisa menyimpan video dalam durasi sekitar 74 menit.

Berikut disajikan tabel perbandingan *data rate* yang dibutuhkan untuk data yang sebelum dan setelah kompresi pada berbagai aplikasi, citra, video, dan audio.

Tabel 8.1 Aplikasi Kompresi pada Citra, Video, dan Audio

Aplikasi	Data Rate	
	Tidak terkompresi	Terkompresi
Suara 11 8 ksamples/s, 8 bits/sample	64 kbps	2-4 kbps
Slow-motion video (10 fps) Ukuran frame 176 x 120, 24 11 bits/pixel	5.07 Mbps	8-16 kbps
Konfrensi Audio 8 ksamples/s, 16 bits/sample	128 kbps	6-64 kbps
Konfrensi video (16 fps) Ukuran frame 352 x 240, 24 11 bits/pixel	30.41 Mbps	64-768 kbps
Digital Audio (stereo) 44.1 ksamples/s, 16 bits/sample	1.5 Mbps	128-768 kbps
Video file transfer (15 fps) Ukuran frame 352 x 240, 24 11 bits/pixel	30.41 Mbps	384 kbps
Digital video pada CD-ROOM (30 fps) 78 Ukuran frame 352 x 240, 24 bits/pixel	60.83 Mbps	1.5-4 Mbps
Aplikasi	Data Rate	
	Tidak terkompresi	Terkompresi
Broadcast video (30 fps) Ukuran frame 720 x 480, 24 11 bits/pixel	248.83 Mbps	3-8 Mbps
HDTV (59.94 fps) Ukuran frame 1280 x 720, 24 bits/pixel	1.33 Gbps	20 Mbps

Berdasarkan contoh-contoh di atas, kompresi data bermanfaat dalam hal memperkecil ukuran data, mempermudah proses penyimpanan data pada suatu media, dan mempercepat proses transmisi pada berbagai jenis koneksi.

Ada beberapa faktor mengapa pada citra, video, dan audio sangat tepat dilakukan proses kompresi. Faktor-faktor tersebut antara lain sebagai berikut.

- a. Pada suatu citra tunggal atau pada frame tunggal video dapat terjadi korelasi yang signifikan antara suatu pixel dengan pixel tetangga. Korelasi ini disebut dengan korelasi spasial (*spatial correlation*).
- b. Pada data yang diambil dari beberapa sensor (multi sensor), terdapat korelasi yang signifikan antarsampel yang diambil oleh sensor-sensor tersebut. Korelasi ini disebut dengan korelasi spektral (*spectral correlation*).
- c. Pada data temporal seperti video, terdapat korelasi yang signifikan antara sampel pada segmen waktu yang berbeda. Korelasi ini disebut sebagai korelasi temporal (*temporal correlation*).
- d. Pada suatu data terdapat informasi yang tidak relevan dengan sudut pandang persepsi mata.

Faktor-faktor tersebut di atas yang memunculkan data berlebihan sehingga harus diperlukan proses kompresi.

8.2 Data Berlebihan (Data Redundancy)

Pada citra, video, maupun audio terdapat beberapa faktor yang memunculkan data berlebihan.

Data berlebihan merupakan isu penting dalam kompresi citra. Data berlebihan ini dapat dinyatakan secara matematis. Bila n_1 dan n_2 menyatakan jumlah satuan (unit) informasi dalam dua himpunan data (*data set*) yang mewakili data yang sama maka data berlebihan relatif (*relative data redundancy*) R_D dari himpunan data pertama dapat dinyatakan sebagai berikut.

$$R_D = 1 - \frac{1}{C_R}$$

Dengan C_R merupakan rasio kompresi (*compression ratio*) yang dinyatakan sebagai berikut.

$$C_R = \frac{n_1}{n_2}$$

Bila $n_1 = n_2$ maka $C_R = 1$ dan $R_D = 0$, berarti bahwa data set pertama tidak mengandung data berlebihan. Bila $n_2 \ll n_1$ (n_2 jauh lebih kecil dari n_1) maka C_R mendekati tak berhingga, sehingga R_D mendekati 1. Ini berarti terjadi kandungan data berlebihan sangat tinggi. Bila $n_2 \gg n_1$ (n_2 jauh lebih besar dari n_1) maka C_R mendekati nol, sehingga R_D mendekati minus tak berhingga. Ini berarti data set kedua mengandung informasi jauh lebih banyak dibandingkan data set pertama. Secara umum, nilai C_R dan R_D berturut-turut berada dalam interval $(1, \infty)$ dan $(-\infty, 1)$. Dalam praktik, rasio kompresi 20 (atau 20 : 1) berarti data set pertama mengandung 20 satuan (unit) informasi untuk setiap 1 unit pada data set kedua (atau pada data terkompresi). Dengan kata lain untuk kasus citra, citra asli (citra belum termampatkan) mengandung 20 bit informasi untuk setiap 1 bit pada data terkompresi. Redundansi 0.8 berarti 80% data pada data set pertama adalah berlebihan.

Dalam kompresi citra digital, terdapat 3 jenis data berlebihan, yaitu *coding redundancy*, *interpixel redundancy*, dan *psychovisual redundancy*.

8.2.1 Coding Redundancy

Pada suatu citra, nilai intensitas setiap pixel diwakili oleh sejumlah bit. *Coding redundancy* terjadi bila jumlah bit yang digunakan untuk mewakili suatu nilai intensitas berlebihan.

Sebagai contoh, suatu citra memiliki 8 level nilai intensitas, yaitu 0, 1, 2, 3, 4, 5, 6, dan 7. Probabilitas kemunculan setiap nilai intensitas pixel tersebut berturut-turut adalah 0.19, 0.25, 0.21, 0.16, 0.08, 0.06, 0.03, dan 0.02. Probabilitas ini dapat dihitung dengan menggunakan histogram citra, yaitu dengan membagi banyaknya (frekuensi) kemunculan suatu nilai intensitas dengan banyaknya pixel yang terdapat dalam citra. Setiap nilai pixel atas dapat diwakili (dikodekan)

dengan sejumlah bit dengan panjang tetap (*fixed-length*) dengan panjang berbeda-beda (*variable-length*). Tabel 8.2 berikut menyajikan pengkodean setiap nilai pixel baik dengan panjang tetap maupun panjang berbeda.

Tabel 8.2 Contoh Pengkodean Pixel dengan Fixed Code dan Variable Code

(data diambil dari Gonzale dkk, 1993)

Nilai Pixel	Probabilitas	Fixed Code	Panjang Kode Tetap	Variable Code	Panjang Kode Variabel
k	P_k	Kode 1	l_{1k}	Kode 2	l_{2k}
0	0.19	000	3	11	2
1	0.25	001	3	01	2
2	0.21	010	3	10	2
3	0.16	011	3	001	3
4	0.08	100	3	0001	4
5	0.06	101	3	00001	5
6	0.03	110	3	000001	6
7	0.02	111	3	000000	6

Rata-rata panjang bit yang digunakan untuk mengkodekan setiap pixel dapat dihitung dengan rumus sebagai berikut.

$$L_{rata} = \sum_{k=0}^{L-1} l_k P_k$$

Dengan L menyatakan banyaknya level intensitas.

Untuk *fixed code*, rata-rata panjang bit yang digunakan untuk mengkodekan setiap nilai pixel adalah 3. Sedangkan bila menggunakan *variable code*, rata-rata panjang bit yang digunakan untuk mengkodekan setiap pixel sebagai berikut.

$$\begin{aligned}
L_{rata} &= \sum_{k=0}^{L-1} l_k P_k \\
&= \\
&2(0.19)+2(0.25)+2(0.21)+3(0.16)+4(0.08)+5(0.06)+6(0.03)+3(0.02) \\
&= 2.7 \text{ bits}
\end{aligned}$$

Rasio kompresi dari pengkodean di atas adalah :

$$C_R = \frac{3}{2.7} = 1.11$$

Sedangkan redundansi datanya adalah :

$$R_D = 1 - \frac{1}{1.11} = 0.099$$

Hal tersebut berarti sekitar 10% data yang dihasilkan dengan pengkodean *fixed length* berlebihan.

8.2.2 Interpixel Redundancy

Data berlebihan model ini terjadi karena adanya korelasi antarpixel dengan pixel tetangganya. Nilai intensitas suatu pixel dapat diprediksi dari pixel tetangga. Informasi yang dibawa oleh suatu pixel terhadap keseluruhan citra (objek) relatif tidak besar atau tidak signifikan. Ini berarti pixel tersebut berlebihan. *Interpixel redundancy* sering juga disebut dengan *spatial redundancy*, *geometric redundancy*, dan *interframe redundancy*.

8.2.3 Psychovisual Redundancy

Psychovisual redundancy berbeda dengan *coding* maupun *interpixel redundancy*. Data berlebihan model ini berkaitan secara langsung dengan informasi visual yang dilihat dengan mata. Variasi intensitas pada suatu citra (lihat bab sebelumnya tentang fenomena *mach band*) dilihat sebagai intensitas yang konstan oleh mata. Fenomena ini terjadi karena mata manusia tidak memiliki sensitivitas yang sama terhadap semua informasi visual. Dengan kata lain mata tidak dapat membedakan antara informasi penting dengan informasi kurang penting. Informasi inilah yang dikatakan sebagai *psychovisual redundancy*. Informasi tersebut dapat dihilangkan karena tidak akan berdampak secara signifikan pada kualitas persepsi terhadap citra. Proses menghilangkan data berlebihan *psychovisual* dapat

menyebabkan hilangnya informasi secara kuantitas. Hal ini menyebabkan dalam proses kompresi, menghilangkan data berlebihan psychovisual dapat menghasilkan rekonstruksi citra kompresi yang tidak sempurna, dalam arti terjadi perbedaan antara citra hasil kompresi dengan citra asli.

8.3 Model Kompresi Citra

Model umum kompresi citra (Gonzales dkk, 1993). Model tersebut dirancang agar dapat mereduksi ketiga jenis data berlebihan (redundansi data) yang mungkin muncul pada citra. Model umum kompres citra terdiri atas dua bagian besar yaitu *encoder* dan *decoder*. *Encoder* berfungsi membuat representasi simbol-simbol (kode) dari citra input $f(x,y)$. Dengan kata lain *encoder* berfungsi membuat citra kompresi (citra terkode) dari citra input. Setelah melalui transmisi pada kanal (*channel*), citra terkompresi tersebut masuk ke sistem *decoder*. Pada sistem ini, citra terkompresi akan direkonstruksi kembali untuk menghasilkan citra output $f(x,y)$. Informasi pada citra $f(x,y)$ dan $f(x,y)$ dapat sama persis atau tidak terjadi kehilangan informasi (*lossless information*), dapat juga tidak sama atau terjadi kehilangan informasi (*lossy information*)

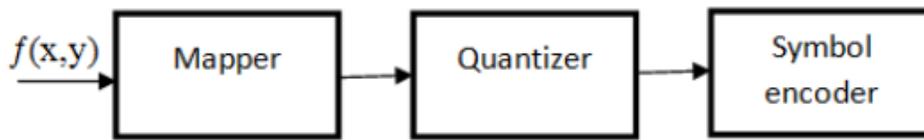
Channel encoder berfungsi agar data yang dihasilkan dari *source encoder* menjadi lebih tahan terhadap derau (*noise*). Hasil proses *channel encoder* ini kemudian akan dibalik kembali oleh *channel decoder*. Salah satu teknik terkenal yang digunakan untuk *channel encoding* adalah teknik Hamming. Bila *channel* antara *channel encoder* dan *channel decoder* terbebas dari derau maka kedua jenis *channel* tersebut dapat dihilangkan.

Source Encoder dan Decoder

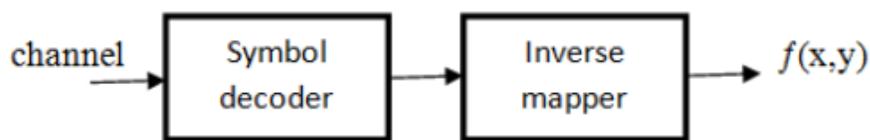
Skema *source encoder* dan *decoder* berturut-turut ditunjukkan pada gambar 6.2 (a) dan 6.2 (b). *Source encoder* berguna untuk mengurangi berbagai data berlebihan yang muncul pada citra. Sistem ini terdiri atas tiga proses, yaitu *mapper*, *quantizer*, dan *symbol encoder*.

Mapper berfungsi mengubah citra input ke dalam suatu format untuk menghilangkan *interpixel redundancy*. Operasi ini dapat mengurangi atau dapat juga tidak ukuran citra asli dan biasanya bersifat *reversible*

(dapat dibalik) artinya informasi pada cita hasil rekonstruksi sama persis dengan citra asli.



(a)



(b)

Gambar 8.1 Source encoder dan Source decoder.

Quantizer berfungsi mengurangi data berlebihan *psychovisual*. Operasi dilakukan dengan melakukan proses kuantisasi terhadap hasil dari tahap sebelumnya. Hasil dari proses kuantisasi pada umumnya bersifat *irreversible* (tidak dapat dibalik) artinya informasi pada citra hasil tidak sama dengan informasi pada citra asli karena terdapat informasi yang hilang. Bila ingin tidak terjadi informasi yang hilang pada citra hasil maka tahap (blok) *quantizer* harus dihilangkan.

Symbol encoder berfungsi membuat kode untuk merepresentasikan keluaran dari *quantizer* dan memetakan setiap keluaran tersebut ke dalam kode. Panjang kode dapat bersifat tetap (*fixed length code*) dapat juga bervariasi (*variable length code*). Kode yang dihasilkan dari tahap ini dapat menghilangkan data berlebihan *coding (coding redundancy)* dan biasanya bersifat *reversible*.

Source decoder merupakan proses kebalikan dari *source encoder*. Pada *source decoder* tidak terdapat proses (tahap) *invers quantizer* karena proses *quantizer* bersifat tidak dapat dibalik.

Tugas Mandiri

1. Sebutkan fungsi dari *MapperQuantizerSymbol encoder*
2. Jelaskan perbedaan dari *Psychovisual redundancy*, *Coding redundancy* maupun *interpixelredundancy*.
3. Jelaskan ada beberapa faktor mengapa pada citra, video, dan audio sangat tepat dilakukan proses kompresi.

PENGENALAN POLA

9.1 Pengertian Pengenalan Pola

Istilah pengenalan pola juga sering disebut dengan *pattern recognition* telah berkembang pesat dalam penelitian komputasi citra. Istilah “pola” erat hubungannya dengan objek yang mempunyai banyak ciri. Misalnya wajah sebagai objek mempunyai ciri rasio antara lebar mata dan tinggi mata, rasio antara lebar bibir dan tinggi bibir, rasio antara jarak pusat mata kiri-kanan dan lebar bibir. Nilai rasio-rasio tersebut merupakan parameter ciri suatu wajah. Ini artinya objek yang berbeda tentunya mempunyai ciri yang berbeda pula. Istilah “pengenalan” erat hubungannya dengan pengelompokan dari suatu objek berdasarkan ciri-ciri yang dimiliki. Suatu objek lebih mudah dikenali, jika mempunyai nilai pembeda yang besar dengan objek-objek lain. Untuk mengenali suatu objek, maka standart ciri yang disyaratkan harus sama. Pengenalan pola dapat diartikan sebagai sebuah proses untuk mengenali objek secara otomatis berdasarkan ciri yang didasarkan pada data pelatihan.

Pengenalan pola merupakan serangkaian proses yang dilakukan secara serial, mulai dari proses deteksi/segmentasi, proses ekstraksi dan proses pengukuran kemiripan atau proses pengenalan. Deteksi/segmentasi merupakan proses untuk memperoleh lokasi objek secara tepat yang akan dikenali. Proses segmentasi/deteksi ini

terkadang juga dapat diabaikan, ketika objek yang akan dikenali sudah terdeteksi secara tepat lokasinya, misalnya pengenalan wajah ORL. Data citra ORL yang disediakan dapat langsung diekstraksi dan diikuti dengan pengukuran kemiripan tanpa melalui proses deteksi terlebih dahulu.

Untuk mengenali pola suatu objek, ada dua pendekatan yang sering digunakan yaitu metode yang menggunakan teori bahasa formal automata atau dikenal dengan nama pendekatan pengenalan secara sintaktik. Model pengenalan sintaktik relatif lebih sulit dibandingkan dengan pendekatan pertama.

Aplikasi pengenalan pola telah banyak diimplementasikan dalam bidang kedokteran atau dikenal dengan istilah biomedik, biometrik, dan pengenalan untuk objek-objek lainnya. Sebagai contoh bidang biometrik telah mengubah cara mengenali seseorang yang semula menggunakan NIP atau Identitas lain bergeser menggunakan perilaku dan ciri yang melekat pada tubuh seperti wajah, sidik jari atau pun ciri lainnya. Pengenalan seseorang menggunakan karakteristik fisik yang melekat pada tubuh seseorang disebut biometrik. Beberapa modalitas fisik yang umum digunakan dalam biometrik antara lain:

1. Sidik jari
2. Wajah
3. Mata Iris
4. Telapak tangan
5. Telinga

Sedangkan biometrik yang berhubungan dengan tingkah laku antara lain:

1. Gaya menulis seseorang
2. Gaya berjalan
3. Tanda tangan

9.2 Aplikasi Pengenalan Pola Untuk Mengenali Citra Sidik Jari

Sidik jari merupakan salah satu ciri yang dimiliki oleh seseorang dan mempunyai bentuk yang unik, karena setiap orang memiliki pola sidik jari yang berbeda. Seperti yang diketahui banyak orang, bahwa sidik jari telah banyak digunakan untuk mengenali seseorang, termasuk ketika terjadi kasus-kasus kejahatan melalui sidik jari yang telah ditinggalkan oleh pelaku. Sebelum teknologi sidik jari digunakan, sidik jari secara manual telah digunakan sebagai bentuk ciri pada KTP, Paspor dan keperluan lain seperti pilkada atau pemilu. Dalam hal ini seseorang menempelkan sidik jarinya pada tempat yang telah disediakan setelah sidik jarinya dibasahi dengan tinta. Namun dengan berkembangnya teknologi biometrik, sensor sidik jari telah banyak digunakan untuk mengambil data sidik jari seseorang. Dalam biometrik, pengenalan sidik jari dilakukan dengan mencocokkan minusi seseorang dengan basis data minusi yang telah disimpan sebelumnya. Minusi merupakan detil pola dari sidik jari seseorang, diantaranya :

- a. Dua titik percabangan
- b. Tiga titik bercabangan
- c. Titik terminasi

Namun yang sering digunakan dalam identifikasi seseorang adalah dua titik percabangan dan titik terminasi. Beberapa basis data sidik jari yang dapat digunakan untuk keperluan penelitian antara lain adalah :

1. UPEK
2. CASIA
3. SfinGe
4. FVC2006
5. FVC2004
6. FVC2002
7. FVC2000

8. Neurotechnologija
9. Biometrix
10. Innovatrics

9.3 Aplikasi Pengenalan Pola Untuk Mengenali Citra Wajah

Citra wajah umumnya diambil menggunakan kamera digital. Citra wajah merupakan salah satu modalitas dalam biometrik yang mengalami perubahan, meskipun dengan waktu yang lama dibandingkan dengan citra sidik jari. Sedangkan citra sidik jari cenderung tidak berubah, kecuali mengalami kecelakaan yang mengakibatkan sidik jari rusak. Sementara citra wajah akan mengalami perubahan karena beberapa hal, diantaranya adalah :

1. Postur seseorang yang berubah menjadi lebih kurus atau lebih gemuk
2. Perubahan usia akan mengakibatkan wajah berubah
3. Ekspresi seseorang juga akan menyebabkan wajah berubah

Untuk keperluan penelitian pengenalan citra wajah, berikut ini merupakan basis data wajah dengan model pencahayaan dan pose berbeda :

1. AR
2. Richard's MIT
3. CVL
4. MUCT
5. Yale-B
6. YALE-A
7. PIE
8. UMIST
9. Olivetti - Att - ORL
10. JAFFE
11. XM2VTSDB
12. ORL

Ada tiga metode yang digunakan untuk melakukan pengenalan, analisis dan interpretasi citra wajah, diantaranya adalah :

1. Metode yang berbasis penampakan (*Appearance Based*)
2. Metode yang berbasis fitur (*Feature Based*)
3. Metode yang berbasis hybrid (*Hybrid Based*)

Beberapa penelitian pengenalan citra wajah yang berbasis penampakan telah dilakukan oleh beberapa peneliti (Arif Dkk, 2008; Arif Dkk, 2011; Kresimir, 2005), dan yang berbasis fitur (Arif et al., 2009). Metode tersebut menggunakan semua penampakan objek pada citra. Berbeda dengan metode yang berbasis fitur, metode pengenalan wajah berbasis fitur menggunakan ciri-ciri dominan dan tidak mudah berubah yang terdapat pada objek, lebar mata, lebar bibir, ketinggian hidung, kelengkungan wajah dan ciri geometris lainnya. Dalam buku ini juga telah dibahas pengenalan wajah berbasis penampakan, yaitu PCA dan LDA yang dilengkapi dengan kode program matlab versi 6.0.

