

KLASIFIKASI JENIS BUNGA KAMBOJA JEPANG (*ADENIUM SP.*) BERDASARKAN CITRA MAHKOTA MENGGUNAKAN EKSTRAKSI FITUR WARNA DAN DETEKSI TEPI

^[1]Urai Nur Ichسانی, ^[2]Dedi Triyanto, ^[3]Ikhwan Ruslianto

^[1] ^[2] ^[3]Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura
Jl. Prof. Dr. H. Hadari Nawawi, Pontianak
Telp./Fax.: (0561) 577963

e-mail:

^[1]u12nurichسانی@gmail.com, ^[2]dedi.triyanto@siskom.untan.ac.id,

^[3]ikhwanruslianto@siskom.untan.ac.id

Abstrak

Warna dan bentuk dari mahkota bunga antara bunga yang satu dengan bunga yang lain memiliki perbedaan, seperti pada bunga kamboja jepang (*Adenium sp.*). Bunga *Adenium sp.* dibagi menjadi 2 jenis, yaitu *adenium spesies* dan *adenium hibrida*. Semakin banyak persilangan antara jenis *adenium* satu dengan *adenium* yang lainnya, maka semakin banyak pula tercipta jenis *adenium hibrida* baru. Penelitian ini bertujuan untuk membuat suatu aplikasi klasifikasi jenis bunga *Adenium sp.* yang memanfaatkan pengolahan citra menggunakan fitur gabungan dari nilai pixel histogram hue dan deteksi tepi sobel. Hasil dari penelitian memberikan persentase keberhasilan aplikasi dalam mengklasifikasikan jenis bunga *Adenium sp.* sebesar 96%, yang mana klasifikasi berdasarkan nilai kemiripan histogram hue tertinggi sebagai acuan adalah sebesar 25%, klasifikasi berdasarkan fitur gabungan dengan nilai kemiripan citra hasil deteksi tepi sobel tertinggi sebagai acuan adalah sebesar 33,33%, dan klasifikasi berdasarkan fitur gabungan dengan nilai kemiripan histogram hue tertinggi sama dengan nilai kemiripan citra hasil deteksi tepi sobel tertinggi sebesar 41,67%. Berdasarkan hasil tersebut, maka dapat disimpulkan bahwa klasifikasi menggunakan fitur gabungan dengan metode sobel dan histogram hue dapat digunakan untuk klasifikasi jenis bunga kamboja jepang (*Adenium sp.*).

Kata Kunci : Histogram Hue, Klasifikasi *Adenium sp.*, Operator Sobel, Pengolahan Citra

1. PENDAHULUAN

Warna dan bentuk dari mahkota bunga antara bunga yang satu dengan bunga yang lainnya memiliki perbedaan, seperti pada bunga kamboja jepang (*Adenium sp.*). Pada *Adenium sp.*, terdapat beberapa bunga yang memiliki pola dan warna mahkota yang hampir sama namun dari jenis yang berbeda, sehingga dapat mempersulit dalam pengklasifikasiannya, oleh sebab itu diperlukan bantuan teknologi pengolahan citra yang dapat mengenali karakteristik mahkota bunga, baik menurut bentuk, menurut warna, maupun gabungan dari keduanya.

2. LANDASAN TEORI

2.1. *Adenium sp.*

Adenium sp. berasal dari daerah gurun pasir di daratan Afrika, seperti di Senegal

sampai Sudan, Kenya, Mozambique, Namibia, dan sekitarnya. *Adenium sp.* juga mempunyai julukan *Desert Rose* yang artinya mawar padang pasir.[1]

Bunga *Adenium sp.* berbentuk seperti terompet dengan 5 helai mahkota dan 5 kelopak. Jenis mahkotanya ada yang runcing, bergerigi, atau bulat. Di tengah tabung terompet terdapat benang sari. Secara umum, warna bunga *Adenium sp.* adalah merah muda. Diameter bunga rata-rata 7-8,5cm, tetapi ada jenis *Adenium sp.* yang memiliki diameter 2-4cm. Ukuran bunga pada jenis yang sama dipengaruhi oleh kesehatan dan umur tanaman, bukan tinggi tanaman.[1]

Obyek yang akan digunakan pada penelitian ini merupakan bunga dari jenis *adenium*, macam jenisnya diberikan pada tabel berikut.

Tabel 1. Jenis Adenium Hibrida Yang Digunakan Di Dalam Penelitian

No	Jenis Bunga	Gambar	Tujuan Penggunaan	
			Latih	Uji
1	Cabang Seribu		Ya	Ya
2	Cellona		Ya	Ya
3	Flutter		Ya	Ya
4	Lucky Star		Ya	Ya
5	Red Jobilee		Ya	Ya
6	Red Lucas		Ya	Ya
7	Harry Potter		Ya	Ya
8	Chiara		Tidak	Ya
9	Triple White Jasmine		Tidak	Ya
10	Triple Shocking Pink		Tidak	Ya
11	Golden Jobilee		Tidak	Ya

2.2. Citra Digital

Citra digital adalah citra kontinu yang diubah dalam bentuk diskrit, baik koordinat ruang maupun intensitas cahayanya. Citra digital dapat dinyatakan dalam matrik dua dimensi $f(x,y)$ dimana 'x' dan 'y' merupakan koordinat *pixel* dalam matrik dan 'f' merupakan derajat intensitas *pixel* tersebut. Citra digital berbentuk matrik dengan ukuran $M \times N$ akan tersusun sebagai berikut. [2]

$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ f(2,0) & f(2,1) & f(2,2) & \dots & f(2,N-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{pmatrix}$$

Dimana :

