

# Aplikasi *Image Retrieval* dengan Histogram Warna dan *Multi-scale GLCM*

Arwin Halim<sup>1</sup>, Hardy<sup>2</sup>, Mytosin<sup>3</sup>

STMIK Mikroskil, Jl. Thamrin No. 112, 124, 140, Telp. (061) 4573767, Fax. (061) 4567789

<sup>1,2,3</sup>Jurusan Teknik Informatika, STMIK Mikroskil, Medan

<sup>1</sup>arwin@mikroskil.ac.id, <sup>2</sup>hardy@mikroskil.ac.id, <sup>3</sup>mytosin.liu@mikroskil.ac.id

## Abstrak

*Content-based image retrieval adalah teknik pencarian gambar dari database gambar yang besar dengan menganalisa fitur-fitur gambar. Fitur gambar dapat berupa warna, tekstur, bentuk, dan lain-lain. Penelitian ini menggunakan fitur warna dan tekstur pada pencarian gambar. Histogram warna digunakan untuk mengekstrak fitur warna dengan pendekatan kuantisasi pada HSV. Fitur tekstur pada gambar diperoleh dari perhitungan Gray-Level Co-occurrence Matrix (GLCM) dan multi-scale GLCM. Multi-scale GLCM menggunakan Gaussian smoothing untuk mengurangi noise pada gambar dan memperhitungkan perbedaan skala pada gambar. Hasil pencarian gambar diperoleh dari perbandingan fitur warna dan tekstur pada database menggunakan Euclidean distance. Hasil penelitian menunjukkan pencarian gambar pada database Wang menggunakan histogram warna dan multi-scale GLCM memperoleh nilai precision yang lebih tinggi dibandingkan hanya memakai salah satu metode serta kombinasi histogram warna dan GLCM.*

**Kata kunci**— *content-based image retrieval, histogram warna, multi-scale GLCM*

## Abstract

*Content-based image retrieval is an image search techniques from large image database by analyzing features of the image. Image feature can be color, texture, shape, and others. This study uses color and texture features when Searching image. Color histogram is used to extract color features with quantization approach to HSV. Texture features in image obtained from the calculation of Gray-Level Co-occurrence Matrix (GLCM) and multi-scale GLCM. Multi-scale GLCM using Gaussian smoothing to reduce noise in the image and considering multiple scale from an image. Image search results obtained from the comparison of the features of color and texture in database using Euclidean distance. The results show an image search on Wang database using color histogram and multi-scale GLCM obtain higher precision value than just taking one of the method or combinations of color histogram and GLCM*

**Keywords**— *content-based image retrieval, color histogram, multi-scale GLCM.*

## 1. PENDAHULUAN

Perkembangan informasi multimedia gambar dan teknologi fotografi yang semakin meningkat menyebabkan jumlah gambar digital bertambah dengan sangat cepat, sehingga pencarian gambar menjadi semakin rumit. Pencarian gambar dapat dilakukan dengan pendekatan *text-based* dan *content-based* [1]. Pada pendekatan *text-based*, setiap gambar yang akan di-*training* harus diberi deskripsi, kata kunci, atau *tag* berupa teks secara manual. Namun, kekurangan pencarian dengan teks adalah sering terjadinya perbedaan perspektif, karena pandangan orang terhadap sebuah gambar yang sama bisa berbeda-beda. Tidak hanya itu, penginputan teks secara manual sangat menghabiskan waktu dan membutuhkan sumber daya manusia yang besar. Oleh karena itu, perlu dilakukan pendekatan *content-based* untuk mengatasi permasalahan *text-based* [2].

*Content-based image retrieval* (CBIR) adalah teknik pencarian gambar dari *database* gambar yang besar dengan menganalisa fitur-fitur dari gambar seperti warna, bentuk, tekstur, atau informasi lainnya yang bisa diekstrak dari gambar [1]. Terdapat banyak metode untuk menyelesaikan

permasalahan CBIR, di antaranya *Colour Models* [3], *Polar Raster Edge Sampling Signature* [4], dan *Local Binary Patterns* [5]. Penelitian ini menggunakan histogram warna [6] untuk mengolah fitur warna dan metode *multi-scale gray-level co-occurrence matrix* (GLCM) [7] untuk mengolah fitur tekstur. Penerapan histogram warna tidak memerlukan perhitungan yang rumit dan memiliki waktu komputasi yang cepat [8]. Fitur tekstur merepresentasikan struktur permukaan, perubahan intensitas, atau tingkat kecerahan [9]. GLCM bukan metode terbaru tetapi sudah teruji performa dan keefektifannya untuk mengolah fitur tekstur. Namun, GLCM belum memperhitungkan skala yang berbeda pada gambar ketika mengekstrak fitur teksturnya, sehingga dikembangkan metode *multi-scale GLCM* untuk mengatasi kekurangan tersebut [7].