9.4 Aplikasi Pengenalan Pola Untuk Mengenali Citra Iris Mata

Selain sidik jari dan wajah, modalitas lain yang digunakan dalam biometrik adalah iris mata. Iris mata terletak antara pupil dan sclera. Iris mata antara satu orang dengan lainnya adalah unik (tidak sama). Oleh karena itu iris mata telah banyak digunakan untuk identifikasi seseorang. Pada kasus anak kembar identik, secara genetika mereka mempunyai kesamaan, namun iris matanya tetap berbeda. Identifikasi personal menggunakan iris mata mempunyai tingkat kesalahan yang sangat kecil, sehingga identifikasi personal menggunakan iris mata merupakan alternatif yang telah banyak digunakan. Untuk melakukan penelitian menggunakan iris mata sebagai dominan penelitian, beberapa basis data iris mata yang dapat digunakan antara lain :

1. CASIA CBR
2. MMU1 dan MMU2
3. UBIRIS

9.5 Aplikasi Pengenalan Pola Untuk Mengenali Citra Telapak Tangan

Telapak tangan juga merupakan salah satu modal dalam biometrik, meskipun tidak sepopuler sidik jari, namun citra telapak tangan juga telah banyak diteliti oleh para peneliti sebagai salah satu alternatif untuk mengenali seseorang (Duta *et al.*, 2001; You *et al.* 2002). Citra telapak tangan menggunakan ciri pola yang terdapat pada seluruh telapak tangan seseorang. Pola pada citra telapak tangan adalah pola unik, setiap orang mempunyai pola yang berbeda. Fitur-fitur yang umum digunakan untuk identifikasi telapak tangan seseorang antara lain:

1. Luas
2. Panjang
3. Lebar
4. Garis utama
5. Garis lemah
6. Minusi

Hasil pencocokan citra uji terhadap citra yang berada dalam basis data juga sangat dipengaruhi oleh objek telapak yang diambil secara otomatis, kesalahan dalam mengambil citra akan memberikan dampak kesalahan pada saat identifikasi (Ketut dkk., 2011).

Selain metode pengenalan pola berbasis geometris pada telapak tangan, Penelitian pengenalan telapak seseorang juga ada yang menggunakan metode ekstraksi fitur berbasis penampakan (Bedi, 2013). Hal ini menunjukkan bahwa metode berbasis penampakan juga dapat digunakan untuk identifikasi citra telapak tangan seseorang.

9.6 Aplikasi Pengenalan Pola Untuk Temu Balik Citra

Istilah temu balik citra sering digunakan untuk memperoleh kembali informasi citra, baik berbasis warna ataupun isi. Citra berbasis isi mempunyai korelasi yang kuat dengan tekstur suatu objek. Metode untuk pengambilan kembali citra berbasis isi pada umumnya melakukan analisis tekstur suatu objek seperti *Linear Binary*

Pattern, Gray Level Co-Occurrence Matrix, Gray Level Run Length Matrix, dan *Color Co-Occurrence Matrix*. Analisa tekstur untuk mendapatkan ciri-ciri tekstur pola objek adalah berbasis statistik. Temu balik citra banyak digunakan untuk pencarian citra. Misalnya ingin mencari citra bunga, dengan memasukkan contoh citra bunga, keluaran hasil pencarian adalah citra jenis-jenis bunga. Meskipun dalam kenyataan terdapat beberapa citra yang bukan bunga namun ikut menjadi luaran hasil pencarian, hal semacam ini merupakan kesalahan yang terjadi pada proses pencarian.

9.7 Aplikasi Pengenalan Pola Untuk Pengenalan Tulisan

Pengenalan tulisan tangan atau cetak juga merupakan aplikasi pengenalan pola yang telah banyak diimplementasikan. Mulai dari pengenalan huruf telah banyak diimplementasikan pada layar sentuh handphone untuk mengenali karakter tulisan tangan. Pengenalan plat nomor kendaraan juga telah diteliti oleh banyak peneliti. Bahkan pengenalan tulisan tangan *multi line* juga banyak diteliti (Arif, 2013).

Pada prinsipnya pengenalan tulisan tangan *multi line* juga merupakan pengenalan karakter (huruf), karena untuk mengenali tulisan tangan *multi line* juga melalui proses segmentasi citra setiap baris, kemudian hasil segmentasi citra setiap baris disegmentasi lagi setiap huruf. Hasil segmentasi setiap huruf selanjutnya diukur kemiripannya dengan database huruf pelatihan yang telah disediakan (Arif, 2013).

9.8 Aplikasi Pengenalancitra Pada Bidang Kedokteran

Selain dibidang bioemetrika, pengenalan pola juga telah berkembang pesat pada dunia kedokteran, istilah ini sering dikenal dengan nama biomedik. Penelitian-penelitian pengenalan pola yang berhubungan dengan dunia kedokteran telah banyak dilakukan oleh para peneliti seperti segmentasi pembuluh darah citra fundus (Hoover, 2003 Fengshou *et al.*, 2011, segmentasi optic disk pada citra fundus (Osareh *et al.*, 2002; Handayani *et al.*, 2012), penentuan klasifikasi penyakit *diabetic retinopathy* berdasarkan citra fundus.

Pada prinsipnya untuk melakukan proses pengenalan, umumnya mempunyai beberapa tahapan, diantaranya adalah :

1. Pemrograman awal (deteksi/segmentasi)
2. Ekstraksi fitur digunakan untuk memperoleh ciri objek
3. Pencocokan ciri hasil ekstraksi fitur

Pada tahapan pemrosesan awal, setiap kasus mempunyai prapemrosesan yang berbeda. Contoh pra-pemrosesan yang digunakan antara lain :

1. Konversi dari citra warna menjadi citra keabuan
2. Konversi citra keabuan menjadi citra biner
3. Menghilangkan gangguan citra dalam bentuk noda-noda kecil
4. Merubah ukuran citra menjadi ukuran tertentu
5. Melakukan penipisan objek.

Untuk kasus yang berbeda, pra-pemrosesan mungkin saja berbeda. Jadi tidak ada standart yang harus dilakukan untuk melakukan pra-pemrosesan suatu objek sebelum proses ekstraksi fitur.

Tugas Mandiri

1. Ada tiga metode yang digunakan untuk melakukan pengenalan, analisis dan interpretasi citra wajah, sebutkan dan jelaskan.
2. Jelaskan apa yang dimaksud dengan *pattern recognition*.

APLIKASI SEARCH ENGINE BERBASIS CBIR

10.1 Prototipe Search Engine Flora dan Fauna

63
Searching dan *surfing* di internet telah menjadi bagian dari kehidupan kita sehari-hari, dimana web browser telah mengubah cara dalam mencari dan berinteraksi dengan informasi. Teknologi search engine menyediakan standard interface dengan informasi yang tidak terbatas jumlahnya yang ada di internet. *Search engine* dapat dianggap sebagai suatu radar yang mengarahkan ke *site* atau halaman tertentu. Setelah search engine menemukan halaman (*site*) dimaksud, maka search engine memberikan informasi yang dicari oleh user. Selanjutnya user mungkin kemudian melakukan browsing *site* tersebut atau melanjutkan ke link yang lain. Search engine didefinisikan sebagai suatu halaman web atau *web site* yang mengumpulkan dan mengatur semua *content* yang ada di internet (Levene, 2010) , disini user memasukan suatu pertanyaan atau *query* yang diinginkan, maka search engine akan menyediakan link ke *content* yang sesuai dengan mereka inginkan. Yahoo (www.yahoo.com) merupakan salah satu contoh search engine yang paling lama (sejak tahun 1994), disusul kemudian oleh Infoseek,

AltaVista (www.altavista.com), AlltheWeb (www.alltheweb.com), Ask Jeeves (www.ask.com), dan Google.

Persoalan yang sampai sekarang masih menjadi perhatian para ahli dalam pencarian citra diam dalam suatu database dan Internet (yang tersebar) dengan menggunakan teks atau *keyword* adalah karena hasilnya masih sangat jauh dari yang diharapkan. Pencarian citra dengan menggunakan teks atau *keyword* sangat bias dan sulit dipertanggung jawabkan karena *keyword* sangat sulit untuk menerangkan informasi yang ada pada sebuah citra.

prototipe *search engine* yang digunakan untuk mengetahui perubahan atau penurunan jumlah populasi flora dan fauna yang ada di Indonesia. Disamping itu prototipe *search engine* ini juga ditujukan untuk mendeteksi adanya spesies baru yang belum diketemukan sebelumnya. Dengan *search engine* yang berbasis citra (*image*), maka hasil pencarian atau *matching* akan lebih akurat dibandingkan dengan pencarian yang berbasis teks atau *keyword*.

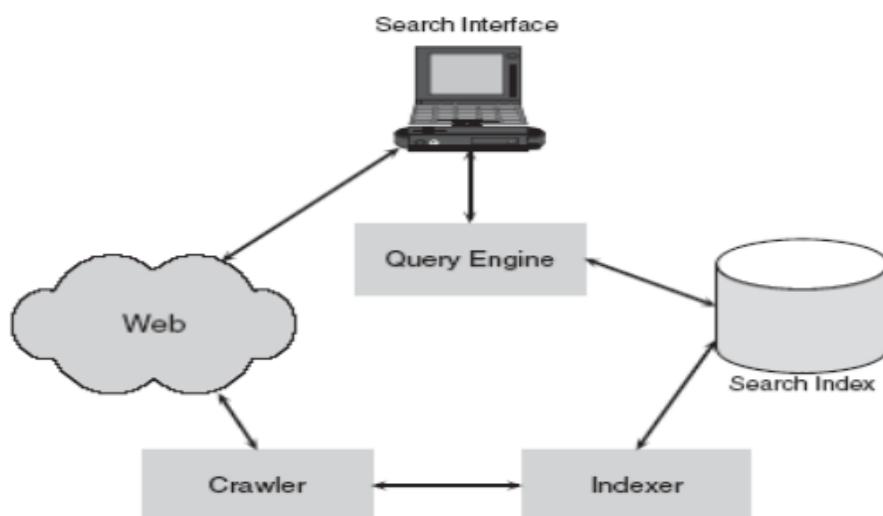
Prototipe *search engine* untuk mencari citra dengan menggunakan fitur citra ditujukan agar hasilnya jauh lebih akurat dibanding dengan menggunakan *keyword*. Memanfaatkan fitur warna, tekstur, struktur dan bentuk suatu citra untuk *matching* citra *query* dengan citra yang ada di database dan atau internet. Fitur yang akan digunakan adalah fitur warna dan fitur tekstur, serta fitur bentuk dan struktur.

Karena besarnya angka kepunahan jenis flora dan fauna yang terjadi sangat tinggi, maka perlu dilakukan usaha untuk mengurangi kecepatan kepunahan tersebut, maka perlu diusahakan teknik untuk memonitor keberadaan dan status jenis-jenis flora dan fauna yang ada di Indonesia. *Search engine* merupakan salah satu tool yang dapat digunakan untuk memonitor jenis-jenis flora yang ada dengan teknik *content based image retrieval* (CBIR). CBIR adalah teknik pencarian gambar dengan berbasis isi atau fitur dari citra. Selama ini pencarian citra masih dikerjakan dengan teknik *searching* yang berbasis teks, dimana akurasi masih jauh dari yang diharapkan. Oleh karena itulah jika monitor kepunahan dan deteksi flora dan fauna dikerjakan dengan menggunakan *search engine* yang berbasis

teks hasilnya tidak akan seperti yang diharapkan, karena dalam search engine berbasis teks dalam mencari citra yang dicari hanya *keyword*-nya, bukan *content* atau fitur dari citra yang dicari. Protipe ini akan menjadi salah satu alat yang digunakan untuk mengetahui kondisi atau status flora dan fauna yang ada di Indonesia, sehingga pemerintah dapat mengambil tindakan - tindakan yang diperlukan untuk menjaga punahnya sumber hayati yang secara tidak langsung menentukan kelangsungan hidup manusia terutama bagi penduduk Indonesia. Denga search engine Pemerintah dan pihak-pihak lain yang *concern* terhadap kelasungan hidup flora dan fauna akan dengan cepat dan tepat untuk mengurangi kecepatan kepunahan flora dan fauna, khususnya yang ada di Indonesia.

10.2 . Search engine

Search engine pada dasarnya digunakan user untuk mencari informasi yang ada di web, dimana menggunakan tahap-tahap *searching* yang dapat dijelaskan sebagai berikut (Levene, 2010): i). *query formulation*, ii). *Selection*, iii). *Navigation*, iv). *Query modification*. Sementara itu banyak sekali jenis arsitektur search engine yang berkembang sampai dengan saat ini, ilustrasi arsitektur search engine dapat dilihat pada gambar 10.1



Gambar 10.1 Arsitektur *search engine*

Web crawler adalah software yang digunakan untuk membaca suatu halaman web dan menguduhnya untuk kemudian di-indeks (*indexing*) yang selanjutnya mengembalikan ke link atau hyperlink berdasarkan halaman yang diunduh tadi (Level, 2010). Sementara itu *indexer* bertanggung jawab terhadap *creating* indeks pencarian atau *search index* dari halaman web yang diterima dari *crawler*. *Query engine* merupakan suatu algoritma yang sangat penting dalam suatu *search engine*. *Query engine* menyediakan interface antara *search index*, *user*, dan *web*. Dalam hal ini *query engine* memproses *query* dari *user* dalam dua tahap, pertama *query engine* memanggil informasi dari *search index* yang relevan dengan halaman web dengan *keyword* yang diketik oleh *user*. Kedua mengurutkan hasil informasi yang terpanggil dari yang relevan sampai pada yang paling tidak relevan.

61

10.3 . Content Based Image Retrieval

Aplikasi *content based image retrieval* telah secara luas digunakan di berbagai bidang kehidupan seperti bidang biomedis, kriminalitas, militer, *commerce*, budaya, pendidikan, hiburan, dan pertanian. Banyak proyek dan penelitian telah dikerjakan, diantaranya adalah yang dikerjakan oleh IBM (<http://www.qbic.almaden.ibm.com>), Virage (www.virage.com), AltaVista (www.altavista.com), Yahoo (www.yahoo.com), and Google (www.google.com).

Dalam *CBIR* untuk menentukan kemiripan suatu citra dalam suatu database dengan citra referensi adalah dengan menghitung *Euclidean distance*-nya. Menurut (Zhang, *et.al.*, 2014) terdapat empat macam ukuran untuk menghitung kemiripan, yaitu berdasarkan kemiripan warna, bentuk, tekstur dan struktur. Isi (*content*) dari suatu citra terdiri dari fitur persepsi (warna, bentuk, tekstur, dan struktur), fitur semantik, peranan, pandangan, dan fitur impresi dan emosi.

Sementara itu untuk mengukur efektivitas dari pencarian gambar dengan menggunakan *precision* dan *Recall*, *precision* adalah jumlah citra yang mirip yang terpanggil dibagi dengan semua citra yang terpanggil (*retrieved*). Sedang *Recall* adalah jumlah citra yang

terpanggil dibagi dengan jumlah citra yang dalam kategori, (Cho, 2004; Thamarai and Dharma, 2014)

$$p = \frac{a}{z}$$

$$r = \frac{b}{y}$$

Dimana p adalah **precision** dan r adalah **recall**, sementara a adalah jumlah citra yang relevan atau mirip terpanggil, z adalah jumlah semua citra yang terpanggil, dan y jumlah citra relevan yang ada dalam database. Sedangkan untuk mengukur efisiensi teknik pencarian citra (*image retrieval*) dihitung dengan menggunakan persamaan berikut:

$$\text{Error rate} = \frac{\text{Jumlah citra yang tidak relevan yang terpanggil}}{\text{Total jumlah citra yang terpanggil}}$$

$$\text{Retrieval efficiency} = \frac{\text{Jumlah citra relevan terpanggil}}{\text{Total jumlah citra yang terpanggil}}$$

Nilai precision atau recall sendiri tidak cukup memberikan informasi, karena nilai recall akan bisa dibuat 1 dengan memanggil semua citra dalam database. Demikian juga precision dapat dibuat tinggi dengan cara memanggil beberapa citra.

10.4 Matching Citra JPEG Berbasis Content

Dalam penelitian ini, citra yang dipakai adalah citra dengan format JPEG (citra dipadatkan). Dalam citra JPEG, citra dibagi-bagi kedalam blok-blok (matrik 8x8), peindeksan (indexing) citra dilakukan dengan menggabung N blok kedalam satu gabungan blok dan setiap blok bertindak sebagai satu ruang dalam gabungan blok. Kunci indeks (indexing key) akan mempunyai 64 elemen dan setiap elemen mempunyai satu lokasi. Pengindeksan kunci dapat dihitung dengan menggunakan formula berikut.

$$h_i = \frac{\sum_{i=1}^N (DCT_i)^2}{N}$$

Dimana h_i adalah elemen ke i dari vektor, dan *indexing key* dapat dijelaskan sebagai berikut:

$H_1 = \{h_0, h_1, \dots, h_{63}\}$, sedangkan untuk menghitung Euclidean distance dapat dihitung dengan formula, $d(h_q, h_k) = \left(\sum_{i=1}^N \frac{|h_i^q - h_i^k|}{64} \right)$, $d(h_q, h_k)$ adalah jarak antara citra referensi dan citra dalam database.