- M = banyaknya baris pada array citra
- N = banyaknya kolom pada array citra

2.3. Model Warna Hue

Hue adalah variabel yang menyatakan warna dari merah hingga violet. Hue mengukur sudut sekitar roda warna (merah pada 0° , 120° di hijau, dan biru di 240°). Nilai dari Hue berkisar antara 0° - 360° . Skala Hue ditampilkan pada Gambar 1. [3]



Gambar 1. Skala Hue

(Sumber : Rakhmawati, 2013)

2.4. Konversi RGB ke HSV

RGB merupakan singkatan dari *Red*, *Green*, *Blue*. Warna yang dibentuk oleh model warna adalah hasil campuran dari warna-warna primer merah, hijau, dan biru berdasarkan komposisi tertentu. Perhitungan konversi RGB ke HSV dirumuskan sebagai berikut : [4]

$$H = \tan \left[\frac{3(G-B)}{(R-G)+(R-B)} \right] \dots \dots \dots (1)$$

$$S = 1 - \frac{\min(R,G,B)}{V} \dots \dots \dots (2)$$

$$V = \frac{R+G+B}{3} \dots \dots \dots (3)$$

Pada rumus di atas, jika $S = 0$ maka H tidak dapat ditentukan, untuk itu diperlukan normalisasi RGB dahulu dengan rumus berikut : [4]

$$r = \frac{R}{R+G+B} \dots \dots \dots (4)$$

$$g = \frac{G}{R+G+B} \dots \dots \dots (5)$$

$$b = \frac{B}{R+G+B} \dots \dots \dots (6)$$

Setelah nilai r , g , dan b sudah dinormalisasi, maka rumus transformasi RGB ke HSV sebagai berikut : [4]

$$V = \max(r, g, b) \dots \dots \dots (7)$$

$$S = \begin{cases} 0 & \text{jika } V = 0 \\ V - \frac{\min(r,g,b)}{V} & \text{jika } V > 0 \end{cases} \dots \dots \dots (8)$$

$$H = \begin{cases} 0 & \text{jika } S = 0 \\ \frac{60x(g-b)}{SxV} & \text{jika } V = r \\ 60x \left[2 + \frac{(b-r)}{SxV} \right] & \text{jika } V = g \\ 60x \left[4 + \frac{(r-g)}{SxV} \right] & \text{jika } V = b \end{cases} \dots \dots \dots (9)$$

$$H = H + 360 \quad \text{jika } H < 0 \dots \dots \dots (10)$$

2.5. Histogram Citra

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 ditampilkan secara grafik distribusi tiap level nilai *pixel* tersebut. [4]

2.6. Thresholding (Pengambangan)

Segmentasi paling sederhana dilaksanakan dengan menggunakan *threshold* (ambang intensitas). Proses pengambangan akan menghasilkan citra biner, yaitu citra yang memiliki 2 nilai tingkat keabuan yaitu hitam dan putih. Secara umum proses pengambangan citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut. [4]

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) \geq T \\ 0 & \text{if } f(x,y) < T \end{cases} \dots\dots\dots (11)$$

Dengan $g(x,y)$ adalah citra biner dari citra *grayscale* $f(x,y)$ dan T menyatakan nilai ambang (*threshold*). Nilai T memegang peranan yang sangat penting dalam proses pengambangan. Kualitas hasil citra biner sangat tergantung pada nilai T yang digunakan.

2.7. Pendeteksi Tepi (Edge Detection)

Deteksi tepi (*edge detection*) merupakan salah satu proses pengolahan yang sering dibutuhkan pada analisa citra digital. Proses ini bertujuan untuk meningkatkan penampakan garis tepi pada suatu citra digital.[5]

Tujuan lain *edge detection* yaitu untuk identifikasi suatu daerah pada citra digital dimana terdapat perubahan besar dalam intensitas. Dengan deteksi tepi pada suatu sistem pemrosesan citra (berbasis komputer) akan menemukan tanda-tanda permukaan atau garis bentuk yang timbul dari obyek.[5]

2.7.1. Deteksi Sobel

Deteksi tepi operator Sobel diperkenalkan oleh Irwin Sobel pada tahun 1970. Kelebihan dari operator sobel ini adalah kemampuannya untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi. [6]

Operator sobel identik dengan bentuk matrik 3x3 atau jendela ukuran 3x3 *pixel*, dengan Gx dan Gy dihitung menggunakan kernel (*mask*) seperti pada gambar 2 berikut.[4]

$$Gx = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad Gy = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Gambar 2. Matrik Operator Sobel
(Sumber : Putra, 2010)

Kombinasi kedua kernel tersebut digunakan untuk menghitung jarak absolut sebuah gradient, namun kombinasi tersebut bisa juga diaplikasikan secara terpisah untuk menghitung masing-masing proses vertikal dan horizontal. Untuk menghitung jarak gradient, digunakan persamaan berikut :[4]

$$G = \sqrt{Gx^2 + Gy^2} \dots\dots\dots (12)$$

2.8. Ukuran Pemusatan Data

2.8.1. Mean (Rata-rata)

Kumpulan data yang digunakan untuk menghitung *mean* atau rata-rata hitung adalah kumpulan data kuantitatif. Mean dari sekumpulan data kuantitatif dinyatakan dengan simbol \bar{x} (dibaca x bar). Rumus untuk *mean* atau rata-rata hitung dari data tunggal adalah sebagai berikut.[7]

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + x_3 \dots + x_n) \quad \text{atau} \quad \bar{x} = \frac{\sum x_i}{n} \dots\dots\dots (13)$$