Tujuan penelitian ini adalah untuk mengembangkan aplikasi pencarian gambar dengan menggunakan histogram warna dan *multi-scale GLCM*. *Multi-scale GLCM* menggunakan pendekatan *gaussian smoothing* untuk mengurangi *noise* dan mengurangi detail pada gambar. Histogram warna diperoleh dari perhitungan ruang warna HSV karena ruang warna HSV menghasilkan histogram warna terbaik di antara ruang warna lainnya [10], dan penelitian ini menggunakan dataset Wang karena memiliki tingkat akurasi yang tinggi dan tingkat *error* yang rendah [11].

## 2. KAJIAN PUSTAKA

### 2.1 Content-Based Image Retrieval

*Content-based image retrieval* (CBIR) adalah teknik pencarian gambar dari *database* gambar yang besar dengan menganalisa fitur-fitur dari gambar seperti warna, bentuk, tekstur, atau informasi lainnya yang bisa diekstrak dari gambar [1]. Ada 3 tahap fundamental pada *content-based image retrieval*, yaitu tahap ekstraksi fitur dari gambar, penyimpanan fitur, dan pencarian gambar [3]

Pada *content-based image retrieval*, gambar pada *database* melewati tahap ekstraksi fitur dan hasilnya di simpan ke *database*. Kemudian, gambar *query* juga diekstrak fiturnya kemudian dibandingkan dengan fitur pada *database* dengan menggunakan metode *similarity measure* (pengukuran kemiripan) yang menghasilkan hasil pencarian gambar yang mirip.

### 2.2 Histogram Warna

Histogram warna adalah representasi distribusi warna pada gambar [6]. Untuk gambar digital, histogram warna mewakili jumlah *pixel* yang memiliki warna pada masing-masing kelompok, dengan rentang warna tertentu yang mencakup ruang warna gambar tersebut. Langkah-langkah untuk menghitung fitur warna, yaitu:

#### a. Konversi ruang warna

Pada tahap ini, ubah semua warna pada gambar dari ruang warna RGB (*Red, Green, Blue*) menjadi ruang warna HSV. HSV adalah *Hue, Saturation, Value*. *Hue* digunakan untuk mewakili warna, *saturation* adalah banyaknya cahaya putih yang ditambahkan ke warna dasar, dan *value* adalah intensitas cahaya. Untuk mengubah RGB menjadi HSV diperlukan beberapa tahapan, yaitu: pertama hitung nilai  $R'$ ,  $G'$ , dan  $B'$  dengan Persamaan 1. Kedua, hitung nilai  $\Delta$  dengan Persamaan 2, 3 dan 4. Kemudian hitung nilai  $H$ ,  $S$ , dan  $V$  dengan Persamaan 5.

$$R' = \frac{R}{255} \quad G' = \frac{G}{255} \quad B' = \frac{B}{255} \quad (1)$$

$$Cmax = \max(R', G', B') \quad (2)$$

$$Cmin = \min(R', G', B') \quad (3)$$

$$\Delta = Cmax - Cmin \quad (4)$$

$$H = \begin{cases} 60^\circ \times \left( \frac{G' - B'}{\Delta} \bmod 6 \right), C_{max} = R' \\ 60^\circ \times \left( \frac{B' - R'}{\Delta} + 2 \right), C_{max} = G' \\ 60^\circ \times \left( \frac{R' - G'}{\Delta} + 4 \right), C_{max} = B' \end{cases} \quad S = \begin{cases} 0, C_{max} = 0 \\ \frac{\Delta}{C_{max}}, C_{max} \neq 0 \end{cases} \quad (5)$$

$$V = C_{max}$$

#### b. Kuantisasi nilai HSV

Pada ruang warna HSV, setiap komponen memiliki rentang nilai yang besar. Jika menggunakan nilai H, S, dan V secara langsung, maka memerlukan banyak komputasi. Jadi lebih baik jika ruang warna HSV dikuantisasi secara *non-equal intervals*, yaitu H: 8 bagian, S: 3 bagian, dan V: 3 bagian, yang menggunakan Persamaan 6.