Menurut (Yanuar dan Harjoko, 2014) fitur isi citra mencakup i). bentuk, warna, tekstur, dan spasial, ii). Semantic, iii). Impresi, emosi, dan arti yang berkaitan dengan fitur-fitur yang ada. Sementara itu (Aisyah et al., 2015) menyatakan ada empat cara atau teknik dalam mengukur kemiripan citra, yaitu berdasarkan kesamaan bentuk, warna dan kesamaan hubungan objek. Beberapa aplikasi CBIR dengan menggunakan histogram warna, layout warna, tekstur local, koefisien wavelet, dan bentuk adalah aplikasi yang diperkenalkan oleh Smeulders (Smeulders, 2000 dan Inoune 2005). Secara umum algoritma matching dapat dijelaskan sebagai berikut, misalnya terdapat dua histogram I dan M yang mana tiap histogram mempunyai storage (bin) sebanyak n , maka perpotongan dari kedua histogram dapat didefinisikan sebagai berikut:

$$H(I, M) = \frac{\sum_{j=1}^n \min(I_j, M_j)}{\sum_{j=1}^n M_j}$$

Dikarenakan kekurangan dan kelebihan image retrieval berbasis teks dan konten, maka beberapa peneliti telah mencoba menggabungkan keduanya untuk pencarian citra. Terdapat banyak pendekatan telah digunakan untuk menggabungkan teks atau keyword dan teknik berbasis konten untuk pengambilan gambar. Metode sederhana yang digunakan untuk melakukan teks dan pengambilan konten berbasis secara terpisah dan menggabungkan hasil pengambilan telah diperkenalkan oleh para penulis (Besançon et al 2005; Jones et al 2005).

Wang (Wang et al 2006.) telah memperkenalkan pencarian citra berbasis anotasi, dalam sistem ini perlu keyword sebagai awal untuk mempercepat pencarian dengan memanfaatkan teknologi pencarian berbasis teks. Sayangnya keyword tidak selalu ada, maka jika keyword tidak ada kata sistem tidak akan efisien. Selanjutnya, sistem ini cenderung bias oleh karena tidak ada keyword yang digunakan sebagai awal dari pencarian suatu citra.

Untuk koefisien AC yang mempunyai nilai nol, maka fitur elemen dianggap sebagai nol sehingga satu vektor dari 64 elemen yang ada akan mewakili fitur tekstur blok pixel. Untuk mencirikan fitur tekstur dari seluruh citra, vektor seperti ini kemudian digunakan sebagai blok untuk digunakan sebagai indexing key. Jika misalnya terdapat sebanyak N blok koefisien DCT dalam suatu citra, maka indexing key dapat dicari dengan menggunakan persamaan berikut:

$$H = \sum_{i=1}^{64} C_i$$

Dimana $C_i = \frac{\sum_{k=1}^N (category)_k}{N}$ adalah kategori ke i diantara ke 64 koefisien DCT.

Banyak pendekatan telah digunakan untuk menggabungkan teks atau *keyword* dan teknik berbasis konten untuk pengambilan gambar. Metode sederhana yang digunakan dengan teks dan pengambilan konten berbasis secara terpisah dan menggabungkan hasil citra yang terpanggil telah diperkenalkan oleh beberapa ahli (Besançon et al 2005; Jones et al 2005).

Untuk mengukur kemiripan antara citra query dengan citra di dalam database dikerjakan dengan menghitung jarak Euclidean dengan menggunakan persamaan sebagai berikut.

$$d(Q, I) = \frac{\sum_i (Q_i - I_i)^2}{M}, \text{ dimana } Q \text{ dan } I \text{ merupakan citra}$$

query dan I adalah citra dalam, sedangkan M singkatan dari jumlah content descriptor.

10.5 Fitur Bentuk (Shape) dan Struktur

Struktur dari suatu objek dalam sebuah citra didefinisikan oleh spatial layout dari komponen-komponen penting yang dimiliki objek tersebut. Hubungan spatial (lokasi, kedekatan, overlapping, dan regresi) digunakan untuk membedakan citra yang ada dalam database. Sedangkan untuk menangkap hubungan spatial diantara objek digunakanlah teknik segmentasi berdasarkan batas minimum bidang epp atau berdasarkan titik pusat objek yang bersangkutan. Wang mengatakan (Wang dan Bovik, 2005) bahwa terdapat tiga representasi struktur yang sering digunakan yaitu: i). string 2D dimana symbol citra diwakili oleh string 2D berdasarkan proyeksi sumbu x dan sumbu y, ii). Perbedaan tipe diagram pohon (R-tree dan turunannya) yang memungkinkan suatu struktur data dinamis untuk epp dengan multi dimensi. Sementara itu

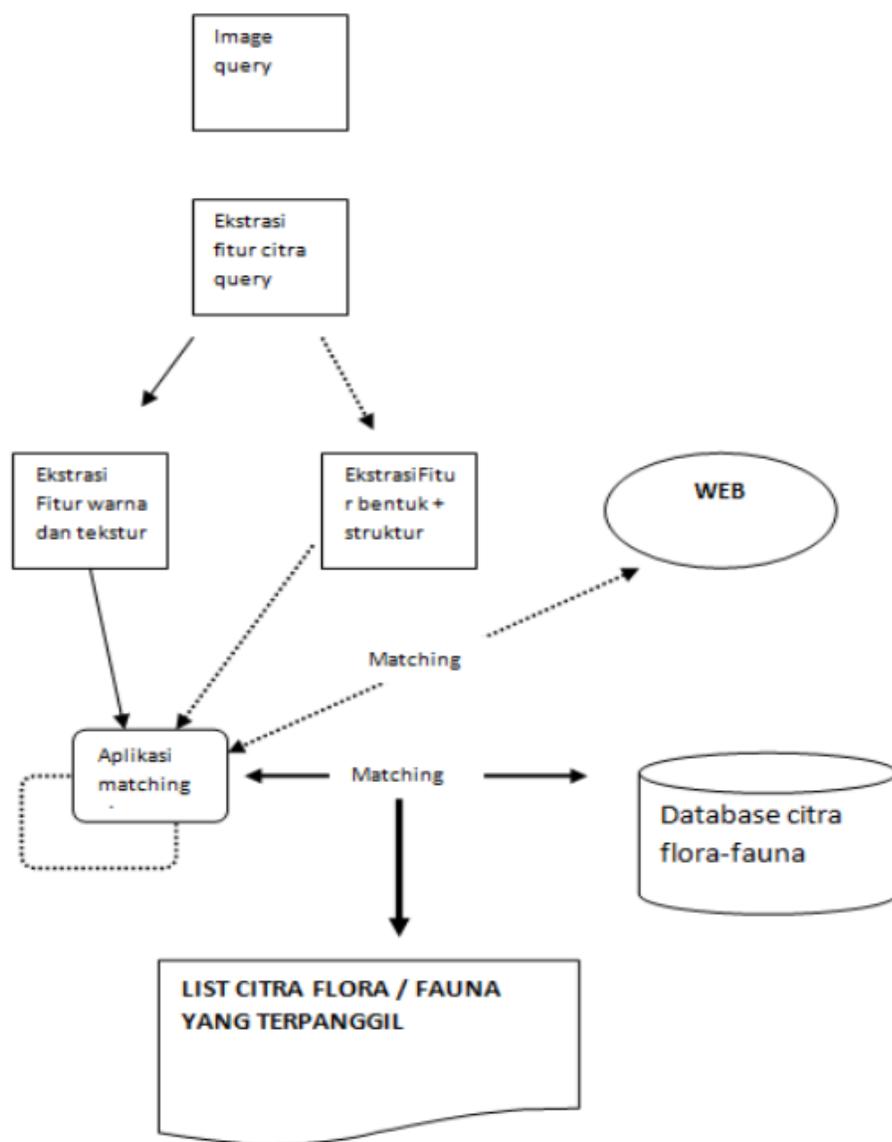
Sheikh (Sheikh *et.al.*, 2005) menggunakan Structure Similarity Measure (SSIM) untuk mengukur kemiripan berdasarkan struktur. Fitur bentuk titik-titik kontur yang telah dinormalisasi digunakan sebagai fitur bentuk asli dengan alasan untuk mengurangi kompleksitas. Translasi invariant dihitung dengan normalisasi koordinat kontur yang sesuai dengan pusat rotasinya. Rotasi invariant dihitung dengan memutar sumbu dari bentuk terhadap sumbu z (Limin, 2004). Sementara untuk mengukur kemiripan atau similarity dari bentuk citra kita digunakan rumus untuk menghitung Euclidean distance untuk fitur bentuk yang diperkenalkan oleh Xiaoxiang (Xiaoxiang, 2008), sebagai berikut:

$$\begin{aligned} d(U, V) &= \|V - U\| \\ &= \sqrt{\sum_{i=0}^{n-1} (v_i - u_i)^2} \end{aligned}$$

Dimana U adalah citra query dan V adalah citra dalam database, U_i merupakan fitur ke i dari citra query dan V_i adalah fitur ke i dari citra dalam database, sedangkan n adalah dimensi dari feature space. Untuk mengukur kemiripan berdasarkan struktur, dalam penelitian akan digunakan metoda structure similarity measure (SSIM). Kemiripan atau similarity diukur berdasarkan hubungan antara dua objek, yaitu fungsi struktur kemiripan dan fungsi content. Sementara itu kemiripan struktur atau structure similarity diukur berdasarkan struktur dari dua objek, (Jana et.al., 2009). Sheik (Sheikh et al. 2006) menyatakan bahwa untuk hitung similarity berbasis struktur dapat digunakan persamaan dibawah ini:

$$SSIM(x,y) = \left(\frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \cdot \left(\frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right) \cdot \left(\frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \right),$$

Dimana μ_x adalah mean (rata-rata) x dan μ_y adalah mean dari y . σ_x dan σ_y adalah standar x dan y , sedangkan σ_{xy} adalah kovarian dari x dan y , C_1, C_2 dan C_3 konstanta. SSIM adalah structure similarity measure atau ukuran untuk menentukan kemiripan berdasarkan fitur struktur.



Gambar 10.2. Diagram alur searching citra flora-fauna dengan *query by example*, searching citra berbasis warna dan tekstur, berbasis bentuk dan struktur

10.6 Fitur Warna dan Fitur Tekstur

1. Fitur Warna

Histogram warna untuk pencarian citra dikerjakan dengan mengitung jumlah koefisien DCT dari setiap warna. Dalam penelitian ini untuk memanggil citra dari suatu database dikerjakan dengan menggunakan histogram tahapan sebagai berikut: i). Pilih color space, ii). Kuatisasi color space, iii). Hitung histogram, iv). Hitung turunan fungsi jarak dari histogram.

Histogram suatu citra adalah fungsi dari intensitas warna, histogram warna dapat didefinisikan sebagai:

$$h_{A,B,C}(a,b,c) = N \cdot \text{Prob}(A = a, B = b, C = c)$$

Dimana A , B dan C merupakan komponen warna (R,G,B atau H,S,V) dan N jumlah Koefisien DCT dalam suatu citra. Untuk menghitung Euclidean distance antara histogram warna g dan histogram h , digunakan rumus berikut ini:

$$d^2(h,g) = \sum_A \sum_B \sum_C (h(a,b,c) - g(a,b,c))^2$$

2. Fitur Tekstur

Fitur tekstur adalah sifat yang ada dalam sebuah citra yang membedakan citra satu dengan yang lain berdasarkan keseragaman, kepadatan, kekasaran, keteraturan, intensitas, statistic, arah, dan randomness. Metoda statistika menganggap setiap tekstur sebagai vektor multi-dimensi. Metoda yang akan digunakan untuk ekstrasi fitur dalam penelitian kali ini adalah metoda yang sering dipakai para ahli diantara oleh Shapiro (Shapiro, 2001) dan Umbaugh (Umbaugh, 2006), yaitu : i). pixel neighborhood, dimana nilai intensitas dari pixel satu dibandingkan dengan nilai probalitas dari pixel yang ada disampingnya,ii). Matriks co-occurrence, adalah model yang telah dikerjakan sebelumnya pada konfigurasi tertentu grey-level dalam suatu tekstur, iii). Representasi tekstur tamura, metoda ini

didasarkan pada presepsi penglihatan manusia terhadap kekasaran, kontras, kecenderungan, regularitas, dan ketebalan, iv). Markov random field, adalah perhitungan statistik terhadap tekstur asli, v). Wavelent and Gabor transforms. Untuk pencarian atau matching citra query flora/fauna dengan citra yang ada dalam database dan internet akan kita gunakan metoda yang diperkenal oleh Castelli (Castelli & Bergman, 2006). Markov Random Field. Metode MRF ini dicirikan oleh struktur geometris dan besarnya kekuatan interaksi diantara piksel yang berdekatan, dalam metode ini tekstur dapat fungsi regresi sebagai berikut::

$$g(x, y) = \sum_{(m,n) \in N} a(m,n)g(x+m, y+n) + sw(x, y)$$

Dimana N adalah banyak parameter karakteristik dari setiap pixel atau koefisien DCT, (x,y) adalah probabilitas signal dalam setiap piksel Sementara untuk menghitung similarity dapat dihitung dengan menggunakan persamaan :

$$D(\mathbf{g}, \mathbf{q}) = \sum_{t=1}^T f_{\mathbf{g},t} \log \frac{f_{\mathbf{g},t}}{f_{\mathbf{q},t}}$$

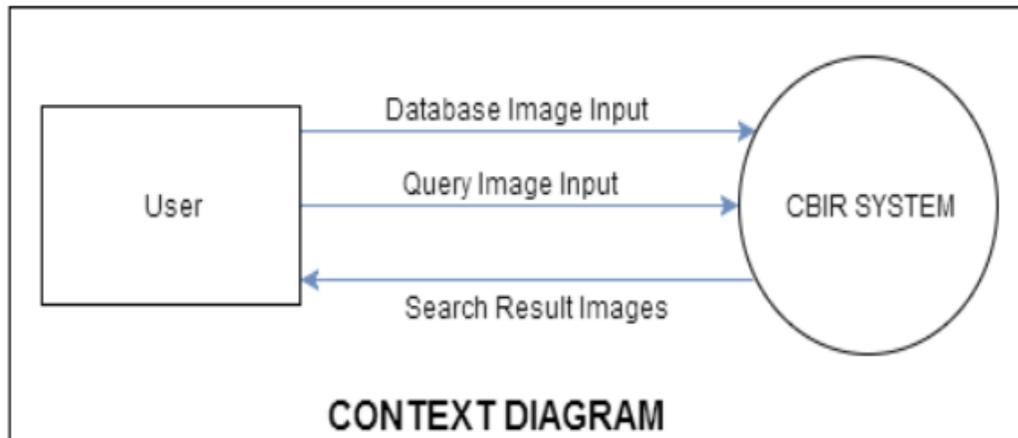
$D(\mathbf{g}, \mathbf{q})$ merupakan selisih antara dua fungsi sebaran, $\mathbf{f}_{\mathbf{g}} = (f_{\mathbf{g},t} : t=1, \dots, T)$ dan $\mathbf{f}_{\mathbf{q}} = (f_{\mathbf{q},t} : t=1, \dots, T)$, kemudian kemiripan didapat dari rata-rata $D(\mathbf{g}, \mathbf{q})$ dan $D(\mathbf{q}, \mathbf{d})$.

10.7 Desain CBIR untuk Search Engine

Analisis dilakukan dengan beberapa proses tahapan dalam pembuatan Prototipe Search Engine. Dengan melakukan desain sistem search engine Flora dan Fauna, untuk mengetahui kelemahan /kekurangan dari metode-metode tersebut. Kelemahan tersebut sebagai acuan dalam melakukan perbaikan dalam desain sistem yang akan dirancang.

1. Desain Diagram Konteks

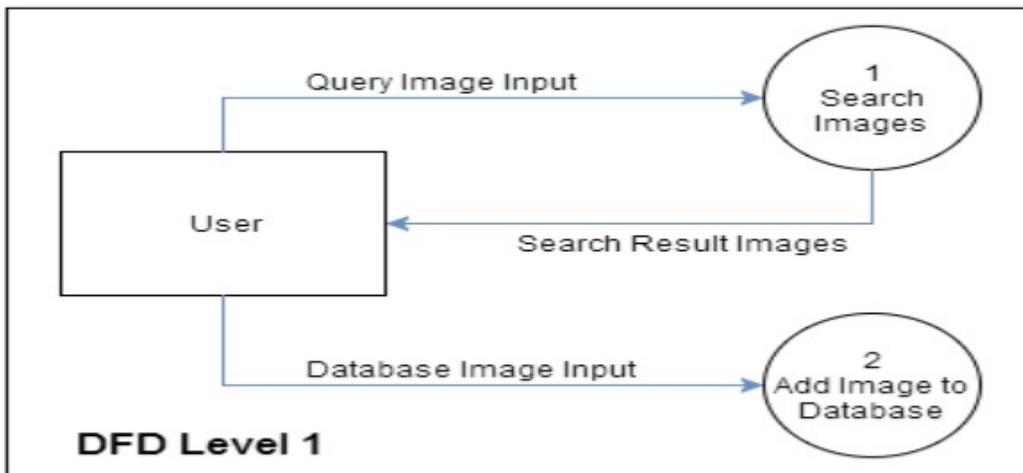
Sistem yang akan dirancang digambarkan dengan diagram konteks. Sebelum sistem CBIR rancang database dari image Flora dan Fauna dibuat dengan setiap image diberi indek. Image diinputkan ke dalam sistem CBIR dan queri image diinputkan, setelah itu sistem CBIR akan melakukan searching pada image, jika image yang dicari mirip dengan database image maka sistem CBIR akan memberikan laporan image yang sama.