2.9. Pencocokan Pola

2.9.1. Koefisien Korelasi

Metode pencocokan pola ini adalah salah satu metode terapan teknik konvolusi. Teknik konvolusi dilakukan dengan mengkombinasikan citra masukan dengan citra sumber acuan, hingga akan didapatkan nilai koefisien korelasi yang besarnya antara -1 dan +1. Saat nilai koefisien korelasi semakin mendekati +1, bisa dikatakan citra masukan semakin sama (mirip) dengan citra acuannya. Rumus yang digunakan adalah : [8]

$$r = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (x_{ij} - \bar{x})(y_{ij} - \bar{y})}{\sqrt{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (x_{ij} - \bar{x})^2 \cdot \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (y_{ij} - \bar{y})^2}} \dots\dots\dots (14)$$

Keterangan :

- r : Koefisien korelasi
- x : Citra acuan (*template*)
- \bar{x} : Nilai rata-rata citra acuan
- y : Nilai rata-rata citra masukan
- y : Citra masukan
- M, N : Jumlah *pixel* citra

2.9.2. Euclidean Distance (Jarak Euclidean)

Metode pencocokan pola dengan jarak *Euclidean* ini dilakukan dengan cara menghitung akar dari kuadrat perbedaan 2 vektor, yang mana semakin kecil jarak, maka vektor masukan dan vektor acuan akan semakin mirip. Rumus dari jarak *Euclidean* adalah sebagai berikut. [4]

$$d_{ij} = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2} \dots \dots \dots (15)$$

Keterangan :

d_{ij} = jarak *Euclidean*

x_{ik} = Citra Uji

x_{jk} = Citra Acuan

3. METODOLOGI PENELITIAN

Penelitian ini menggunakan metodologi penelitian yang mencakup studi literatur, kemudian dilakukan analisa kebutuhan akan komponen-komponen yang dibutuhkan dalam pembuatan aplikasi klasifikasi jenis kamboja jepang (*Adenium sp.*) berdasarkan citra mahkota bunga hasil deteksi tepi dan ekstraksi fitur warna, selanjutnya perancangan sistem, kemudian dilakukan tahap implementasi dan pengujian, serta analisis. Tahap terakhir adalah kesimpulan dan saran.

4. PERANCANGAN SISTEM

Perancangan pada penelitian ini meliputi perancangan *database*, tahap perancangan perangkat keras, dan tahap perancangan perangkat lunak.

4.1. Perancangan Database

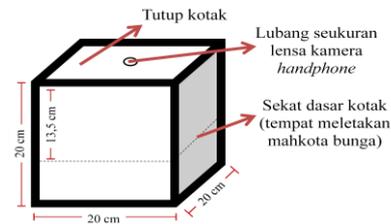
Database pada penelitian ini menggunakan *database* yang disediakan oleh Matlab yang berekstensi *.mat. Masing-masing jenis bunga memiliki 2 *database*, yaitu *database* yang berisikan data nilai *pixel* citra hue dan frekuensi intensitas warna pada histogram hue (contoh nama *database* : 'histocabangseribu.mat'), serta *database* yang berisikan nilai *mean* (rata-rata) citra hasil deteksi sobel (contoh nama *database*: 'sobelcabangseribu.mat').

4.2. Perancangan Perangkat Keras

Perangkat utama yang dibutuhkan dalam penelitian ini adalah *laptop* Asus tipe X45M dengan spesifikasi sebagai berikut :

- Processor : Intel Celeron N2830
- RAM : 2 GB
- Harddisk : 500 GB

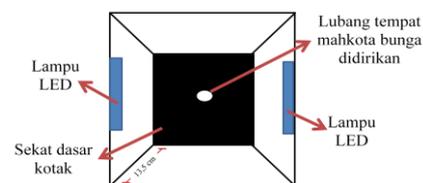
Perangkat pendukung yang dibutuhkan dalam penelitian ini adalah kotak berukuran 20x20x20cm dimana bagian dalam kotak dicat warna hitam, serta dilengkapi dengan pencahayaan yang telah diatur dan jarak kamera *handphone* ke obyek ±13,5 cm. Perangkat pendukung lainnya yaitu kabel data *USB* dan *handphone* Evercoss A65A, dimana kamera *handphone* ini yang nantinya akan digunakan untuk pengambilan citra dari mahkota bunga. Perancangan perangkat keras (kotak) dapat dilihat pada gambar 3 dan gambar 4.



Gambar 3. Kotak Tertutup

Gambar 3 merupakan rancangan kotak dalam keadaan tertutup. Rincian gambar 3 adalah sebagai berikut.

- a. Kotak berukuran panjang 20cm, lebar 20cm dan tinggi 20cm. Pada bagian dalam kotak terdapat sekat dasar kotak, yang mana jarak dari tutup kotak ke sekat tersebut ± 13,5cm.
- b. Pada bagian tutup kotak (tepat dibagian tengah) terdapat lubang seukuran lensa kamera *handphone* (±0,5cm). Hal tersebut dimaksudkan agar pada saat pengambilan gambar, posisi kamera *handphone* dapat konsisten.



Gambar 4. Kotak Terbuka

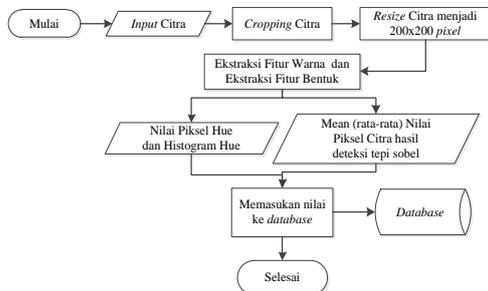
Gambar 4 merupakan rancangan kotak dalam keadaan terbuka. Rincian gambar 4 adalah sebagai berikut.