$$H = \begin{cases} 0 \text{ if } h \in [316, 20] \\ 1 \text{ if } h \in [21, 40] \\ 2 \text{ if } h \in [41, 75] \\ 3 \text{ if } h \in [76, 155] \\ 4 \text{ if } h \in [156, 190] \\ 5 \text{ if } h \in [191, 270] \\ 6 \text{ if } h \in [271, 295] \\ 7 \text{ if } h \in [296, 315] \end{cases} \quad S = \begin{cases} 0 \text{ if } s \in [0, 0.2] \\ 1 \text{ if } s \in [0.2, 0.7] \\ 2 \text{ if } s \in [0.7, 1] \end{cases} \quad (6)$$

$$V = \begin{cases} 0 \text{ if } v \in [0, 0.2] \\ 1 \text{ if } v \in [0.2, 0.7] \\ 2 \text{ if } v \in [0.7, 1] \end{cases}$$

Kemudian digabungkan menjadi 8x3x3 histogram dan mendapatkan vektor 72 dimensi. Karena kemampuan mata manusia untuk membedakan warna sangat terbatas, vektor 3 dimensi untuk kombinasi nilai H, S, dan V akan dinormalisasi menjadi vektor 1 dimensi dengan Persamaan 7. Ini akan mengkuantisasi seluruh ruang warna HSV menjadi 72 bagian warna dengan 1 dimensi.

$$G = 9H + 3S + V \quad (7)$$

Pada tahap ini, gambar yang sudah dikuantisasi akan dihitung histogram warnanya, yaitu frekuensi distribusi nilai HSV setiap *pixel* pada gambar yang telah dikuantisasi.

### 2.3 Gray-Level Co-occurrence Matrix (GLCM)

GLCM dapat memberikan informasi tentang distribusi spasial dari *gray-level* pada tekstur gambar [7]. GLCM menandakan seberapa sering sebuah *pixel* dengan nilai *i* berhubungan secara spasial dengan *pixel* yang bernilai *j*. Pada setiap GLCM, masing-masing elemen  $p(i, j)$  adalah total kemunculan suatu *pixel* *i* dengan *pixel* *j* untuk arah tertentu (0, 45, 90, 135). Untuk menghitung fitur akhir GLCM, maka hitung fitur untuk empat arah GLCM dan hitung rata-rata fiturnya. Pada dasarnya, GLCM adalah sebuah *matrix* simetris dan levelnya ditentukan oleh *gray-level* dari gambar. Intensitas nilai *grayscale* pada gambar dikurangi dari 256 menjadi 8, sehingga *matrix*-nya berukuran 8x8. Elemen pada *matrix* dihitung dengan Persamaan 8.

$$P(i, j | d, \theta) = \frac{P(i, j | d, \theta)}{\sum_i \sum_j P(i, j | d, \theta)} \quad (8)$$

Empat fitur GLCM yang paling sering digunakan adalah *energy*, *contrast*, *correlation*, dan *homogeneity*. Berikut penjelasan masing-masing fitur beserta rumusnya:

#### a. Energy

*Energy* adalah pengukuran tekstur dari gambar *grayscale* yang merepresentasikan perubahan *homogeneity*, merefleksikan keseragaman berat dan tekstur distribusi gambar *grayscale*, yang menggunakan Persamaan 9.

$$E = \sum_x \sum_y P(x,y)^2 \quad (9)$$

b. *Contrast*

*Contrast* adalah diagonal utama yang mengukur nilai matrix yang terdistribusi, merefleksikan kejelasan gambar dan tekstur kedalaman bayangan, yang menggunakan Persamaan 10.

$$I = \sum_x \sum_y (x - y)^2 P(x, y) \quad (10)$$

c. *Correlation*

*Correlation* mengukur keteraturan tekstur gambar, ketika setiap nilai *matrix* sama besar, maka menghasilkan nilai korelasi yang minimum, dan sebaliknya jika nilainya sangat tidak merata, nilai korelasinya akan semakin besar. *Correlation* dihitung dengan menggunakan Persamaan 11.

$$C = - \sum_x \sum_y P(x, y) \log P(x, y) \quad (11)$$

d. *Homogeneity*

*Homogeneity* mengukur nilai perubahan lokal pada tekstur gambar, dengan Persamaan 12.