Gambar10.3 Diagram Konteks

2. Desain DFD

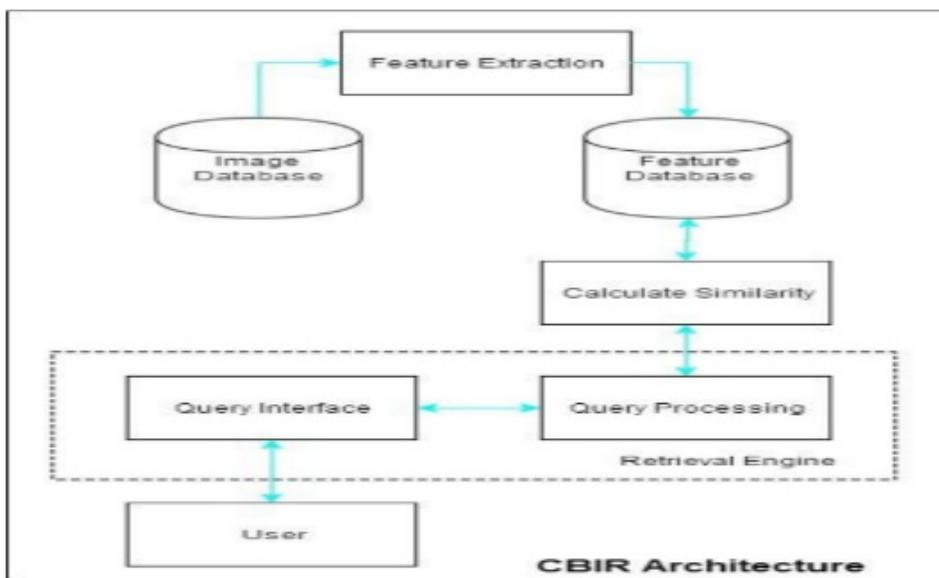
Desain DFD level 1 menggambarkan rincian dari tahapan pada diagram konteks. Sistem yang dibangun terdiri dari dua subsistem, yaitu: *search image* dan *add image to database*. *search image* berfungsi untuk memfasilitasi *user* dalam pencarian *image* tertentu, sedangkan *add image to database* berfungsi untuk melakukan penambahan *image* kedalam database, sebagaimana ditampilkan pada berikut ini:



Gambar 10.4 DFD Level 1

3. Arsitektur CBIR

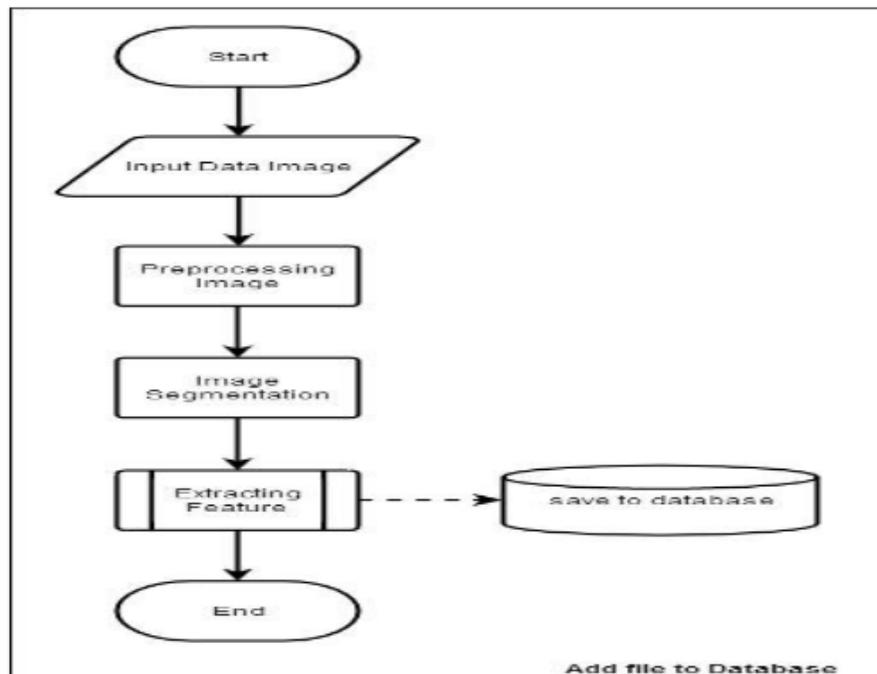
Adapun arsitektur dari CBIR sebagaimana ditampilkan pada gambar, sebagai berikut:



Gambar10.5 Arsitektur CBIR

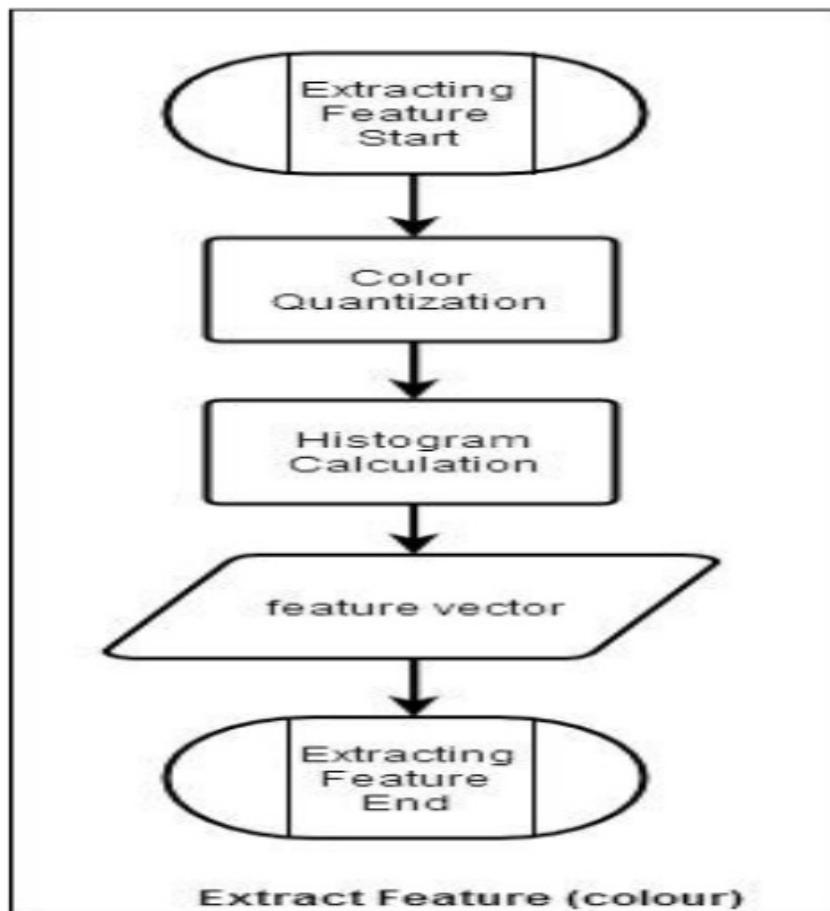
4. Flowchart Program

Proses penambahan *image* kedalam database melalui beberapa tahapan, yaitu: *input image*, *preprocessing*, *image segmentation* dan proses *extracing*. Tahapan tersebut digambarkan pada berikut ini:



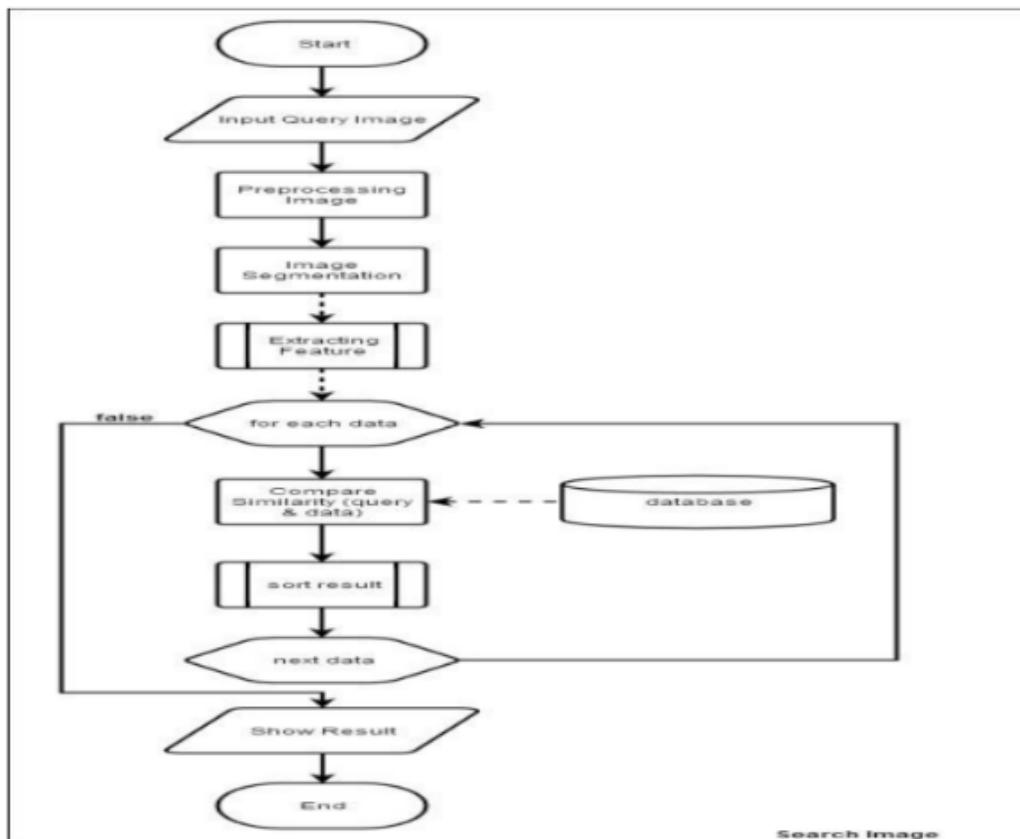
Gambar 10.6 Flowchart Program

Proses *Extract Feature* melalui beberapa tahapan, yaitu: *color quantization*, *histogram calculation*, dan pembentukan *feature vector*. Tahapan tersebut digambarkan berikut ini:



Gambar 10.7 Flowchart Program

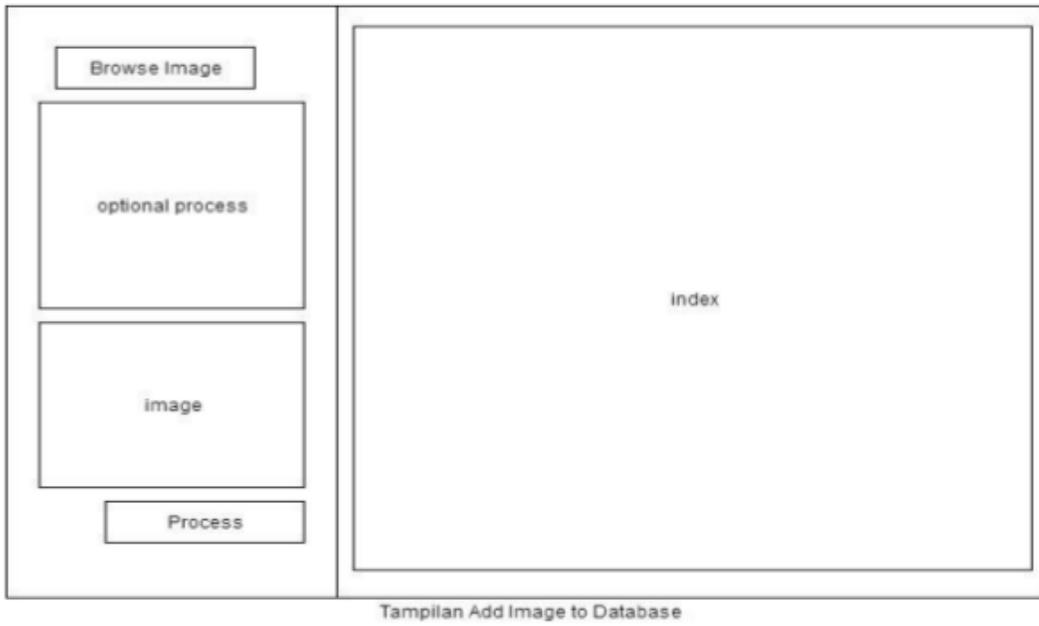
Proses Pencarian *image* kedalam database melalui beberapa tahapan, yaitu: input *image*, *preprocessing*, *image segmentation*, *extacing feature* dan proses *compare similarity*. Tahapan tersebut digambarkan berikut ini:



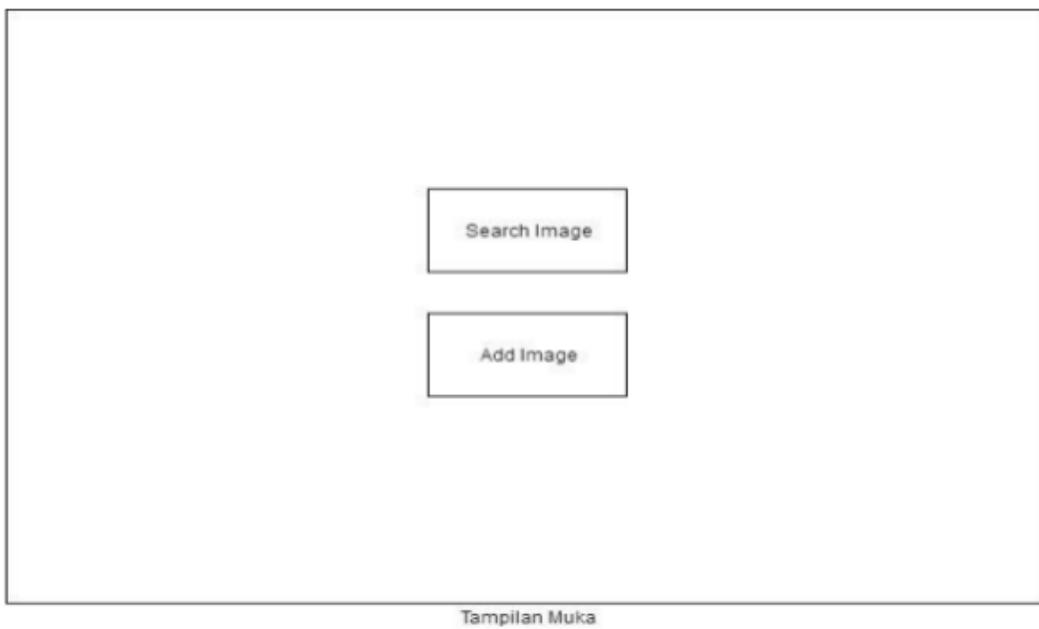
Gambar 10.8 Flowchart Program

5. Desain Input dan Output

Pada menu utama dari desain input dan output terdapat menu browse image untuk mencari gambar. Terdapat menu optional process, image. Terdapat button proses dan tampilan image yang telah di indeks. Seperti terlihat pada gambar 9.10, pada gambar 10.10 terdapat tombol pencarian gambar dan penambahan gambar.



Gambar 10.9 Tampilan Add Image to Database



Gambar 10.10 Tampilan Muka Program

10.8 Tampilan Program dan Pembahasan

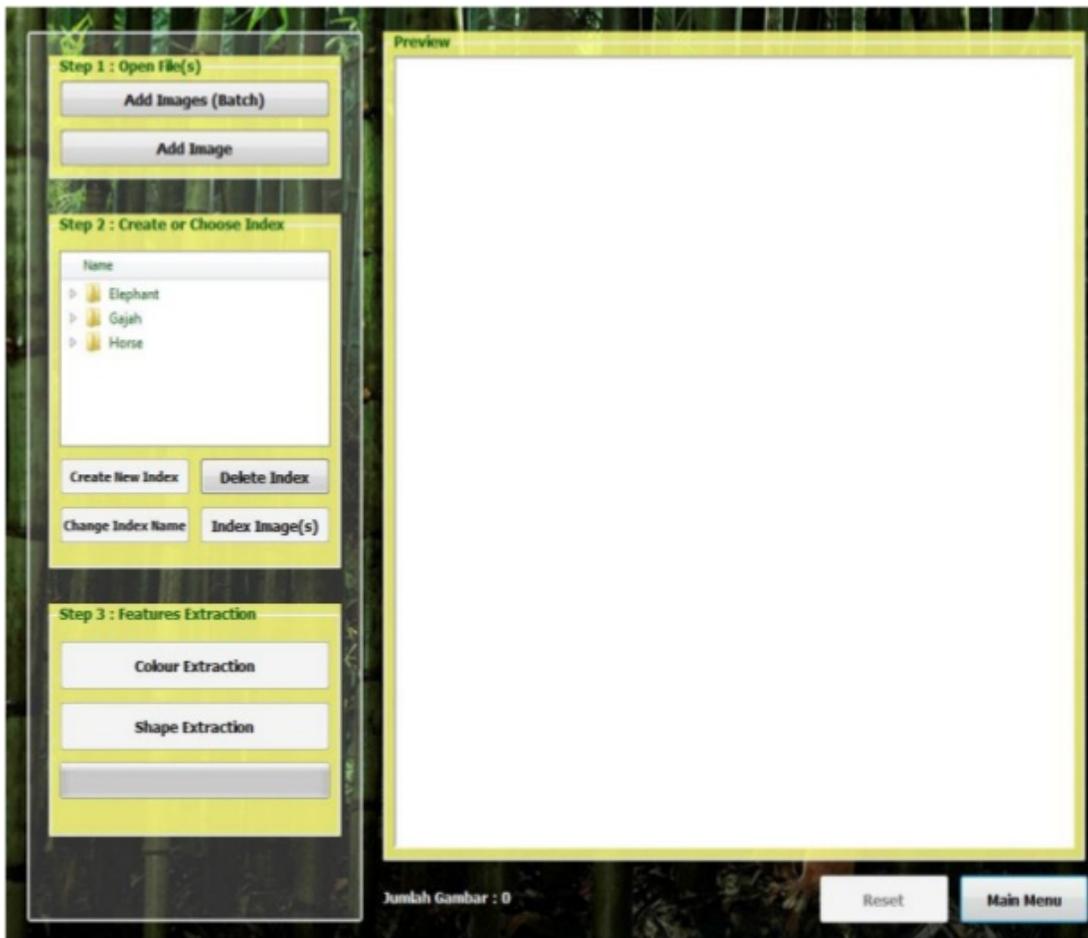
1. Tampilan Program

Pada main menu terdapat menu menambahkan gambar ke dalam database, mencari gambar dan keluar dari program, seperti pada gambar 10.11



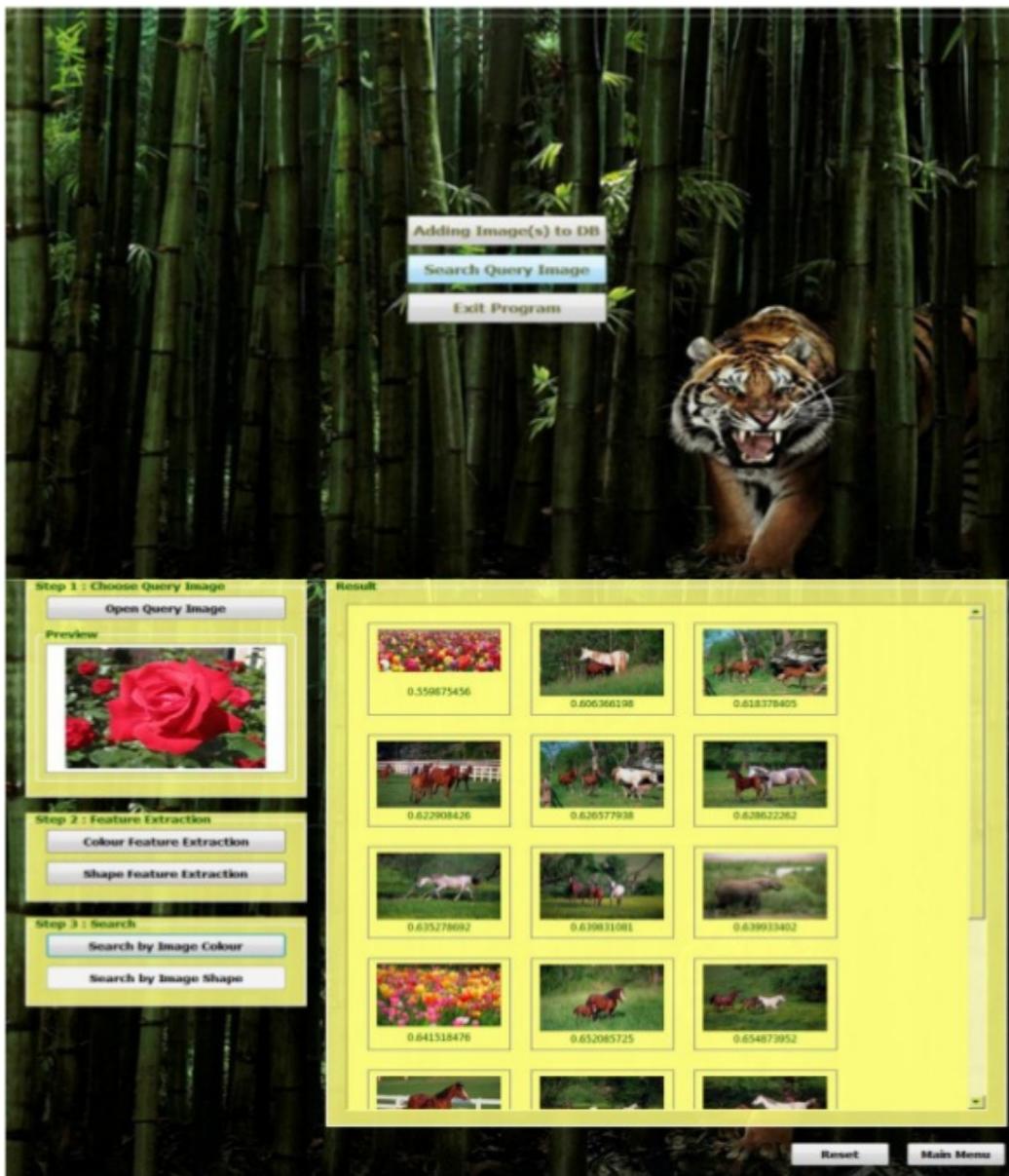
Gambar 10.11 Tampilan Menu Utama

Pada menu menambahkan gambar terdapat langkah pertama menu menambahkan gambar lebih dari satu gambar, menambahkan gambar dengan cara satu gambar. Langkah kedua terdapat menu membuat atau memilih indek. Membuat indek terbaru, mengganti nama indek, menghapus indek dan memberikan indek untuk setiap gambar. Langkah ketiga menu ekstrak kedalam warna dan kedalam bentuk.