- a. Jarak dari tutup kotak ke sekat dasar kotak yaitu ±13,5cm, dimana pada bagian tengah sekat terdapat lubang berdiameter

- 1cm berfungsi sebagai tempat mendirikan bunga agar tegak lurus kearah kamera.
- Warna dasar kotak digunakan warna hitam untuk menghindari adanya efek bayangan mahkota bunga yang terlalu berlebihan dan dapat mempengaruhi hasil dari proses ekstraksi fitur citra.
 - Cahaya penerangan dalam kotak digunakan 12 buah lampu LED, yang dimaksudkan untuk menerangi ruang di dalam kotak agar obyek mendapatkan pencahayaan yang cukup baik dengan intensitas cahaya yang tidak terlalu terang saat pengambilan citra dengan kamera *handphone*.

4.3. Perancangan Perangkat Lunak

Rancangan perangkat lunak pada penelitian ini dibuat dalam bentuk *flowchart*, meliputi *flowchart* latihan dapat dilihat pada Gambar 5, *flowchart* ekstraksi fitur bentuk pada Gambar 7, *flowchart* ekstraksi fitur warna pada Gambar 8, *Flowchart* uji pada Gambar 10, *flowchart* uji kemiripan menggunakan koefisien korelasi pada Gambar 11, dan *flowchart* uji kemiripan menggunakan jarak *Eulidean* pada Gambar 12.

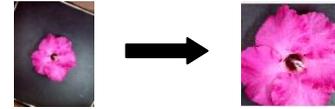


Gambar 5. *Flowchart* Latihan

Pada gambar 5 dapat dilihat *flowchart* latihan aplikasi klasifikasi jenis bunga *Adenium sp.*. Penjelasan rincian dari masing-masing tahapan akan dijelaskan sebagai berikut :

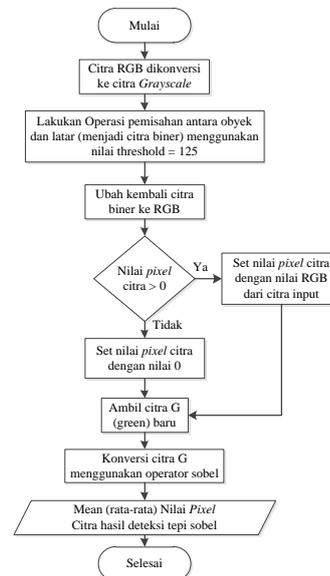
- Proses dimulai dari *input* citra.
- Proses *cropping* citra yang menghasilkan citra berukuran 1280x1280 *pixel*. Proses *cropping* pada sistem ini menggunakan skala yang telah ditetapkan, sehingga terbatas hanya untuk jenis *Adenium sp.* yang ukuran mahkota bunganya maksimal ± 6 cm. Untuk *Adenium sp.* jenis Ipanema yang memiliki mahkota bunga berukuran agak besar, pada saat *cropping*, obyek bunga akan terkena *crop* juga, hasil *crop*

dapat dilihat pada gambar 6 berikut.



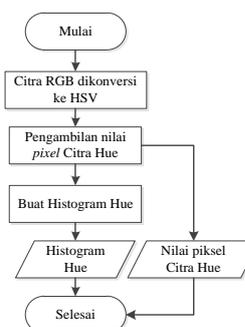
Gambar 6. Hasil *Cropping* Citra Pada Bunga Bermahkota Besar (Ipanema)

- Proses *resize* citra menjadi 200x200 *pixel*.
- Proses ekstraksi fitur warna yang akan menghasilkan nilai *pixel hue* dan histogram hue, serta ekstraksi fitur bentuk yang menghasilkan nilai *mean* (rata-rata) *pixel* citra hasil deteksi tepi sobel.
- Data nilai hasil ekstraksi fitur warna dan hasil ekstraksi fitur bentuk kemudian disimpan di *database* yang mana nantinya akan digunakan pada saat pengujian.
- Proses selesai.



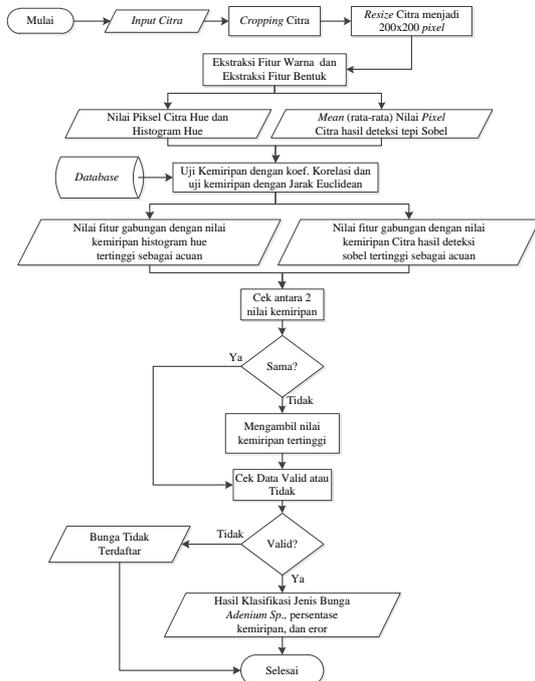
Gambar 7. *Flowchart* Ekstraksi Fitur Bentuk

Gambar 7 merupakan gambaran *flowchart* ekstraksi fitur bentuk yang akan menghasilkan data berupa nilai *mean* (rata-rata) *pixel* citra deteksi tepi sobel.



Gambar 8. *Flowchart* Ekstraksi Fitur Warna

Gambar 8 adalah gambar *flowchart* ekstraksi fitur warna yang menghasilkan data nilai *pixel* citra hue dan histogram hue.