$$H = \sum_x \sum_y \frac{1}{1 + (x - y)^2} P(x, y) \quad (12)$$

## 2.4 Multi-Scale Gray-Level Co-occurrence Matrix (GLCM)

GLCM akan dikembangkan menjadi *multi-scale GLCM* untuk memperhitungkan skala yang berbeda. Pada penelitian ini, *multi-scale GLCM* menggunakan pendekatan *gaussian smoothing*. *Gaussian smoothing* merupakan teknik untuk mengaburkan gambar dengan tujuan mengurangi *noise* dan mengurangi detail pada gambar. *Gaussian smoothing* dihitung dengan Persamaan 13.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (13)$$

## 2.5 Euclidean Distance

*Euclidean distance* digunakan untuk mengukur tingkat kesamaan di antara dua vektor yang berbeda. Untuk menghitung *euclidean distance* digunakan Persamaan 14.

$$d(Q, D) = \sqrt{\sum_{i=1}^n [Q_i - D_i]^2} \quad (14)$$

Setelah menghitung *euclidean distance*, fitur warna dan fitur tekstur akan digabung dengan bobot  $w = 0.5$  (Deepak, J., et al, 2014), menggunakan Persamaan 15.

$$d(Q, D) = w_1 * d(FCQ, FCD) + w_2 * d(FTQ, FCD) \quad (15)$$

### 3. METODE PENGEMBANGAN SISTEM

Metodologi yang digunakan dalam penelitian ini adalah *waterfall* dengan langkah-langkah sebagai berikut:

1. Perencanaan berupa kajian pustaka terhadap literatur berupa buku, website, jurnal, conference yang berhubungan dengan topic content-based image retrieval khususnya histogram warna, GLCM dan *multi-scale GLCM*.
2. Analisis sistem. Pada tahapan ini, dilakukan analisis kebutuhan fungsional dengan use case diagram, analisis data dengan entity-relational diagram dan analisis proses dengan *Flowchart*.
3. Perancangan sistem. Pada tahapan ini, dilakukan proses perancangan fisik untuk basis data dan *user interface* aplikasi pencarian gambar CBIR.
4. Implementasi. Pada tahapan ini, analisis dan rancangan yang dibuat dibangun menjadi aplikasi dengan menggunakan bahasa pemrograman C#.NET.
5. Pengujian. Pada tahapan ini, dilakukan pengujian dengan membandingkan hasil pencarian gambar menggunakan kombinasi teknik histogram warna, GLCM dan *multi-scale GLCM*. Pengujian terhadap data dilakukan dengan *leave-one-out cross-validation*.

### 4. HASIL DAN PEMBAHASAN

#### 4.1 Analisis Kebutuhan Fungsional

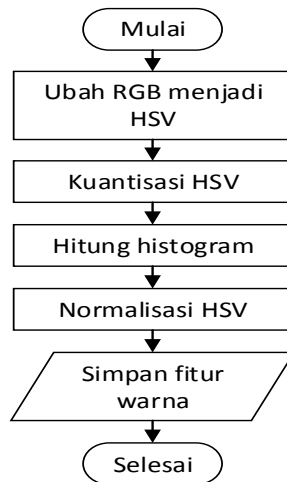
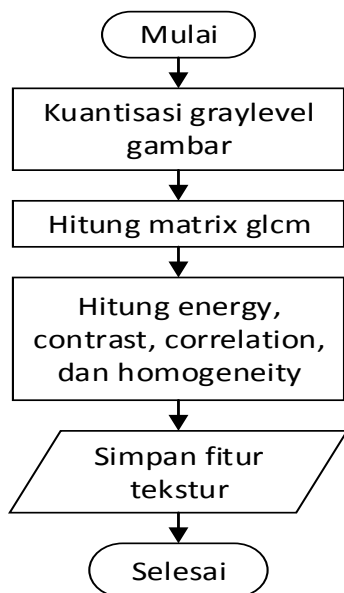
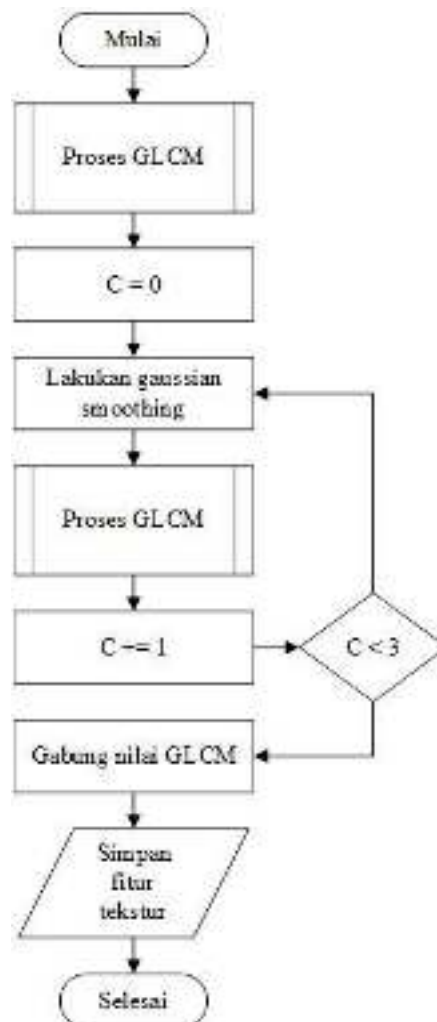
Analisis kebutuhan fungsional digambarkan dengan menggunakan use case diagram. Aktor pada use case diagram terdiri dari admin dan user. Gambar 1 menunjukkan use case diagram dari aplikasi image retrieval dengan histogram warna dan *multi-scale GLCM*.