Gambar 10.12 Menu Menambah Gambar

Pada menu pencarian langkah pertama terdapat menu open query image terdapat preview. Untuk langkah kedua setelah gambar ditampilkan lalu lakukan extraction untuk warna atau extraction untuk bentuk. Langkah ketiga menu search, dengan menu search by image colour dan search by image shape, pilih sesuai dengan pencarian warna atau bentuk. Setelah dilakukan pencarian dengan warna/bentuk maka gambar 10.13 akan ditampilkan.



Gambar 10.13 Pencarian Gambar Berdasarkan Warna dan Bentuk

2. Pembahasan

Hasil dari pencarian warna, dengan ruang warna (colour space) HSV (Hue, Saturation dan value) nilai yang terdapat dari hasil pencarian dari gambar merupakan nilai antara perbandingan Histogram gambar yang diiputkan dengan gambar yang ada di database.

Untuk bentuk, nilai yang terdapat dari hasil pencarian kemiripan gambar merupakan perbandingan nilai contours.

Penerapan metode *Content Based Image Retrieval* dalam pembuatan *search engine like* berbasis citra untuk fauna ini dilakukan melalui tahapan: desain sistem, flowchart program dan coding. Fasilitas yang disediakan dalam *prototype* ini terdiri dari menu penambahan *image* dan pencarian *image*. Menu penambahan *image* ke dalam data base untuk bertujuan untuk memperkaya *content image* di dalam database, sedangkan menu pencarian *image* untuk mencari *image* yang memiliki kesamaan (*similarity*) dengan *image* yang dicari (diinputkan).

Tugas Mandiri

1. Jelaskan pengertian dari *Web crawler*.
2. Buatlah Algoritma untuk *search engine* dengan menerapkan metode CBIR
3. Jelaskan perbedaan dari fitur warna dengan fitur bentuk.

DIGITAL WATERMARKING

69

Dewasa ini permasalahan pelanggaran kepemilikan hak cipta tak akan pernah selesai dan akan menjadi lebih rumit ketika kepemilikan citra digital dipertanyakan pemilik sebenarnya, pemalsuan tanda tangan, pemalsuan surat-surat berharga, dan pemalsuan karya-karya seni yang lain. Oleh karena itu perlindungan hak cipta dari suatu karya seni dan surat-surat berharga sangat diperlukan. Untuk mengatasi masalah ini maka perlu dibuat suatu teknik sekaligus aplikasi proteksi hak cipta yang mampu melindungi hak cipta seseorang.

Di bidang pengolahan citra atau image processing pemalsuan atau plagiarism dapat ditanggulangi dengan banyak teknik diantara teknik *watermarking*, *image blending*, dan teknik *content based image retrieval*. Teknik water marking dikerjakan dengan cara menyisipkan kedalam data citra tersebut. Watermark dapat dianggap sebagai sidik digital (*digital signature*) dari pemilik sah atas produk multimedia tersebut. Dengan demikian, watermark yang disisipkan menjadi hak cipta dari pemiliknya. Pemberian signature dengan teknik watermarking ini dilakukan sedemikian sehingga informasi yang disisipkan tidak merusak data digital yang dilindungi. Sehingga orang yang membuka produk multimedia yang sudah disisipkan watermark tidak menyadari kalau didalam data multimedia tersebut terkandung label atau *signature* kepemilikan pembuatnya.

Menurut Hassanien (2006) watermarking pada citra digital berdasarkan kenampakannya dibedakan menjadi dua yaitu *visible watermark* dan *invisible watermark*. Visible Watermark adalah watermark terlihat dapat dilihat, biasanya berupa teks atau logo semi transparan yang ditambahkan atau digabungkan pada citra dan berfungsi sebagai sandi atau *password*. Sedangkan *invisible watermark* adalah watermark yang tak terlihat biasanya tersembunyi. Sementara itu menurut Saphiro dan Stockman (2005) mengemukakan bahwa metode image blending sering digunakan dalam implementasi teknik watermarking. Image blending adalah cara untuk menggabungkan dua citra digital, dimana citra sebagai objek dan citra lain sebagai background untuk menghasilkan citra baru. Untuk pengimplementasian invisible watermark Utami(2009) menyatakan dapat menggunakan metode image blending dapat dikerjakan dengan menghitung *Least Significant Bit (LSB)*, yang dapat digunakan untuk menyisipkan "sandi" berupa citra sebagai watermark kedalam citra digital yang lain. "sandi" yang disisipkan tidak terlihat oleh mata dan tidak merubah citra aslinya .

11.1 Citra Digital

Menurut Gonsale *et.al.*, (2007) citra (*image*) adalah gambar pada bidang dwimatra atau dua dimensi. Citra juga dapat diartikan sebagai kumpulan titik-titik dengan intensitas warna tertentu yang membentuk suatu kesatuan dan mempunyai pengertian artistik. Citra sebagai salah satu komponen multimedia yang memegang peranan sangat penting sebagai salah satu bentuk informasi visual. Lebih lanjut Gonsale *et.al.*, (2007) menyatakan citra sebagai keluaran dari suatu sistem perekam data dapat bersifat analog, berupasinyal-sinyal video seperti gambar pada monitor televisi atau bersifat digital yang dapat langsung disimpan pada suatu media magnetik. Citra ada dua macam yaitu citra kontinu dan citra diskrit. Citra Kontinu dihasilkan dari sistem optik yang menerimasinyal analog, contohnya mata manusia, kamera analog.

Suatu Citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitude f di titik koordinat (x,y) dinamakan intensitas atau tingkat

keabuan dari citra pada titik tersebut. Apabila nilai x dan y , dan nilai Amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut citra digital. Nilai Pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut sebagai *picture element* atau, *image elements* atau *pels* atau *pixels*, Russ (2010).

11.2 Citra Warna

Russ (2011) menjelaskan bahwa citra warna adalah citra dengan sistem grafik yang memiliki satu set nilai tersusun (*aset of ordered values*) yang menyatakan berbagai tingkat warna. Citra warna bukanlah seperti citra *grayscale*. Dimana setiap set nilai tersusun mewakili satu 'scale' arna atau 'hue'. Lebih lanjut Russ (2011) mengemukakan bahwa citra yang memiliki warna *grayscale* cenderung kurang menarik untuk dilihat dibandingkan dengan citra berwarna., karena kamera pada jaman dahulu hanya mampu menghasilkan citra dengan format warna *greyscale*, sehingga hasil citra tersebut menjadi kurang menarik untuk dilihat. Padahal, banyak citra zaman dahulu memiliki nilai sejarah yang cukup tinggi yang semestinya disampaikan dari generasi ke generasi. Sementara itu Gonzales (2005) menyatakan bahwa sistem yang dipakai untuk mewakili warna yaitu sistem RGB (*Red, Green, Blue*). Sistem RGB adalah sistem penggabungan antara warna-warna primer (*additive primary colours*) yaitu merah (*Red*), hijau (*Green*) dan biru (*Blue*) untuk memperoleh warna tertentu. Misalnya warna putih diperoleh dari hasil gabungan warna merah = 255, hijau = 255, dan biru = 255.

11.3 Format File Gambar

Berdasarkan pendapat Russ (2011) pada umumnya *file* gambar digunakan untuk menyimpan gambar yang ditampilkan dilayar ke dalam suatu media penyimpanan data. Untuk menyimpan sebuah *file* gambar ini digunakan salah satu format *file*. Ada banyak format *file* gambar yang dapat digunakan untuk menyimpan file gambar, salah satunya adalah BMP. Format file ini mampu menyimpan gambar dalam mode warna RGB, *Grayscale*, *Indexed Color*, dan *Bitmap*. Kelemahan format file ini adalah tidak mampu menyimpan alpha channel serta ada kendala dalam pertukaran platform. Kelebihan tipe

file BMP adalah dapat dibuka oleh hampir semua program pengolah gambar. Baik file BMP yang terkompresi maupun tidak terkompresi, file BMP memiliki ukuran yang jauh lebih besar daripada tipe-tipe yang lain.

Format citra lain yang cukup populer adalah citra dengan format JPG/JPEG (*Joint Photographic Experts Group*). Menurut Wallace (1991) *Joint Photographic Experts (JPEG)* di rancang untuk kompresi beberapa *full-color* atau *gray-scale* dari suatu gambar yang asli, seperti pemandangan asli di dunia ini. JPEGs bekerja dengan baik pada *continous tone images* seperti *photograph* tetapi tidak terlalu bagus pada ketajaman gambar dan seni pewarnaan seperti penulisan, kartun yang sederhana atau gambar yang menggunakan banyak garis. JPEG sudah mendukung untuk 24-bit *color depth* atau sama dengan 16,7 juta warna ($2^{24} = 16.777.216$ warna).

11.4 Operasi Berbasis Bingkai

Menurut Petro (2010), operasi berbasis bingkai adalah operasi yang melibatkan dua buah citra atau lebih dan menghasilkan sebuah citra keluaran yang merupakan hasil dari operasi matematis. Operasi ini dilakukan titik per titik dengan lokasi yang bersesuaian pada citra-citra masukan tersebut. Untuk operasi berbasis bingkai antara 2 buah citra, misalnya antara citra A dan citra B yang menghasilkan citra C, secara umum dapat dituliskan sebagai berikut :

$$C(x, y) = A(x, y) \text{ op } B(x, y)$$

Dimana *op* adalah operator yang akan diterapkan terhadap kedua citra tersebut. Berdasarkan operator yang dipakai, operasi berbasis bingkai meliputi operator aritmatika dan operasi logika.

11.5 Operator Aritmatik

Berdasarkan pernyataan Zhao et.al., (2005) operator aritmatik biasanya diterapkan terhadap citra keabuan dan warna. Berdasarkan operator aritmatik yang umum digunakan, terdapat empat operasi aritmatik yang dapat dirumuskan sebagai berikut (untuk dua citra), yaitu:

$$C(x, y) = A(x, y) + B(x, y)$$

$$C(x, y) = A(x, y) - B(x, y)$$

$$C(x, y) = A(x, y) * B(x, y)$$

$$C(x, y) = A(x, y) / B(x, y)$$

Operasi yang paling banyak digunakan pada berbagai aplikasi adalah penjumlahan dan pengurangan, sedangkan operasi perkalian dan pembagian hanya digunakan pada kasus-kasus tertentu. Contoh aplikasi operasi aritmatik adalah pengurangan derau (noise), penggabungan citra (*image blending*), perbaikan ketidaklinieran pencahayaan, deteksi gerakan, pembuatan citra diferensi untuk tepi, serta *digital angiography*.

11.6 Operator Aritmatik

Menurut Xuelong (2003) *image blending* atau penggabungan citra dilakukan dengan cara menimpakan (*superimpose*) sebuah citra dengan citra yang lain. Operasi yang dilakukan sesungguhnya adalah dengan melakukan operasi penjumlahan terhadap suatu citra dengan bobot masing-masing. Bobot menggabungkan 2 citra ini sering disebut sebagai koefisien atau nilai alpha. Rumus penggabungan dua buah citra adalah:

$$C(x, y) = w_A \cdot A(x, y) + w_B \cdot B(x, y)$$

Dengan w_A dan w_B adalah bobot untuk citra A dan B. Biasanya nilai total dari untuk sebuah lokasi piksel adalah 1, sehingga :

$$w_B = 1 - w_A$$

58

Digital watermarking adalah teknik untuk menyisipkan informasi tertentu ke dalam sebuah data dengan suatu cara tertentu sehingga watermark itu sulit dirusak dan dihapus, hal ini didasari oleh Ajay Goel (2010). Secara garis besar, watermark terbagi menjadi dua tipe, yaitu *visible watermark* dan *invisible watermark*, berikut penjelasan kedua tipe tersebut menurut Jemi (2007).

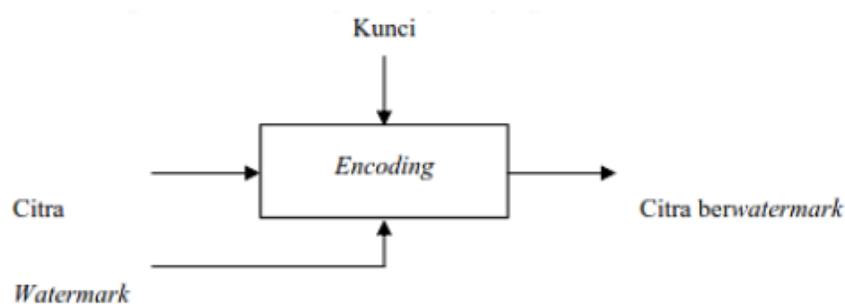
- *Visible watermark*, Watermark jenis ini dapat terlihat oleh indera manusia. *Visible watermark* bersifat sangat *robust* karena

keberadaan watermark dapat dikenali dengan mudah dan biasanya sangat sulit untuk dihapus. Watermark yang disisipkan dapat bersifat *solid* ataupun *semitransparan*, dan untuk memindahkannya membutuhkan *cropping* yang signifikan.

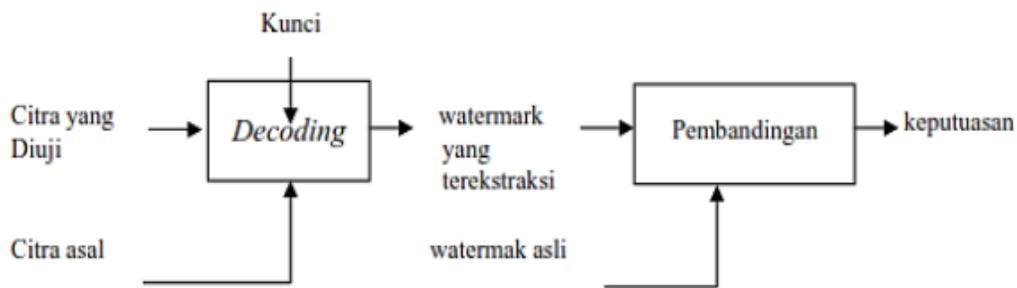
- *Invisible watermark*, Watermark jenis ini tidak dapat terlihat oleh indera manusia, tetapi dapat diekstraksi dengan metode komputasional tertentu. Tujuan dari *invisible watermark* adalah untuk memverifikasi kepemilikan atau memverifikasi integritas dari suatu citra atau sejumlah informasi. Biasanya pada saat ekstraksi *invisible watermarking* dibutuhkan sebuah password yang disebut watermark key.

a. Proses Watermarking

Dugelay (2006) menyatakan bahwa proses penyisipan *watermark* ke dalam citra disebut *encoding*. *Encoding* dapat disertai dengan pemasukan kunci atau tidak memerlukan kunci. Kunci diperlukan agar *watermark* hanya dapat diekstraksi oleh pihak yang sah. Kunci juga dimaksudkan untuk mencegah *watermark* dihapus oleh pihak yang tidak berhak.



Gambar 11.1 Proses Watermark pada Citra Digital



Gambar 11.2 Proses *Verification* pada *Citra Digital*

Verifikasi watermark dilakukan untuk membuktikan status kepemilikan citra digital yang disengketakan. Verifikasi watermark terdiri dari dua sub-proses, yaitu ekstraksi watermark dan perbandingan. Proses ekstraksi watermark disebut juga decoding, bertujuan mengungkap watermark dari dalam citra. Decoding dapat mengikutsertakan citra asal atau tidak sama sekali, tujuannya adalah untuk meningkatkan kinerja yang lebih baik. Proses perbandingan bertujuan untuk membandingkan watermark yang diungkap dengan watermark asli dan memberi keputusan tentang watermark tersebut.

b. Metode *Image Blending*

Penggunaan metode *image blend* diterapkan pada teknik *visible watermark*. Penyisipan *watermark* dengan menggunakan metode ini dilakukan dengan cara mencampurkan nilai pixel suatu citra dengan citra lain. Operasi yang dilakukan adalah dengan melakukan penjumlahan terhadap nilai pixel suatu citra dengan citra lain dengan bobot masing-masing, dengan aturan total nilai dari penjumlahan bobot suatu pixel adalah 1 Rankov(2005). Sementara itu metode LSB pada penelitian ini digunakan sebagai metode proses penyisipan dan ekstraksi. Penerapan metode LSB pada proses Penyisipan dan Ekstraksi akan dijelaskan sebagai berikut :

- Proses Penyisipan, Proses penyisipan watermark dengan metode LSB dilakukan dengan cara mengganti bit ke 24 pada representasi biner file gambar dengan representasi biner *image watermark* .dengan demikian setiap file gambar BMP24

bit dapat disisipkan 3 bit dari *image watermark*. Terlihat pada bit 8, 16 dan 24 diganti dengan representasi representasi biner watermark, dan hanya 7 bit rendah yang berubah (cetak tebal), untuk penglihatan mata manusia sangatlah mustahil untuk dapat membedakan warna pada file gambar yang sudah disisipkan watermark jika dibandingkan dengan file gambar asli sebelum disisipi dengan watermark.

- Proses Ekstraksi, proses ekstraksi watermark adalah proses membaca watermark pada

citra yang telah terwatermark dengan mengambil kembali informasi dari setiap bit rendah

dari citra ter-watermark dan mengkonversinya kembali menjadi citra gambar.

11.7 Least Significant Bit Hiding (LSB)

31

Onkar (2007) berpendapat metode LSB (*Least Significant Bit*) merupakan salah satu metodewatermarking yang bekerja dalam mode warna RGB (*Red, Green, Blue*). Metode ini bekerja dengan cara menyisipkan informasi pada bit-bit paling kanan dari setiap elemen RGB. Perubahan bit paling kanan hanya menimbulkan perubahan nilai RGB sebesar 1 dari 256 warna yang ada. Selanjutnya Onkar (2007) menambahkan bahwa metode LBS secara langsung memanipulasi nilai intensitas dari sejumlah pixel. Penyembunyian data dilakukan dengan mengganti bit-bit data di dalam segmen citra dengan bit-bit data rahasia. Pada susunan di dalam sebuah byte (1 byte = 8 bit), ada bit yang kurang berarti disebut bit LSB. Bit LSB inilah yang akan dimanipulasi untuk penyisipan *watermark*.