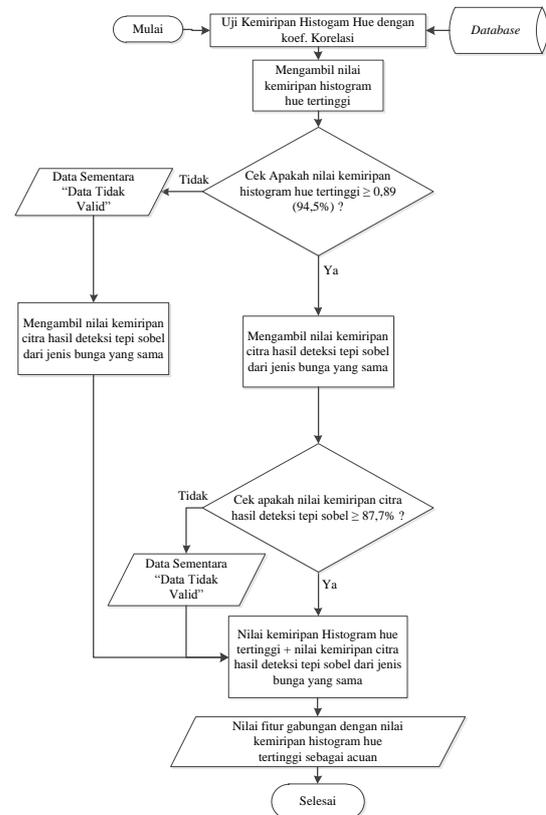


Gambar 9. Flowchart Uji

Gambar 9 berisi *flowchart* uji sistem menghasilkan keluaran berupa informasi klasifikasi jenis bunga *Adenium sp.* Penjelasan rincian tiap tahap dijelaskan sebagai berikut :

1. Proses dimulai dari *input* citra uji.
2. Proses *crop* citra yang menghasilkan citra berukuran 1280x1280 *pixel*, kemudian proses *resize* citra menjadi 200x200 *pixel*.
3. Proses ekstraksi fitur warna menghasilkan nilai *pixel* hue dan histogram hue, dan ekstraksi fitur bentuk menghasilkan nilai *mean pixel* citra hasil deteksi tepi sobel.
4. Data nilai hasil ekstraksi fitur warna dan hasil ekstraksi fitur bentuk diuji kemiripan dengan koefisien korelasi untuk fitur warna (*flowchart* ditampilkan pada gambar 11), dan dengan jarak *euclidean* untuk fitur bentuk (*flowchart* ditampilkan pada gambar 12). Proses uji kemiripan menghasilkan data 'nilai fitur gabungan dengan nilai kemiripan histogram hue tertinggi sebagai acuan' dan 'nilai fitur gabungan dengan nilai kemiripan citra hasil deteksi sobel tertinggi sebagai acuan'.
5. Pengecekan 2 nilai kemiripan (pada poin 4). dilakukan sebagai berikut :
 - Jika sama, proses lanjut ke poin 6.

- Jika tidak sama, sistem akan mengambil nilai tertinggi, dan proses lanjut ke poin 6.
6. Pengecekan data valid atau tidak dilakukan sebagai berikut :
 - Jika data valid, maka program akan mengeluarkan keluaran berupa informasi hasil klasifikasi bunga *Adenium sp.*, persentase kemiripan, dan eror.
 - Jika tidak, maka program mengeluarkan keluaran berupa informasi "Bunga Tidak Terdaftar".
 7. Proses selesai.



Gambar 10. Flowchart Uji Kemiripan Menggunakan Koefisien Korelasi

Penjelasan rincian dari tiap tahapan pada Gambar 11 akan dijelaskan sebagai berikut :

1. Proses mulai.
2. Sistem melakukan proses uji kemiripan histogram hue dengan metode koefisien korelasi antara citra uji dan citra latihan yang tersimpan di *database*.
3. Sistem mengambil nilai kemiripan histogram hue tertinggi.
4. Sistem mengecek nilai kemiripan tertinggi tersebut sebagai berikut :

Nilai kemiripan histogram hue tertinggi harus $\geq 0,89$ (94,5%). Proses selanjutnya adalah pengecekan sebagai berikut :

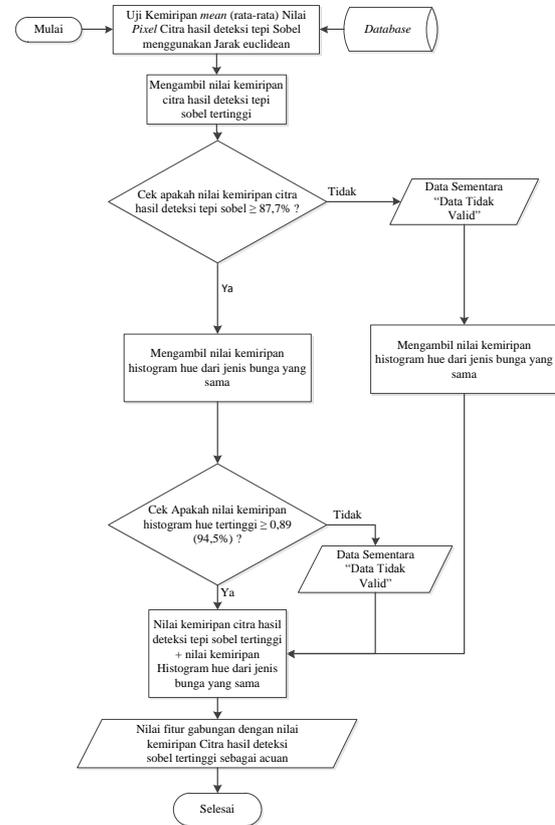
- a. Jika tidak memenuhi syarat, maka sistem akan menyimpan data sementara berupa informasi “Data Tidak Valid”. Dilanjutkan dengan proses mendapatkan nilai fitur gabungan tertinggi dengan nilai kemiripan histogram hue tertinggi sebagai acuan, kemudian mengambil nilai kemiripan citra hasil deteksi tepi sobel dari jenis bunga yang sama. Contohnya apabila nilai kemiripan histogram hue tertinggi adalah bunga jenis ‘A’, maka sistem akan memanggil nilai kemiripan citra hasil deteksi sobel dari bunga ‘A’ pula. Proses dilanjutkan dengan penjumlahan antara nilai kemiripan histogram hue tertinggi (dalam kasus ini bunga ‘A’) dengan nilai kemiripan citra hasil deteksi tepi sobelnya, dan didapat data berupa ‘nilai fitur gabungan dengan nilai kemiripan histogram hue tertinggi sebagai acuan’. Proses lanjut ke poin 5.