Gambar 1 Use Case Diagram Aplikasi Image Retrieval dengan Histogram Warna dan *Multi-scale GLCM*

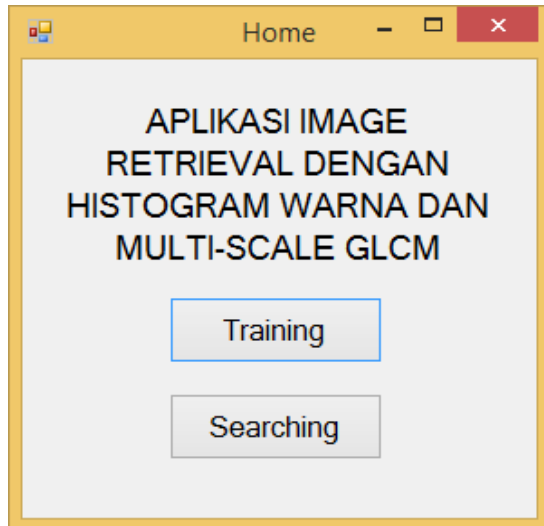
#### 4.2 Analisis Proses

Teknik pencarian yang digunakan dalam menemukan gambar yang sama pada aplikasi terdiri dari algoritma histogram warna, GLCM dan *multi-scale GLCM*. *Flowchart* untuk histogram warna dapat ditunjukkan pada Gambar 2. *Flowchart* untuk GLCM ditunjukkan pada Gambar 3 dan *multi-scale GLCM* pada Gambar 4.

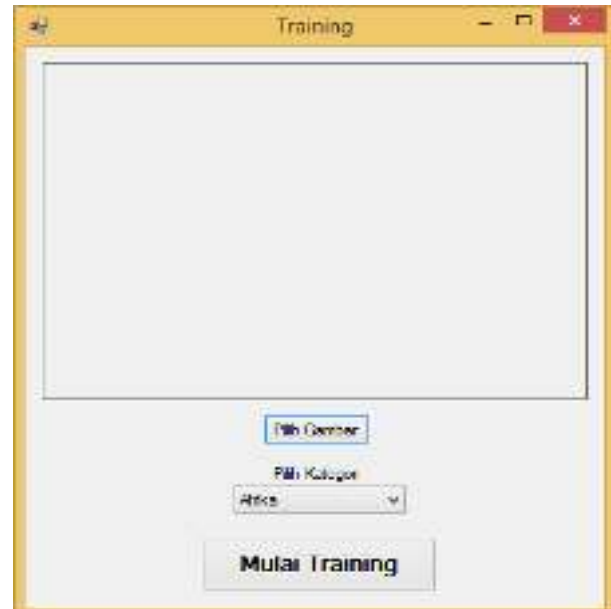
Gambar 2 *Flowchart* Algoritma Histogram WarnaGambar 3 *Flowchart* Algoritma GLCMGambar 4 *Flowchart* Algoritma Multi-scale GLCM

### 4.3 Desain *User Interface*

Hasil dari implementasi berupa tampilan dan fungsionalitas dari aplikasi pencarian gambar. Tampilan awal dari aplikasi pencarian gambar terdiri dari menu *training* dan *Searching* seperti pada Gambar 6. Ketika menu *training* dipilih, maka muncul sebuah form baru untuk memasukkan data pembelajaran gambar pada aplikasi seperti pada Gambar 7.