Operasi Pergeseran Citra

Operasi pergeseran citra pada perancan ini digunakan dalam melakukan peletakan *watermark* atau citra tanda agar dapat sesuai dengan apa yang dikehendaki. Berikut adalah persamaan yang digunakan pada pergeseran di teks *watermark*:

$$Y_{\text{step}} = Y_{\text{old}} + S_y;$$

$$X_{\text{step}} = X_{\text{old}} + S_x;$$

Dengan menggunakan persamaan diatas akan didapatkan sebuah koordinat baru peletakan nilai pixel suatu citra. Dalam penelitian ini persamaan diatas digunakan sebagai pengatur peletakan posisi *watermark* pada citra. Dan satuan yang digunakan untuk S_y dan S_x adalah pixel. Jika S_y bernilai 5 maka citra akan bergeser 5 pixel ke atas, jika S_x bernilai 3 maka citra akan bergeser 3 pixel ke kanan, sedangkan jika S_x dan S_y bernilai negative arah pergeserannya akan berlawanan. Algoritma pemutaran citra dengan menggunakan *transformasi affine* ini digunakan sebagai pemutar citra tanda pemutaran dilakukan dengan menggunakan persamaan berikut.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\text{rad}) & \sin(\text{rad}) \\ -\sin(\text{rad}) & \cos(\text{rad}) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Dengan menggunakan persamaaan diatas suatu citra akan dapat dirotasi sesuai dengan radian. Sehingga jika ingin memutar citra 45° maka nilai radian adalah $\pi/4$ radian. Algoritma penskalaan citra, suatu citra dapat diperbesar ataupun diperkecil dengan membuat setiap piksel menjadi beberapa piksel. Pada penelitian ini penskalaan citra dilakukan dengan menggunakan rasio. Rasio didapatkan dari perbandingan tinggi dan lebar dari citra dengan tempat menampilkan citra (*viewer*). Berikut persamaan rasio tinggi dan lebar :

$$S_h = \text{Viewer}_h / H$$

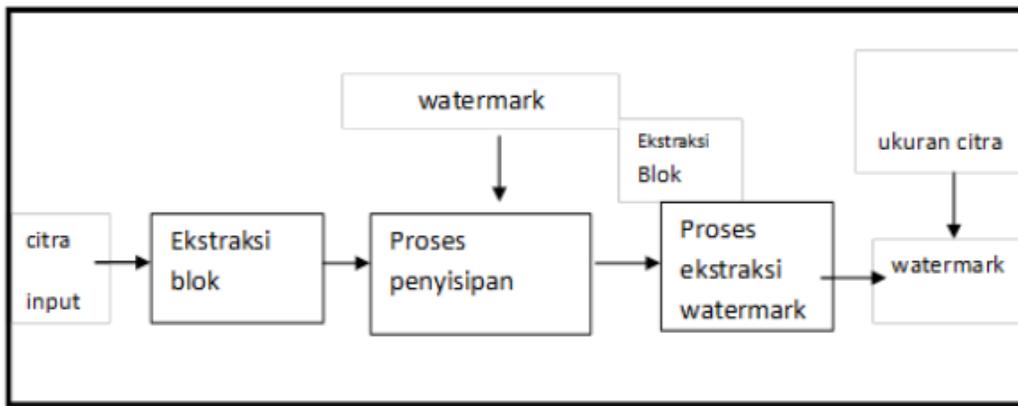
$$S_w = \text{Viewer}_w / W$$

Rasio tinggi S_h didapatkan dari pembagian tinggi *viewer* dengan tinggi asli citra, sedangkan rasio lebar didapatkan dari pembagian lebar *viewer* dengan lebar asli citra.

49

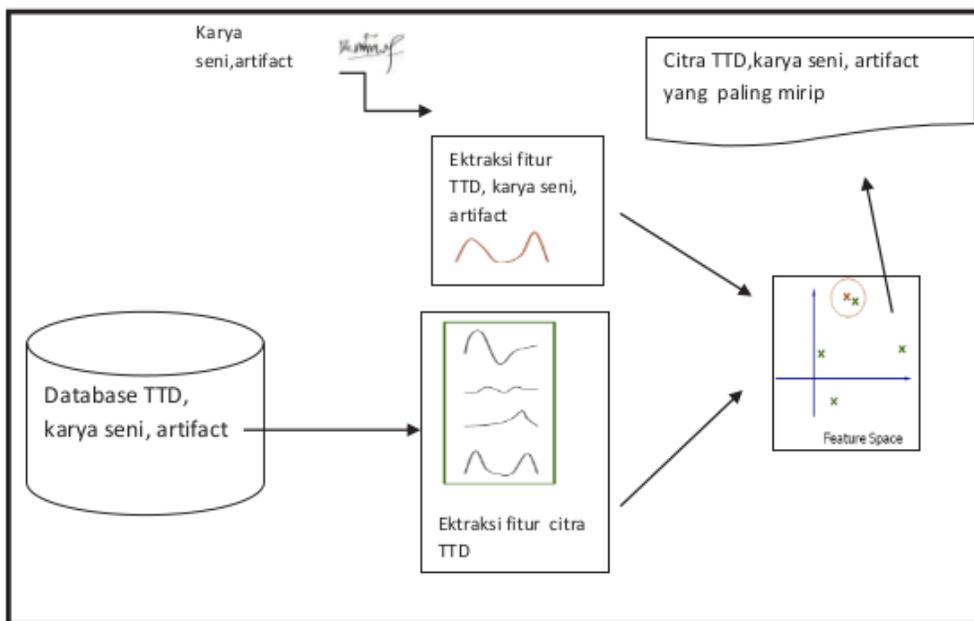
11.8 Metode Watermarking dan Content Based Image Retrieval

Metode *watermarking* dan *content based image retrieval* untuk mendeteksi dan verifikasi keaslian tanda tangan dan artifact lainnya. Metode-metode yang kami kerjakan dapat dijelaskan pada gambar 3, 4 dan 5. Untuk verifikasi tanda tangan dengan menggunakan teknik *watermarking* sedangkan untuk pada verifikasi karya seni dan artifact lainnya dengan menggunakan *watermarking* dan CBIR.

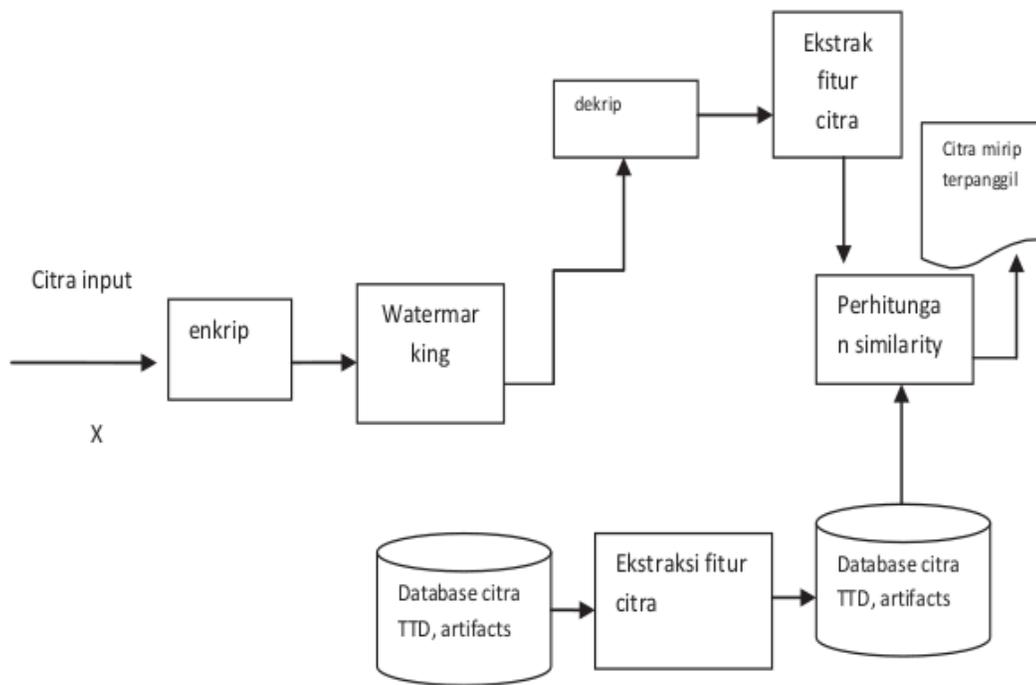


Gambar 11.3 Proses penyisipan dan ekstraksi watermarking

Sedangkan proses matching keaslian citra artifact /tanda tangan dapat dijelaskan pada gambar dibawah ini:



Gambar11.4 Sistem verifikasi TTD, karya seni dan artifact dengan teknik CBIR



Gambar 11.5 Verifikasi keaslian TTD dan artifact menggunakan watermarking dan Content Based Image Retrieval.

11.9 Aplikasi Watermarking

Analisis dilakukan pada beberapa proses watermarking yang telah ada sebelumnya, untuk mengetahui kekurangan/kelemahan dari metode-metode tersebut. Kelemahan tersebut akan dijadikan sebagai acuan dalam melakukan perbaikan dalam mendesain algoritma proses watermarking yang lebih baik. Desain dihasilkan sebagai usulan perbaikan sebagai upaya dalam melakukan proses penyempurnaan. Desain yang dibuat meliputi desain system dan desain logika program/ algoritma yang menggambarkan urutan jalannya proses. Banyak literature telah dipekenalkan oleh beberapa peneliti, kami mencoba membangun algoritma yang kami desain untuk melakukan penyempurnaan dari algoritma-algoritma sebelumnya, Secara umum algoritma aplikasi dapat dijelaskan sebagai berikut:

\\ mengembunyikan tulisan/text pada citra

1. *Input citra asli*
2. *Convert citra ke vector*
3. *Conver vektor to binary*
4. *Convert text to vector*
If text hided then

mat = 1 x n matrix andhide into ASCII form

Else mat= m x n matix

Extract RGB component

End

5. *Go to 1*

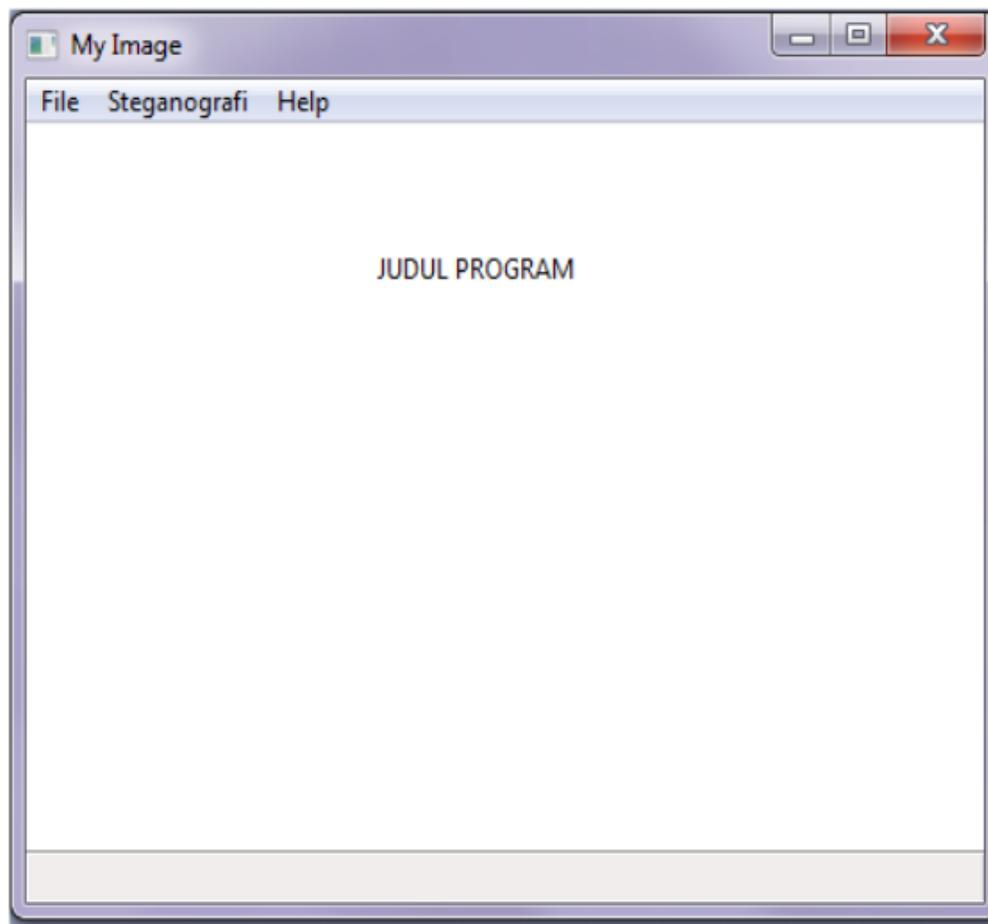
\\ mengembalikan tulisan/text yang disembunyikan pada citra

1. *Convert hided image into vector*
2. *Conver vector to binary*
3. *Take the most righth bit from each vector n*
4. *Make number of power of 8 from step 3*
5. *If text then conver decimal to text value*
6. *If image constract RGB each pixel*

Aplikasi yang dihasilkan sangat berguna untuk menyembuyikan citra logo kedalam dokumen untuk melindungi keaslian dari dokumen tersebut. Beberapa output dari hasil pengujian terhadap aplikasi yang dihasilkan dapat dilihat pada gambar berikut ini:

a. Tampilan Program

Pada menu awal program ketika dijalankan adalah terdiri dari 3 menu yaitu file, steganografi dan help. Adapun tampilan menu awal program tersebut ditampilkan pada gambar 3 berikut ini :

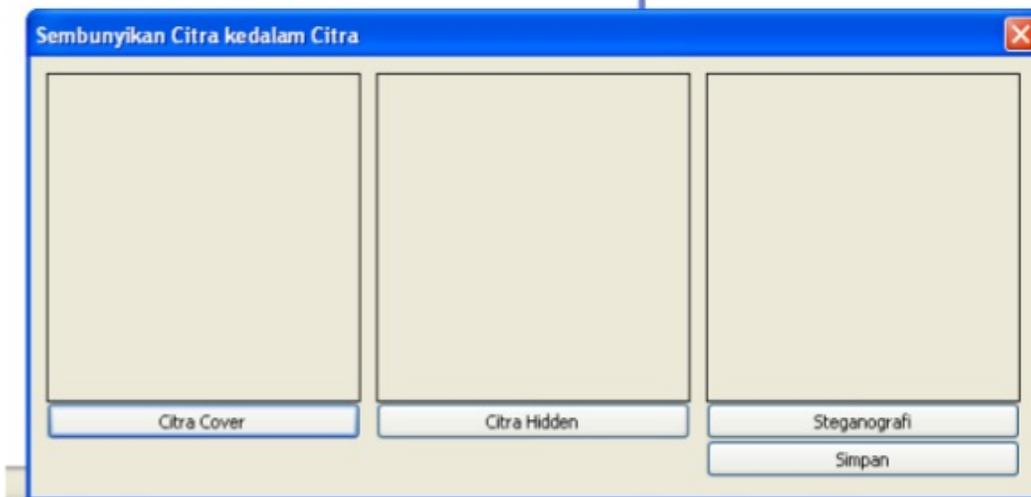


Gambar 11.6Tampilan Menu Awal

Menu steganografi merupakan fasilitas proses watermarking. Dilakukan dengan menentukan citra coverberformat png, jpg dan bmp, lalu pilih citra hiden/yangakan disembunyikan. Kemudian tekan tombol steganografi, maka akan muncul citra baru yangmengandung pesan tersembunyi. Kemudian citra tersebutdapat disimpan kedalam format .png dan .bmp. Pada menu

stenografi tersebut terdapat 3 tahapan seperti ditunjukkanpada gambar berikut ini :

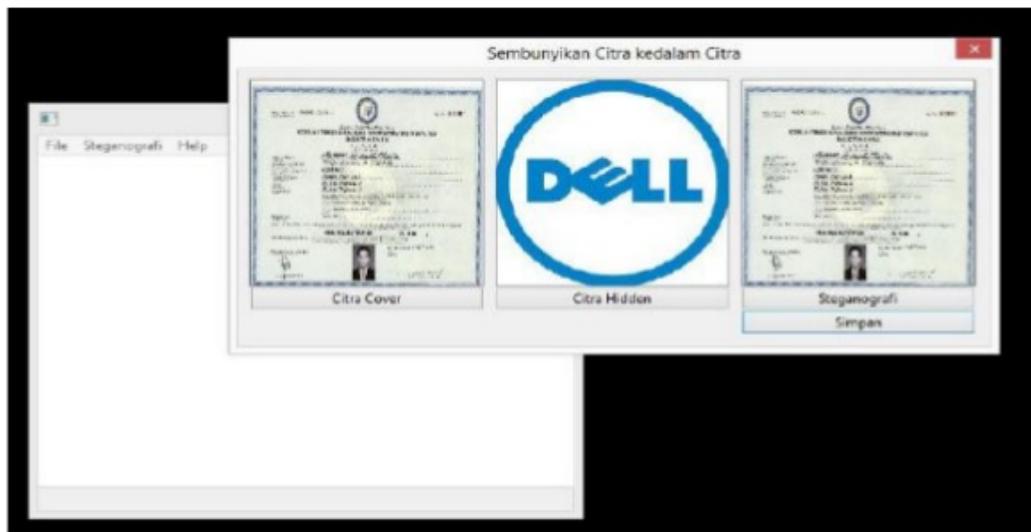
b. Tampilan Menu Steganografi



Gambar 11.7 Menu Steganografi

c. Output Penyisipan Citra Kedalam Dokumen

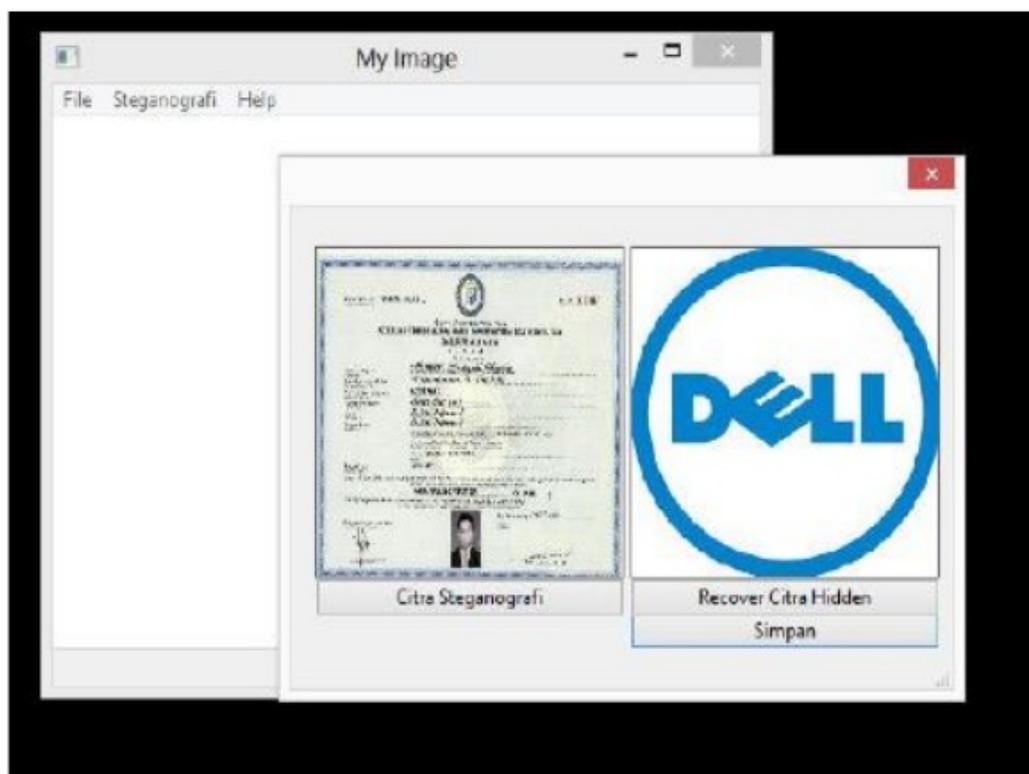
Tampilan penyisipan watermark berupa citra kedalam dokumen ditampilkan pada gambar berikut ini



Gambar 11.8 Snapshot menyisipan/mensembunyikan logokedalam dokumen

d. Output Hasil Ekstraksi Watermark Citra

Watermark berupa citra yang telah disisipkan sebelumnya, dapat diekstrak kembali. Hasil ekstraksi Citra dari dalam dokumen ditampilkan pada gambar berikut ini:



Gambar 11.9 Snapshot memunculkan kembali logo yang disembunyikan dalam dokumen

Proses Ekstraksi dilakukan dengan memilih citra steganografi (yang sebelumnya telah diinsertkan citra watermark), Dokumen tersebut berformat png dan bmp tekan tombol recover citra hidden, maka selanjutnya akan muncul citra yang disembunyikan. Kemudian citra tersebut dapat disimpan kedalam format .jpg, .png dan .bmp.