- b. Jika memenuhi syarat, proses dilanjutkan dengan mendapatkan nilai fitur gabungan tertinggi dengan nilai kemiripan histogram hue tertinggi sebagai acuan, kemudian mengambil nilai kemiripan citra hasil deteksi tepi sobel dari jenis bunga yang sama, contohnya apabila nilai kemiripan histogram hue tertinggi adalah bunga jenis ‘A’, maka sistem akan memanggil nilai kemiripan citra hasil deteksi sobel dari bunga ‘A’ pula. Selanjutnya cek nilai kemiripan citra hasil deteksi tepi sobel yang dipanggil apakah $\geq 87,7\%$, pengecekan dilakukan sebagai berikut :

- Jika tidak, maka sistem akan menyimpan data sementara berupa informasi “Data Tidak Valid”. Proses dilanjutkan dengan penjumlahan nilai kemiripan histogram hue tertinggi (dalam kasus ini bunga ‘A’) dengan nilai kemiripan citra hasil deteksi tepi sobelnya.
- Jika memenuhi, dilanjutkan dengan penjumlahan nilai kemiripan histogram hue tertinggi (dalam kasus ini bunga ‘A’) dengan nilai kemiripan

citra hasil deteksi tepi sobelnya, dan didapat data ‘nilai fitur gabungan dengan kemiripan histogram hue tertinggi sebagai acuan’.

5. Proses selesai.



Gambar 11. Flowchart Uji Kemiripan Menggunakan Jarak Euclidean

Penjelasan rincian dari tiap tahapan pada gambar 11 akan dijelaskan sebagai berikut :

1. Proses dimulai.
2. Sistem melakukan proses uji kemiripan citra hasil deteksi sobel dengan metode jarak *Euclidean*.
3. Sistem mengambil nilai kemiripan citra hasil deteksi sobel tertinggi.
4. Sistem mengecek nilai kemiripan tertinggi tersebut sebagai berikut :
Nilai kemiripan citra hasil deteksi tepi sobel tertinggi harus $\geq 87,7\%$. Pengecekan yang dilakukan sebagai berikut :
 - a. Jika tidak memenuhi syarat, maka sistem akan menyimpan data sementara berupa informasi “Data Tidak Valid”. Proses selanjutnya mendapatkan nilai fitur gabungan tertinggi dengan nilai kemiripan citra hasil deteksi tepi sobel tertinggi sebagai acuan, kemu-

dian mengambil nilai kemiripan histogram hue dari jenis bunga yang sama, contohnya apabila nilai kemiripan citra hasil deteksi tepi sobel tertinggi adalah bunga jenis 'B', maka sistem akan memanggil nilai kemiripan histogram hue dari bunga 'B' pula. Proses dilanjutkan dengan penjumlahan nilai kemiripan citra hasil deteksi tepi sobel tertinggi (dalam kasus ini bunga 'B') dengan nilai kemiripan histogram hue nya, dan didapat data berupa 'nilai fitur gabungan dengan nilai kemiripan citra hasil deteksi sobel tertinggi sebagai acuan'. Proses lanjut ke poin 5.

- b. Jika memenuhi syarat, sistem lanjut mendapatkan nilai fitur gabungan tertinggi dengan nilai kemiripan citra hasil deteksi tepi sobel tertinggi sebagai acuan, kemudian mengambil nilai kemiripan histogram hue dari jenis bunga yang sama, contoh jika nilai kemiripan citra hasil deteksi sobel tertinggi adalah bunga jenis 'B', maka sistem akan memanggil nilai kemiripan histogram hue dari bunga 'B' pula, lalu cek nilai kemiripan histogram hue yang dipanggil apakah $\geq 0,89$ (94,5%), pengecekan sebagai berikut :

- Jika tidak memenuhi syarat, maka sistem akan menyimpan data sementara berupa informasi "Data Tidak Valid". Proses dilanjutkan dengan penjumlahan antara nilai kemiripan citra hasil deteksi tepi sobel tertinggi (dalam kasus ini bunga 'B') dengan nilai kemiripan histogram hue nya.
- Jika memenuhi syarat, proses dilanjutkan dengan penjumlahan antara nilai kemiripan citra hasil deteksi tepi sobel tertinggi (dalam kasus ini adalah bunga 'B') dengan nilai kemiripan histogram hue nya, dan didapat data berupa 'nilai fitur gabungan dengan nilai citra hasil deteksi sobel tertinggi sebagai acuan'.

5. Proses selesai.

4.4. Perancangan User-Interface

User-Interface adalah perangkat lunak yang berfungsi sebagai penghubung antara

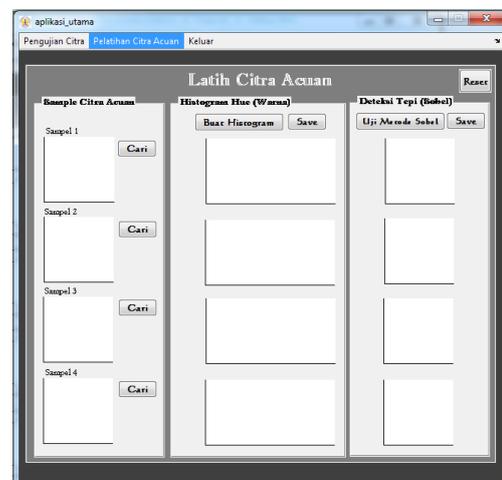
pengguna dan sistem. Desain aplikasi antarmuka klasifikasi jenis bunga *Adenium sp.* ini menggunakan aplikasi Matlab 7.6.0 (R2008a), meliputi *user-interface* aplikasi pembuka pada gambar 13, *user-interface* 'Latih Citra Acuan' gambar 14, dan *user-interface* 'Penguujian Citra' di gambar 15.