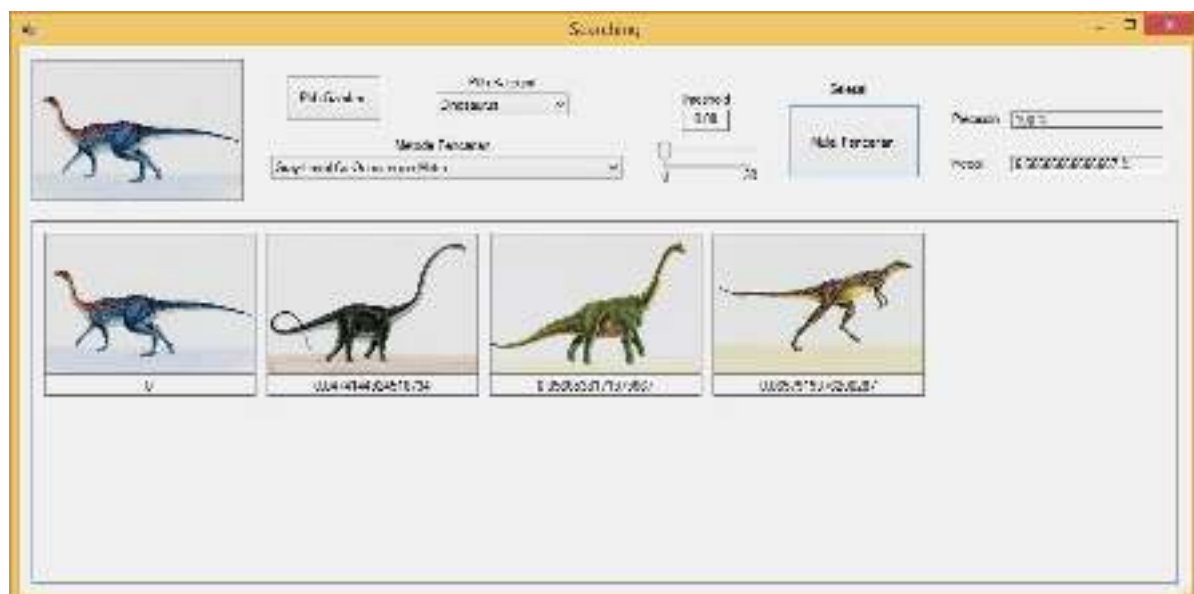


Gambar 6 Tampilan Awal Aplikasi Pencarian



Gambar 7 Tampilan Halaman *Training*

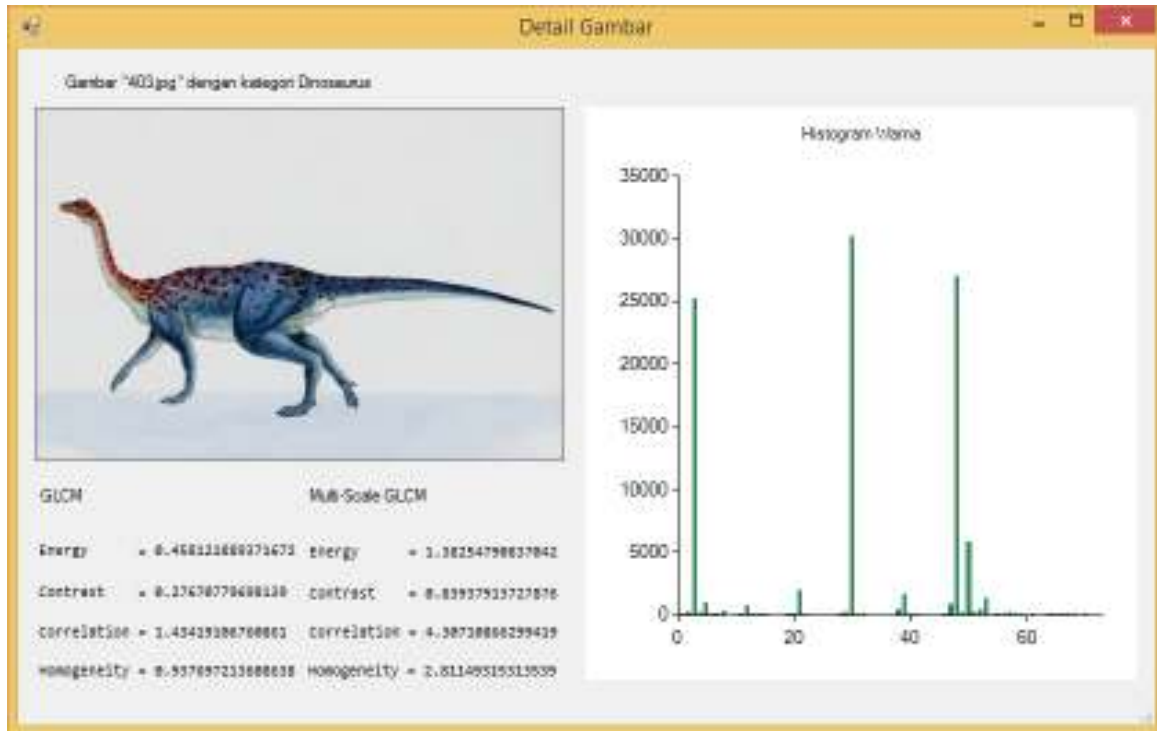
Setelah *training*, pengguna dapat melakukan pencarian gambar dengan memilih menu *Searching* seperti pada Gambar 8. Ketika melakukan pencarian, gambar yang mirip ditampilkan sesuai dengan nilai *threshold* yang telah ditentukan.



Gambar 8 Tampilan Halaman *Searching* dan Hasil Pencarian Gambar

Rincian informasi gambar dapat ditampilkan dengan memilih (klik) salah satu gambar hasil pencarian, seperti terlihat pada Gambar 9.





Gambar 9 Tampilan Halaman Rincian Gambar

#### 4.4 Pengujian

Setelah aplikasi selesai dikembangkan, dilakukan pengujian terhadap teknik pencarian gambar dengan histogram warna, GLCM dan *multi-scale GLCM*. Evaluasi pengujian dilakukan dengan nilai *threshold* tertentu sehingga didapatkan nilai presisi dan *recall* dari pencarian. Evaluasi hasil pengujian presisi dan *recall* dengan *threshold* 17 untuk histogram warna (HW), 0.25 untuk GLCM, dan 0.6 untuk *multi-scale GLCM* (MS-GLCM), 9 untuk kombinasi HW+GLCM, dan 9 untuk HW+MS-GLCM.