Fasilitas penyisipan dan ekstraksi dapat dimanfaatkan untuk mendeteksi watermark dalam suatu dokumen. Penandaan dapat berupa gambar, atau logo. Citra selanjutnya akan disembunyikan kedalam bentuk vector. Jika yang disembunyikan adalah text maka bentuk matriknya $1 \times n\text{-text}$. Citra memiliki bentuk matrik $m\text{-citra} \times n\text{-citra}$. Diketahui $m\text{-citra}$ adalah baris citra dan $n\text{-citra}$ adalah kolom

citra. Selanjutnya ambil nilai intensitas pixel nya berupa merah, hijau dan biru. Selanjutnya akan diubah nilai vector yang disembunyikan kedalam binary citra yang disembunyikan di masukkan kedalam binary citra cover secara berurutan dengan cara melihat nilai bit terkecil dari citra cover dan nilai biner ke n dari citra yang disembunyikan. Dalam penelitian ini adalah document text dan gambar yang kemudian discan sehingga menjadi

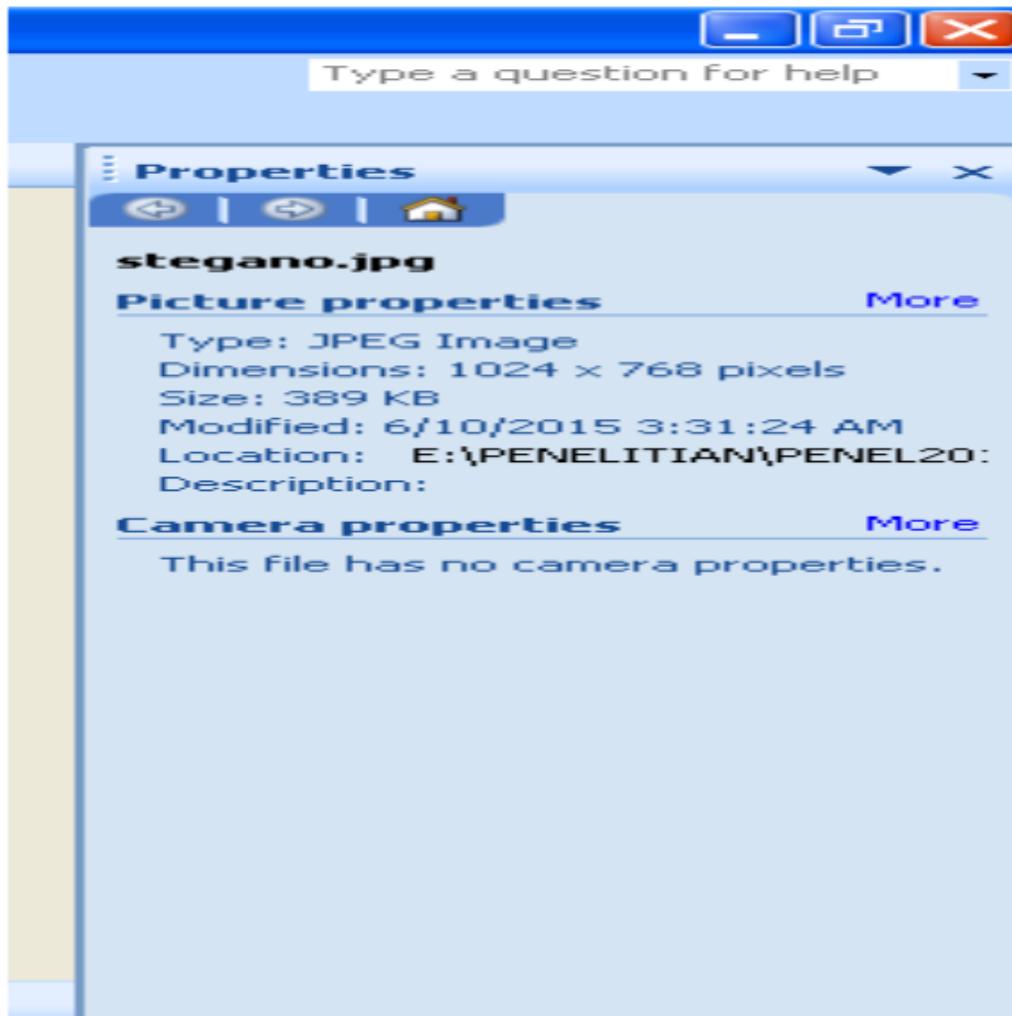
gambar dengan format JPEG, png dan bmp.

Fasilitas penyisipan dan ekstraksi dapat dimanfaatkan untuk mendeteksi watermark dalam suatu dokumen. Penandaan dapat berupa gambar, atau logo. Citra selanjutnya akan di sembunyikan kedalam bentuk vector. Jika yang disembunyikan adalah text maka bentuk matriknya $1 \times n$ -text. Citra memiliki bentuk matrik $m \times n$ -citra. Diketahui m-citra adalah baris citra dan n-citra adalah kolom citra. Selanjutnya ambil nilai intensitas pixel nya berupa merah, hijau dan biru. Selanjutnya akan diubah nilai vector yang disembunyikan kedalam binary citra yang disembunyikan di masukkan kedalam binary citra cover secara berurutan dengan cara melihat nilai bit terkecil dari citra cover dan nilai biner ke n dari citra yang disembunyikan. Dalam penelitian ini adalah document text dan gambar yang kemudian discan sehingga menjadi gambar dengan format JPEG, png dan bmp.

Pada proses penyisipan citra kedalam dokumen dilakukan dengan menentukan citra cover berformat png, jpg dan bmp, lalu menentukan citra hidden/yang akan disembunyikan. Dari proses penyisipan/steganografi, maka akan muncul citra baru yang mengandung pesan tersembunyi. Kemudian citra tersebut dapat disimpan kedalam format .png dan .bmp. Pengujian dilakukan untuk melihat kehandalan atau robustness dari aplikasi ini. Berdasarkan hasil pengujian menunjukan bahwa jika watermark yang disisipkan kedalam dokumen, akan terdapat perubahan terhadap dokumen cover dan watermark.

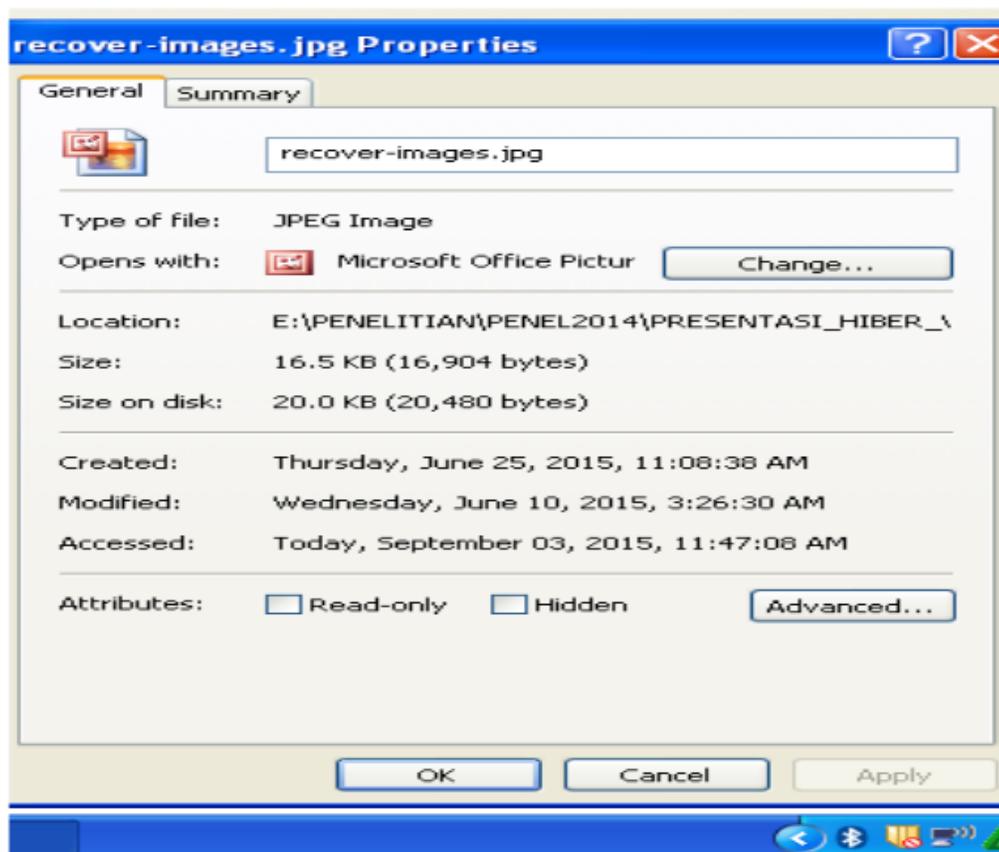
Selain itu proses pengukuran kinerja system yang dihasilkan dilakukan terhadap ukuran file. Dalam menganalisa ukuran file dilakukan dengan membandingkan ukuran dokumen sumber, ukuran

citra yang akan disisipkan dan ukuran hasil *steganografi*. Adapun hasil perbandingan dilakukan terhadap file yang sama akan tetapi dengan format yang berbeda, yaitu format file Jpeg, Bmp, Png. File sumber, file citra hidden dengan format Jpeg ditunjukkan pada gambar berikut ini .



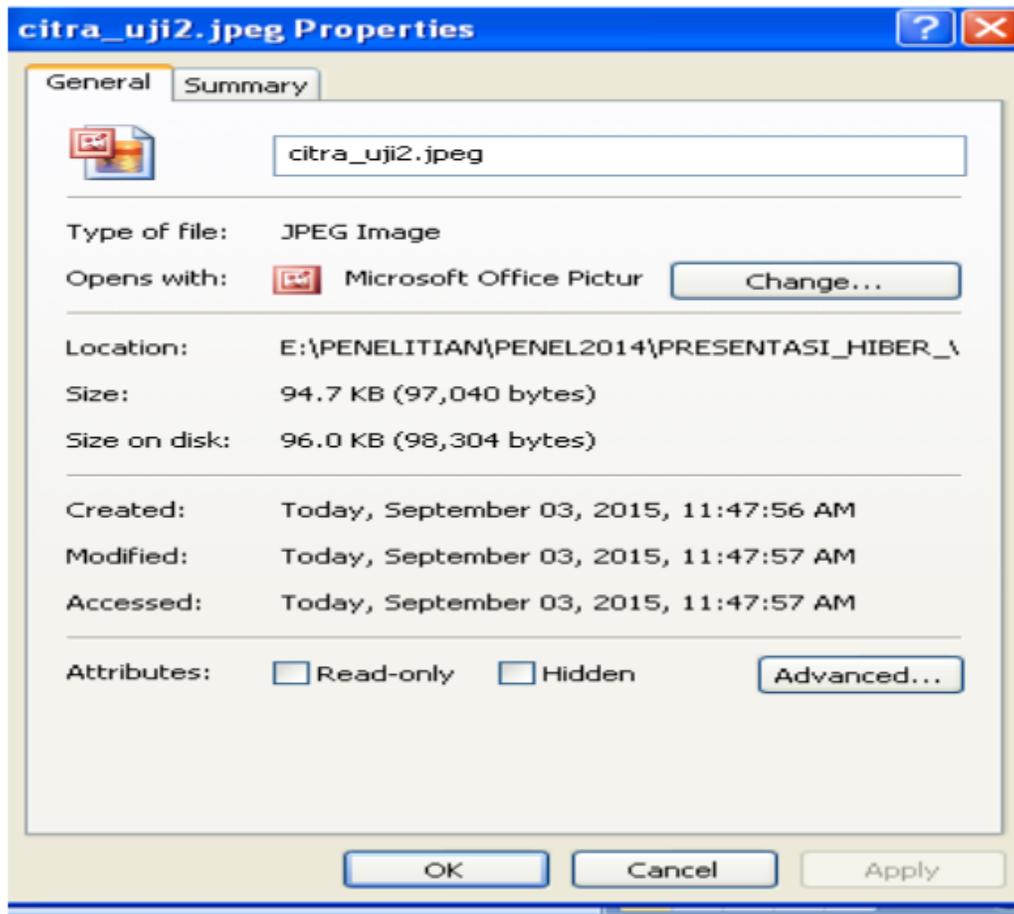
Gambar 11.10 Tampilan Ukuran File Sumber

Pada file sumber berukuran 389 Kb tersebut dilakukan penyisipan citra berukuran 16,5 Kb seperti ditampilkan pada gambar berikut.



Gambar 11.11. Tampilan Ukuran File Citra Hidden.

Hasil dari penggabungan file citra berformat Jpeg tersebut adalah steganografi dengan ukuran file 94,7 Kb.



Gambar 11.12 Tampilan Ukuran File Steganografi.

Selain dilakukan terhadap file berformat Jpeg, pengujian dilakukan pula terhadap file berformat Bmp dan Png. Adapun penggabungan tersebut menghasilkan ukuran file yang berbeda-beda besarnya seperti ditunjukkan pada tabel 1 berikut ini.

Tabel 11.1

File sumber		File Mark		File Hasil	
Jenis	Size	Jenis	Size	Jenis	Size
Jpeg	389 Kb	Jpeg	16.5 Kb	Jpeg	94.7 Kb
Jpeg	389 Kb	Jpeg	16.5 Kb	Bmp	2.25Mb
Jpeg	389 Kb	Jpeg	16.5 Kb	Png	975Kb

menunjukkan bahwa dari ukuran file yang dihasilkan dengan menggabungkan file yang sama akan tetapi penyimpanan hasil dengan format file/ekstensi yang berbeda akan menghasilkan ukuran file yang berbeda. Dalam proses ekstraksi citra steganografi yang dipilih berformat png dan bmp, selanjutnya dilakukan proses recover citra hidden, maka akan muncul citra yang disembunyikan dan citra hasil tersebut dapat disimpan dalam format .jpg, .png dan .bmp.

Aplikasi yang dihasilkan menyediakan dua proses yaitu proses penyisipan watermark dan proses ekstraksi. Dokumen yang telah diberi watermark, dapat di recover melalui proses ekstraksi yaitu mengembalikan citra yang disembunyikan pada citra asli. Hasil pengujian menunjukkan bahwa steganografi yang dihasilkan dari penggabungan file sumber berformat Jpeg dan file citra hidden berformat Jpeg menghasilkan ukuran file terkecil, dilanjutkan dengan jenis file png dan bmp. Akan tetapi untuk jenis file tersebut proses recovery belum discover dalam penelitian ini. Adanya watermark yang disisipkan pada sebuah dokumen berharga dalam dunia pendidikan berupa ijazah, sertifikat, piagam, dan lain-lain diharapkan dapat membantu memudahkan pihak yang berkepentingan untuk mendeteksi keaslian dokumen dalam dunia pendidikan. Hal tersebut diharapkan dapat membantu meningkatkan keamanan (*security*) dokumen sehingga dapat terjaga dari kepemilikan yang tidak syah.

e. Efektivitas Retrieval

Dari 50 query yang dikerjakan dalam penelitian ini, didapat bahwa efektivitas pencarian citra wajah (*image retrieval*) adalah 0,65 untuk precision dan 0,31 untuk recall, lebih jelasnya efektivitas pencarian citra dapat dilihat pada tabel berikut.

Tabel 11.2 Efektifitas image retrieval pencarian tanda tangan

QUERIES	PRECISION	RECALL	QUERIES	PRECISION	RECALL
1	0,15	0,45	18	0,88	0,3
2	0,20	0,45	19	0,89	0,3
3	0,25	0,45	20	0,9	0,1
4	0,30	0,40	21	0,94	0,35
5	0,33	0,55	22	0,95	0,1
6	0,35	0,45	23	0,95	0,35
7	0,43	0,35	24	0,95	0,35
8	0,44	0,35	25	0,96	0,3
QUERIES	PRECISION	RECALL	QUERIES	PRECISION	RECALL
9	0,45	0,30	26	0,98	0,35
10	0,47	0,35	27	1	0
11	0,48	0,35	28	0,74	0,15
12	0,50	0,25	29	0,75	0,05
13	0,55	0,25	30	0,76	0,65
14	0,55	0,25	31	0,77	0,45
15	0,56	0,30	32	0,78	0,3
16	0,61	0,40	33	0,79	0,3
17	0,61	0,20	34	0,81	0,1

Precision dan *Recall* merupakan dua parameter yang sering digunakan dalam menentukan efektif tidaknya dalam bidang *Content Based Image Retrieval*. Semakin tinggi nilai kedua parameter tersebut, terutama parameter *Precision* maka semakin tinggi efektifitas pencarian dan *matching*(tanda tangan). Diukur dengan parameter *precision* dan *recall* yaitu 0,65 dan 0,31 masing-masing. Efektifitas sebesar hampir 70% atau 0,67 menunjukkan bahwa pencarian citra dokumen cukup baik walaupun belum dikatakan sempurna.

Tugas Mandiri

1. Jelaskan pengertian dari watermarking.
2. Jelaskan perbedaan dari histogram dengan steganografi
3. Jelaskan perbedaan dari *visible watermark* dan *invisible watermark*,

DAFTAR PUSTAKA

- 57
Acharya, T., Ray, AK. 2005. *Image Processing Principles and Applications*. Canada: Wiley.
- 54
Arie Anggreani. 2009. "Sistem Pengenalan Kondisi Sel Darah Merah (Erythrocyte) Pada Citra Mikroskopis". Tugas Akhir. Teknik Elektro Udayana.
- Darma Putra, IKG. 2008. *Sistem Biometrika: Teori dan Aplikasi*. Yogyakarta: Andi Offset.
- 51
Gonzales, Rafael C. Woods, Richard E. 2002. *Digital Image Processing Second Edition*. USA: Prentice-Hall
- Jahne, Bernd. 2002. *Digital Image Processing*. Germany: Springer.
- 56
Jain, AK. 1995. *Fundamental of Digital Image Processing*. New Delhi: Prentice-Hall of India.
- Wiskara Manuaba. 2009. "Sistem Identifikasi Mikroaneurisma pada Citra Retina Digital". Tugas Akhir, Teknik Elektro Udayana.
- 57
Sonka, Milan., Hlavac, Vaclav., Boyle, Roger. 1998. *Image Processing Analysis and Machine Vision*. USA: Brooks and Cole Publishing.
2009. "Image Resolution".
http://en.wikipedia.org/wiki/Image_resolution. Akses Tanggal 6 Februari 2018.
- Laurent Deniau. 2003. file elektronik, disp-2003_L23A_14May03.pdf.
- Deniau, Laurent. 2003. "Image Processing". Slide presentasi.