Gambar 12. User-interface Aplikasi Pembuka

Keterangan :

1. Tombol 'Mulai Aplikasi': berfungsi untuk memanggil form (aplikasi) 'aplikasi_utama', dan form 'aplikasi_awal' akan ditutup.

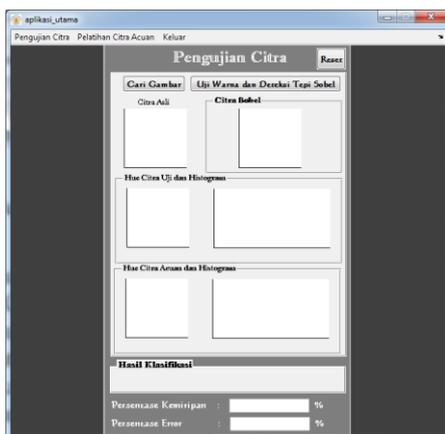


Gambar 13. User-interface 'aplikasi_utama' (Latih Citra Acuan)

Keterangan :

1. Menu 'Penguujian Citra' : berfungsi untuk memanggil *panel* 'Penguujian Citra'.
2. Menu 'Pelatihan Citra Acuan' : berfungsi untuk memanggil *panel* 'Pelatihan Citra Acuan'.
3. Menu 'Keluar' : berfungsi untuk menghentikan dan menutup aplikasi.
4. Tombol "Reset" : berfungsi untuk *reset panel* 'Penguujian Citra'
5. Tombol 'Cari' (ke-1) : untuk mencari *file*

- citra yang akan menjadi sampel latihan pertama dan ditampilkan di *axes* 'sampel 1'.
6. Tombol 'Cari' (ke-2) : untuk mencari *file* citra yang akan dijadikan sampel latihan kedua, dan ditampilkan di *axes* 'sampel 2'.
 7. Tombol 'Cari' (ke-3) : untuk mencari *file* citra yang akan menjadi sampel latihan ketiga, dan ditampilkan di *axes* 'sampel 3'.
 8. Tombol 'Cari' (ke-4) : untuk mencari *file* citra yang akan menjadi sampel latihan keempat, dan ditampilkan di *axes* 'sampel 4'.
 9. Tombol 'Buat Histogram' : untuk membuat histogram hue dari tiap sampel, dimana histogram hue tiap sampel akan ditampilkan pada *axes* sesuai urutan sampel.
 10. Tombol 'Save' (samping tombol 'Buat Histogram') : untuk menyimpan histogram hue dari tiap sampel dalam satu *database* (file berekstensi *.mat).
 11. Tombol 'Uji Metode Sobel' : untuk konversi citra sampel dari citra RGB ke biner dengan metode sobel, yang nantinya citra hasil deteksi tepi sobel dari tiap sampel akan tampil di *axes* sesuai urutan sampel.
 12. Tombol 'Save' (samping tombol 'Uji Metode Sobel') : menyimpan citra hasil deteksi tepi sobel dari tiap sampel ke dalam *database* (file berekstensi *.mat).



Gambar 14. *User-interface* 'aplikasi_utama' (Penguujian Citra)

Keterangan :

1. Menu 'Penguujian Citra' : berfungsi untuk memanggil *panel* 'Penguujian Citra'.
2. Menu 'Pelatihan Citra Acuan' : berfungsi untuk memanggil *panel* 'Pelatihan Citra Acuan'.

3. Menu 'Keluar' : berfungsi untuk menghentikan dan menutup aplikasi.
4. Tombol "Reset" : berfungsi untuk *reset panel* 'Penguujian Citra'.
5. Tombol 'Cari Gambar' : untuk mencari *file* citra yang akan dijadikan citra uji, dan akan ditampilkan pada *axes* 'Citra Asli'.
6. Tombol 'Uji Warna dan Deteksi Tepi Sobel' : untuk membuat histogram hue dan menampilkan citra hue dari citra uji serta histogramnya di *axes* yang terdapat pada *panel* 'Hue Citra Uji dan Histogram', kemudian mengkonversi citra uji dari citra RGB ke biner menggunakan metode sobel, dan menampilkan citra hasil deteksi tepi sobel ke *axes* yang terdapat pada *panel* 'Citra Sobel'. Selanjutnya, aplikasi akan mencari nilai kemiripan tertinggi untuk klasifikasi. Jika klasifikasi berhasil, maka citra hue dan histogram bunga dari *database* yang memiliki tingkat kemiripan tertinggi akan ditampilkan di *axes* pada *panel* 'Hue Citra Acuan dan Histogram'. Selanjutnya, pada *panel* 'Hasil Klasifikasi' akan tampil nama bunga hasil klasifikasi. Pada 'Persentase Kemiripan' tampil berapa persen kemiripan antara citra uji dengan citra yang terdapat di *database*, dan pada 'Error' tampil berapa persen eror dari kemiripan. Tetapi jika klasifikasi gagal, maka pada *axes* di *panel* 'Hue Citra Acuan dan Histogram' akan muncul gambar tanda seru, kemudian pada *panel* 'Hasil Klasifikasi' akan tampil teks "Bunga Tidak Terdaftar", pada 'Persentase Kemiripan' akan tampil nilai '0', dan pada 'Error' akan tampil nilai '100'.