Tabel 1. Evaluasi Pengujian Presisi Aplikasi Pencarian Gambar

Kategori	Average Precision (%)				
	HW	GLCM	MS-GLCM	HW+GLCM	HW+MS-GLCM
Afrika	33.796	16.185	18.565	33.740	39.114
Bangunan	24.397	20.833	21.260	23.783	26.302
Bunga	60.173	73.853	81.848	64.853	79.470
Bus	34.847	33.320	36.555	33.125	41.233
Dinosaurus	92.652	71.246	80.073	92.847	95.502
Gajah	28.392	18.257	20.041	27.040	31.218
Gunung	34.466	15.102	16.808	33.083	37.566
Kuda	78.479	23.880	27.305	77.342	80.458
Makanan	31.694	17.099	18.150	30.115	33.629
Pantai	32.595	18.630	19.469	31.846	36.029
<b>Total Average (%)</b>	<b>45.149</b>	<b>30.841</b>	<b>34.007</b>	<b>44.777</b>	<b>50.052</b>

Berdasarkan Tabel 1, nilai rata-rata *precision* untuk histogram warna lebih bagus dari GLCM dan *multi-scale GLCM*. Kemudian, kombinasi histogram warna dan GLCM memperoleh nilai rata-rata *precision* yang lebih bagus dari GLCM dan *multi-scale GLCM*, tetapi masih lebih kecil dari nilai



*precision* histogram warna. Selanjutnya, kombinasi histogram warna dan *multi-scale GLCM* memperoleh nilai rata-rata *precision* yang paling bagus dibandingkan menggunakan salah satu metode serta kombinasi histogram warna dan GLCM.