- Fisher, Robert., Perkins, Simon., Walker, Ashley., Wolfart, Erik. 2003. "Hypermedia Image Processing Reference 2 (HIPR2)". Mathwork, Inc. 2001. 'Matlab Help Release 12.1'.
- Pratt, William K. 2007. *Digital Image Processing Fourth Edition*. California: John Wiley & Sons
- Zhang, D; Wai-Kin Kong; You, J; Wong, M. 2003. *Online Palmprint Identification*. IEEE Transaction on Pattern Analysis and Machine Intelligence, 25(9).
- "Introduction to Image Processing and Computer Vision". Buku Elektronik.
- Awcock G.J., Thomas R. 1996. *Applied Image Processing. International Edition*, McGraw-Hill, Inc.
- Green, Bill. 2002. "Canny Edge Detection Tutorial". http://www.pages.drexel.edu/~weg22/can_tut.html. Akses tanggal 5 Agustus 2017.
- Naser Jawas. 2009. "Segmentasi Citra Inis". Skripsi, Teknik Elektro Universitas Udayana.
- Young T.I., Gerbrands JJ., Van Vliet LJ. 1998. *Fundamental of Image Processing*. Netherland: Delft University of Technology.
- Ole Nielsen, "Introduction to Wavelet Analysis". File elektronik pde2002_Wavelet.pdf.
- Pu IM., 2006. *Fundamental Data Compression*. Elsevier.
- Salamon, D. 2002. *Data Compression The Complete Reference*. USA: Springer.
- Suksmadana, I.M. 2001. "Kompresi Citra Berwarna Menggunakan Alihragam Gelombang Singkat". Tesis. Yogyakarta: Universitas Gajah Mada.
- Arif Muntasa, Mochammad Hariadi, Mauridhi Her Purnomo, "A New Formulation of Face Sketch Multiple Features Detection Using Pyramid Parameters Model and Simultaneous Landmark Movement", Vol. 9 Nomor 9, International Journal of Computer Science and Network Security, 2009.

- Arnia F, Pramita N. *Enhancement of Iris Recognition System Based on Phase Only Correlation*. TELKOMNIKA, 2011;9(2): 387-394.
- Bedi Purnama, Ditari Salsabila E, *Identifikasi Telapak Tangan Menggunakan 2DPCA Plus PCAdan K-Nearest Neighbor*, SNATI 2013, UII Jogjakarta.
- Chang K, Bowver K, Bamabas V. *Comparison and Combination of Ear and Face Image in Appearance-Based Biometrics*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2003; 25: 1160-1165.
- D Maltoni, D. Maio, A.k. Jain, S. Prabhakar, "Handbook of Fingerprint Recognition", Springer, New York, 2003.
- DRIVE: *Digital Retinal Images for Vessel Extraction*, Image Sciences Institute. Available on: URL: [http://www.isi.uu.nl / Research/Databases/DRIVE](http://www.isi.uu.nl/Research/Databases/DRIVE), accessed 2011.
- Duta, N., Jain, A.K., Mardia, K.V., 2001. *Matching of PalmPrint*. Pattern Recognit. Lett. 23 (4), 477-485.
- Hoover A, Goldbaum M. *Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels*. IEEE Trans Med Imag. 2003;22(8):951-958.
- I Ketut Gede Darma Putra, Wira Bhuana, and Erdiawan, "Pembentukan Kode Telapak Tangan Berbasis Metode Gabor 2d", Makara, Teknologi, Vol. 15, No. 3, November 2011: 161-167.
- Hassanien. Ella. 2006. *A Copyright Protection using Watermarking Algorithm*. Informatics, Journal. Vol. 17(2): 187-198
- Onkar Dabeer, Kenneth Sullivan, and Upamanyu Madhow. 2007. *Detection Of Hiding In The Least Significant Bit*. IEEE Transactions On Signal Processing. Vol. 52(10)
- Saphiro G. Linda, George C. Stockman. 2005. *Computer Vision*, 1sted. Wiley, Prentice Hall, New Jersey.
- Gonzales Rafael and Richard E. Woods. 2010. *Image Processing: The Fundamentals*, 2thed. Wiley, Prentice Hall.

- Russ C. John. 2011. *The Image Processing Handbook*, 6th ed. CRC Press, Francis.
- Wallace K. Gregory. 1991. The JPEG Still Picture Compression Standard. *Communications of the ACM - Special issue on digital multimedia systems*, Vol 34 (4): 30-44
- Ajay Goel, O.P. Sahu, Ajay Goel. 2011. Improved Digital Watermarking Techniques and Data Embedding In Multimedia. *Cyber Journals Multidisciplinary Journals in Science and Technology*, Vol. 2(2): 164-168
- Besançon, R., Hède, P., Moellic, P.A., and Fluhr, C, "Cross-Media feedback strategies: Merging text and image information to improve image retrieval", 5th *Workshop of the Cross-Language Evaluation Forum, CLEF 2004, LNCS 3491, 2005*, pp. 709-717.
- Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. In: *Proceedings of the International World Wide Web Conference (WWW)*. Brisbane; 1998. pp. 107-117.
- Cho, W.C, dan D. Richards . 2004. Improvement of Precision and Recall for Information Retrieval in a Narrow Domain: Reuse of Concepts by Formal Concept Analysis, *Proceeding WI '04 Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence IEEE Computer Society Washington, DC, USA ©2004 ISBN:0-7695-2100-2*
- Inoue. M and Naonori U, "Retrieving Lightly Annotated Images using Image Similarities", *SAC'05, March 13-17, Santa Fe, New Mexico, USA 2005*
- Inoue. M, "On the need for annotation-based image retrieval", In *Workshop on Information Retrieval in Context (IRiX)*, Sheffield, UK, July 2004, pp. 44-46.
- Irianto, Y. Suhendro. 2008. *Image Retrieval In The Compressed Domain: Integration of low level features, image keywords and descriptions, segmentation techniques and image ontology for image retrieval in the compressed domain*, Ph.D Thesis, School of Informatics, University of Bradford, United Kingdom.

- Jones. G.J.F., Groves D., Khasin, A., Lam-Adesina, A. Mellebeek and Way, "Multilingual Information Access for Text, Speech and Images", 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004, LNCS 3491, Springer-Verlag GmbH, 2005, pp.653-663
- Levene, Mark. 2010. *An Introduction To Search Engines And Web Navigation*. John Wiley & Sons, Inc. Hoboken, New Jersey. pp78-89
- Limin Ma, Qiang Zhou, David Chelberg, and Mehmet Celenk. 2004. Shape-based Image Retrieval with Relevance Feedback, IEEE International Conference on Multimedia and Expo (ICME)
- Min Hyuk Chang¹, Kyung-Sik Seo. 2004. Histogram as a Measure of Shape Matching Image Processing: Algorithms and Systems III, edited by Edward R. Dougherty, Jaakko T. Astola, Karen O. Egiazarian, Proc. of SPIE-IS&T Electronic Imaging, IS&T · 0277-786X
- Petrou, M., and P. Bosdogianni, "Image processing: the fundamentals", John Wiley & Sons Ltd, England, 2005, p.265.
- Ramalingamurthy Meduri, Ashok Samal, and Scott L. Gardner Worm-Web Search: A Content-Based Image Retrieval (CBIR) System for the Parasite Image Collection in the Harold W. Manter Laboratory of Parasitology, University of Nebraska State Museum . *Technical report Number 11, 19 December 2008.*
- Smeulders, A., M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years", IEEE Transactions on Pattern Matching and Machine Intelligence, Vol.22, No. 12, 2000, pp. 1349-1379.
- Wang. J.Z, J. Li, and G. Wiederhold, "SIMPLicity: Semantics-Sensitive Integrated Matching for Picture Libraries", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 23, No. 9, 2001, pp. 947-963.
- Wang. X, L. Zhang, L.Zhang, F.Jing, W. Ma, "AnnoSearch: Image Auto-Annotation by Search", CVPR, 2006.

- Xiaoqian Xu ^a , D.J. Lee ^a , S. Antani ^b , and L.R. Long. 2006. Pre-Indexing for Fast Partial Shape Matching of Vertebrae Images. Proceedings of the 19th IEEE Symposium on Computer Based Medical Systems (CBMS'06) 0-7695-2517-1/06
- Ajay Goel, O.P. Sahu , Ajay Goel 2011. Improved Digital Watermarking Techniques and Data Embedding In Multimedia. CyberJournals Multidisciplinary Journals in Science and Technology, Vol. 02, No. 02, 2010, Pp.164-168
- Dugelay J. L. , S. Roche, C. Rey, G. Doërr, "Still-image water-marking robust to local geometric distortions," IEEE Trans. on Image Proc., vol. 15, no. 9, Pp. 2831-2842, 2006
- Ema Utami. 2009. Pendekatan Metode Least Bit Modification Untuk Merancang Aplikasi Steganography Pada File Audio Digital Tidak Terkompresi. Jurnal Dasi. Vol. 10 No. 1, ISSN: 1411-3201
- Hassaniena. Ella. 2006. A Copyright Protection using Watermarking Algorithm. Informatics, 2006, Vol. 17, No. 2, Pp. 187-198
- Jose-Angel Conchello, Carol J. Cogswell, Tony Wilson .2005. Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing. Proceedings of SPIE -- Volume 5701, Shanghai, China, Pp. 6492-6495
- Jun Zhang,, Yang Xiang,, Wanlei Zhou,, Lei Ye, and Yi Mu. 2011. Secure Image Retrieval Based on Visual Content and Watermarking Protocol. The Computer Journal, Published by Oxford University Press on behalf of The British Computer Society.
- Onkar Dabeer, Kenneth Sullivan, and Upamanyu Madhow. 2007. Detection Of Hiding In The Least Significant Bit. IEEE Transactions On Signal Processing, Vol. 52, No. 10.
- Gonzales Robert and Richard E. Woods. 2010. Image Processing: The Fundamentals. Wiley; 2nd edition, Prentice Hall.
- Russ C. John. 2011. The Image Processing Handbook, Sixth Edition Hardcover. CRC Press, 6th edition.

- Saphiro G. Linda , George C. Stockman. 2005. Computer Vision. Prentice Hall; 1 edition.
- Wallace K. Gregory. 1991. The JPEG Still Picture Compression Standard. Communications of the ACM - Special issue on digital multimedia systems, Volume 34 Issue 4, Pp. 30-44
- XuelongLi.2003.Watermarking in secure image retrieval. Pattern Recognition Letters, Elsevier, Vol. Pp.24 2431-2434
- Zhao Yu-qian,, Gui Wei-hua, Chen Zhen-cheng,, Tang Jing-tian,and Li Ling-yun. Medical Images Edge Detection Based on Mathematical Morphology. Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference.

Aplikasi Content Based Image Retrieval

ORIGINALITY REPORT

13%

SIMILARITY INDEX

11%

INTERNET SOURCES

4%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1	elib.unikom.ac.id Internet Source	1%
2	jutei.ukdw.ac.id Internet Source	1%
3	jurnal.ugm.ac.id Internet Source	1%
4	journal.uii.ac.id Internet Source	1%
5	www.damandiri.or.id Internet Source	<1%
6	Rusbianto Rusbianto, Indah Susilawati. "Identifikasi Citra Tulisan Tangan untuk Menentukan Karakter Kepribadian Introvert atau Extrovert dengan Metode LS Classifier", JMAI (Jurnal Multimedia & Artificial Intelligence), 2019 Publication	<1%
7	raymount.blogspot.com Internet Source	<1%
8	citee2015.jteti.ft.ugm.ac.id Internet Source	<1%
9	064-isnawan.blogspot.com Internet Source	<1%
10	dinus.ac.id Internet Source	<1%
11	www.acoustics.hut.fi Internet Source	<1%

12	Suhendro Y. Irianto, Dona Yuliawati, Sri Karnila. "Chapter 15 Image Based Search Engine - Like Using Color and Shape Features", Springer Science and Business Media LLC, 2019 Publication	<1%
13	abcd.unsiq.ac.id Internet Source	<1%
14	125-nadiarosiana.blogspot.com Internet Source	<1%
15	edoc.site Internet Source	<1%
16	izzuddinalkamal.blogspot.com Internet Source	<1%
17	Markus Dwiyanto Tobi. "RANCANG BANGUN PURWARUPA SISTEM PENDETEKSI KENDARAAN MENGGUNAKAN PUSTAKA OPENCV", Electro Luceat, 2015 Publication	<1%
18	badsector65.blogspot.com Internet Source	<1%
19	jurnal.fmipa.unila.ac.id Internet Source	<1%
20	basallamah.blogspot.com Internet Source	<1%
21	www.elektro.undip.ac.id Internet Source	<1%
22	ejournal3.undip.ac.id Internet Source	<1%
23	jurnal.uisu.ac.id Internet Source	<1%
24	Anton Sutrisno, Iman Fahruzi. "Identifikasi Tandatangan Digital Menggunakan Jaringan	<1%

Syaraf Tiruan Backpropagation", INOVTEK Polbeng - Seri Informatika, 2016

Publication

25	ejurnal.its.ac.id Internet Source	<1%
26	lib.itenas.ac.id Internet Source	<1%
27	guritac-tecnologi4.blogspot.co.id Internet Source	<1%
28	Aria Hendrawan. "Analisa Peningkatan Kualitas Citra Bawah Air Berbasis Koreksi Gamma Untuk Pencocokan Gambar Pada Algoritma SIFT", Jurnal Transformatika, 2014 Publication	<1%
29	Nurhasanah .. "Pendeteksian Tepi Citra CT Scan dengan Menggunakan Laplacian of Gaussian (LOG)", POSITRON, 2012 Publication	<1%
30	ejournal.unikama.ac.id Internet Source	<1%
31	achmatim.net Internet Source	<1%
32	eprints.unm.ac.id Internet Source	<1%
33	afridadamayanti.blogspot.co.id Internet Source	<1%
34	epdf.pub Internet Source	<1%
35	www.zeeopedia.com Internet Source	<1%
36	repository.untag-sby.ac.id Internet Source	<1%

37	ilmucampuraduk.blogspot.com Internet Source	<1%
38	ryanyudhasatria.blogspot.com Internet Source	<1%
39	eprints.uns.ac.id Internet Source	<1%
40	vdokumen.com Internet Source	<1%
41	shofwanalifauji.blogspot.com Internet Source	<1%
42	I Made Gede Sunarya, Made Windu Antara Kesiman, Ida Ayu Putu Purnami. "SEGMENTASI CITRA TULISAN TANGAN AKSARA BALI BERBASIS PROYEKSI VERTIKAL DAN HORIZONTAL", Jurnal Informatika, 2015 Publication	<1%
43	Nur Wakhidah. "IMPLEMENTASI METODE HOUGH TRANSFORM PADA IMAGE SEGMENTATION", Jurnal Transformatika, 2011 Publication	<1%
44	elektronika-dasar.web.id Internet Source	<1%
45	edu.ercess.co.in Internet Source	<1%
46	jurnaleeccis.ub.ac.id Internet Source	<1%
47	cirworld.org Internet Source	<1%
48	Erwin Dwika Putra, Stefanus Santosa. "OPTIMASI KEMAMPUAN SEGMENTASI OTSU PADA IDENTIFIKASI PLAT NOMOR	<1%

KENDARAAN INDONESIA MENGGUNAKAN METODE GAUSSIAN", Pseudocode, 2017

Publication

49 Duman Care Khrisne, Mohamad David Yusanto. <1%
"Content-Based Image Retrieval Menggunakan Metode Block Truncation Algorithm dan Grid Partitioning", S@CIES, 2015

Publication

50 Wara Mitha Nabella, Joko Sampurno, . <1%
Nurhasanah. "Analisis Citra Sinar-X Tulang Tangan Menggunakan Metode Thresholding Otsu Untuk Identifikasi Osteoporosis", POSITRON, 2013

Publication

51 ijsrst.com <1%
Internet Source

52 ijaers.com <1%
Internet Source

53 www.pgbiom.ufrpe.br <1%
Internet Source

54 doaj.org <1%
Internet Source

55 www.ijert.org <1%
Internet Source

56 www.popejohnpaulcollege.org <1%
Internet Source

57 link.springer.com <1%
Internet Source

58 pelita-informatika.com <1%
Internet Source

59 ugly.ph.tn.tudelft.nl <1%
Internet Source

60	grahailmu.co.id Internet Source	<1%
61	www.e-bi.gr Internet Source	<1%
62	www.wisesatravel.com Internet Source	<1%
63	tentanghackerdancracker.blogspot.com Internet Source	<1%
64	I Ketut Gede Putra, Wira Bhuana, Erdiawan Erdiawan. "Establishment Code Hand Palm (Palm Code) 2D Gabor-Based Method", Makara Journal of Technology, 2013 Publication	<1%
65	repositori.uin-alauddin.ac.id Internet Source	<1%
66	dodyirwandi.blogspot.com Internet Source	<1%
67	www.easyjapanese.org Internet Source	<1%
68	www.neliti.com Internet Source	<1%
69	beritabaik.web.id Internet Source	<1%
70	plj.ac.id Internet Source	<1%
71	Wu, X.. "Fisherpalms based palmprint recognition", Pattern Recognition Letters, 200311 Publication	<1%
72	Anita Rahmawaty. "MODEL ADOPSI PRODUK BANK SYARI'AH DI KUDUS: PENGEMBANGAN THEORY OF REASONED	<1%

ACTION DAN SYARIAH COMPLIANCE", INFERENSI, 2014

Publication

73	arsyanita1221cerita.blogspot.com Internet Source	<1%
74	duniafisika-suherman.blogspot.com Internet Source	<1%
75	catatanretnoanggra.blogspot.com Internet Source	<1%
76	pnj.ac.id Internet Source	<1%
77	www.facweb.iitkgp.ernet.in Internet Source	<1%
78	Vasudev Bhaskaran, Konstantinos Konstantinides. "Image and Video Compression Standards", Springer Nature, 1997 Publication	<1%

Exclude quotes On

Exclude matches Off

Exclude bibliography On

Aplikasi Content Based Image Retrieval

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

13/0

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71

PAGE 72

PAGE 73

PAGE 74

PAGE 75

PAGE 76

PAGE 77

PAGE 78

PAGE 79

PAGE 80

PAGE 81

PAGE 82

PAGE 83

PAGE 84

PAGE 85

PAGE 86

PAGE 87

PAGE 88

PAGE 89

PAGE 90

PAGE 91

PAGE 92

PAGE 93

PAGE 94

PAGE 95

PAGE 96

PAGE 97

PAGE 98

PAGE 99

PAGE 100

PAGE 101

PAGE 102

PAGE 103

PAGE 104

PAGE 105

PAGE 106

PAGE 107

PAGE 108

PAGE 109

PAGE 110

PAGE 111

PAGE 112

PAGE 113

PAGE 114

PAGE 115

PAGE 116

PAGE 117

PAGE 118

PAGE 119

PAGE 120

PAGE 121

PAGE 122

PAGE 123

PAGE 124

PAGE 125

PAGE 126

PAGE 127

PAGE 128

PAGE 129

PAGE 130

PAGE 131

PAGE 132

PAGE 133

PAGE 134

PAGE 135

PAGE 136

PAGE 137

PAGE 138

PAGE 139

PAGE 140

PAGE 141

PAGE 142

PAGE 143

PAGE 144

PAGE 145

PAGE 146

PAGE 147

PAGE 148

PAGE 149

PAGE 150

PAGE 151

PAGE 152

PAGE 153

PAGE 154