Tabel 2. Evaluasi Pengujian Presisi Aplikasi Pencarian

Kategori	Average Recall (%)				
	HW	GLCM	MS-GLCM	HW+GLCM	HW+MS-GLCM
Afrika	63.972	30.222	23.305	72.166	66.833
Bangunan	33.027	27.944	18.972	37.722	31.722
Bunga	30.222	56.888	45.916	33.611	30.722
Bus	80.694	35.833	25.138	85.611	80.722
Dinosaurus	38.805	39.944	29.333	43.777	39.666
Gajah	29.138	38.611	29.555	33.611	29.444
Gunung	31.388	29.666	20.222	34.888	30.388
Kuda	70.333	45.944	34.277	73.333	70.333
Makanan	66.666	32.666	23.25	71.888	65
Pantai	25.972	32.166	22.805	29.6111	24.777
<b>Total Average (%)</b>	<b>47.022</b>	<b>36.988</b>	<b>27.277</b>	<b>51.622</b>	<b>46.961</b>

Berdasarkan Tabel 2, nilai rata-rata *recall* untuk histogram warna lebih bagus dari GLCM dan *multi-scale GLCM*. Kemudian, kombinasi histogram warna dan GLCM memperoleh nilai rata-rata *recall* yang paling bagus dibandingkan menggunakan salah satu metode serta kombinasi histogram warna dan *multi-scale GLCM*.

#### 4. KESIMPULAN

Hasil pencarian gambar dengan kombinasi histogram warna dan *multi-scale GLCM* mendapatkan nilai *average precision* yang lebih bagus dibandingkan pada saat menggunakan salah satu metode serta histogram warna dan *gray-level co-occurrence matrix* yang belum memperhitungkan skala yang berbeda, yaitu dengan nilai *precision* 50.052% dan *recall* 46.961% untuk nilai *threshold* 9.

#### 5. SARAN

Penelitian ini telah memperhitungkan skala pada gambar ketika dilakukan pencarian, namun belum memperhitungkan rotasi pada gambar yang dicari. Penelitian selanjutnya dapat dilanjutkan dengan memperhatikan aspek rotasi gambar pada saat pencarian gambar.

#### DAFTAR PUSTAKA

- [1] Singh, B. dan Ahmad, W., 2014, Content Based Image Retrieval: A Review Paper, International Journal of Computer Science and Mobile Computing, volume 3, issue 5.
- [2] Prabhu, J. dan Kumar, J. S., 2014, Wavelet Based Content Based Image Retrieval using Color and Texture Feature Extraction by Gray Level Coocurrence Matrix and Color Coocurrence Matrix, Journal of Computer Science, volume 10, halaman 15-22.
- [3] Arthi, K., Vijayaraghavan, J., 2013, Content Based Image Retrieval Algorithm using Colour Models, International Journal of Advanced Research in Computer and Commucation Engineering, volume 2, issue 3.
- [4] Mathew, S. P., Balas, V. E., Zachariah K. P., Samuel, P., 2014, A Content-based Image

Retrieval System Based on Polar Raster Edge Sampling Signature, *Acta Polytechnica Hungarica*, volume 11, nomor 3.

- [5] Wu, Z., Xia, Y., Wan, S., 2014, An Extension to the Local Binary Patterns for Image Retrieval, *Advanced Science and Technology Letters*, volume 45, halaman 81-85.
- [6] Deepak, J., Sreekumar, K., Tharani, S. T., 2014, Content Based Image Retrieval using HSV-Color Histogram and GLCM, *International Journal of Advance Research in Computer Science and Management Studies*, volume 2, issue 1.
- [7] Siqueira, F. R. D., Schwartz, W. R., Pedrini, H., 2012, Multi-Scale Gray Level Co-Occurrence Matrices for Texture Description, *Institute of Computing, University of Campinas*.
- [8] Hemachandran, K. dan Singha, M., 2012, Content Based Image Retrieval using Color and Texture, *An International Journal*, volume 3, nomor 1.
- [9] Pathak, B. dan Barooah, D., 2013, Texture Analysis Based on the Gray-Level Co-Occurrence Matrix Considering Possible Orientations, *International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering*, volume 2, issue 9.
- [10] Ali, N. M., Rashid, N. K. A. M., Mustafah, Y. M., 2013, Performance Comparison between RGB and HSV Color Segmentations for Road Signs Detection, *Applied Mechanics and Materials*, volume 393, halaman 550-555.
- [11] Deselaers, T., Keysers, D., Ney, H., 2007, Features for Image Retrieval: An Experimental Comparison.