5. IMPLEMENTASI DAN PENGUJIAN

Implementasi dan pengujian sistem meliputi proses akusisi data, pelatihan sistem, dan proses pengujian sistem.

5.1. Proses Akusisi Data

Sistem akusisi data berkaitan dengan pengambilan data yang dilakukan dengan menggunakan kamera *handphone* 8 megapixel dan alat bantu kotak yang berukuran 20x20x20cm, dimana *background*

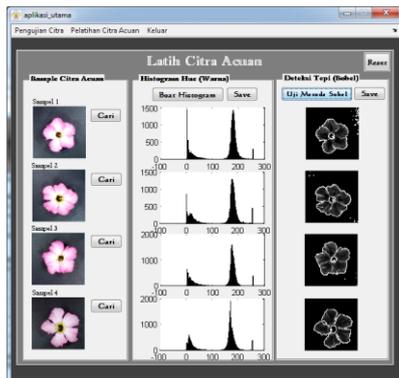
kotak berwarna hitam, serta dengan pencahayaan dalam kotak yang telah diatur. Jarak dari kamera ke obyek ± 15 cm (jarak tetap), dan pengambilan citra dilakukan tegak lurus terhadap obyek, dan melakukan tes fokus pada layar *handphone*, hal ini berpengaruh terhadap pencahayaan saat pengambilan citra agar terfokus pada obyek bunga.



Gambar 15. Pengambilan Citra Bunga

5.2. Proses Pelatihan Sistem

Proses ‘Latih Citra Acuan’ pada aplikasi klasifikasi jenis bunga *Adenium sp.* (sebagai contoh pelatihan bunga *Adenium sp.* jenis Cabang Seribu) ditampilkan pada gambar 16 berikut.



Gambar 16. Tampilan Antarmuka Proses ‘Latih Citra Acuan’

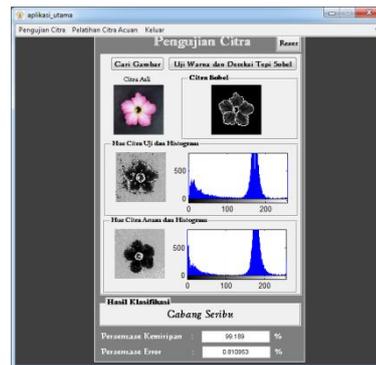
Tombol ‘Save’ pada *panel* ‘Histogram Warna (Hue)’ digunakan untuk menyimpan data citra hue dan histogram hue dari keempat sampel citra acuan, yang mana pelatihan citra acuan pada gambar 16 data histogram hue akan disimpan dengan nama “histocabangseribu”, *file* yang tersimpan akan berekstensi *.mat.

Tombol ‘Save’ pada *panel* ‘Deteksi Tepi (Sobel)’ digunakan untuk menyimpan data nilai *mean* citra hasil deteksi sobel dari keempat sampel citra acuan, dimana sesuai gambar 16 data akan disimpan dengan nama “sobelcabangseribu”, *file* yang tersimpan akan berekstensi *.mat.

5.3. Proses Pengujian Sistem

5.3.1. Pengujian Sistem untuk Bunga Terdaftar

Pada proses pengujian terdapat tombol ‘Cari Gambar’ yang berfungsi memilih *file* citra yang akan diuji. Tiap jenis bunga memiliki 3 *file* citra yang akan digunakan untuk pengujian seperti pada gambar 17 berikut.



Gambar 17. Tampilan Proses Pengujian Citra untuk Bunga Terdaftar

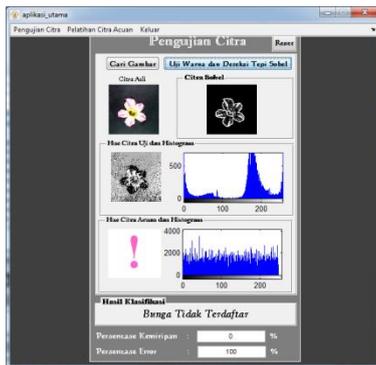
Tombol ‘Uji Warna dan Deteksi Tepi Sobel’ digunakan dalam pengujian. Apabila diklik akan menampilkan citra hasil deteksi sobel dari citra uji pada *panel* ‘Citra Sobel’, serta citra hue dan histogram hue dari citra uji ditampilkan pada *axes* di *panel* ‘Hue Citra Uji dan Histogram’. Ketika sistem mendapatkan nilai hasil uji kemiripan, sistem akan menampilkan citra hue dan histogram hue dari *database* yang memiliki persentase kemiripan tertinggi, ditampilkan pada *axes* di *panel* ‘Hue Citra Acuan dan Histogram’, kemudian akan tampil nama bunga “Cabang Seribu” (seperti pada gambar 17) pada *panel* ‘Hasil Klasifikasi’, akan ditampilkan pula persentase kemiripan “99,189%”, serta persentase *error* sebesar “0,81%”.

5.3.2. Pengujian Sistem untuk Bunga Tidak Terdaftar

Sampel yang akan digunakan untuk pengujian jenis bunga yang tidak terdaftar adalah 4 buah sampel. Gambar 18 menampilkan pengujian untuk jenis bunga Chiara (tidak terdaftar di *database*).

Proses pengujian untuk ‘Bunga Tidak Terdaftar’ melewati tahap-tahap seperti yang telah dijelaskan pada pengujian sebelumnya. Setelah didapatkan hasil dengan presentase kemiripan tertinggi, yang mana tidak

memenuhi salah satu atau kedua syarat (kemiripan histogram hue $\geq 0,89$ (94,5%) dan kemiripan citra hasil deteksi tepi sobel $\geq 87,7$)) yang telah ditetapkan, maka selanjutnya sistem akan menampilkan gambar tanda seru (!) dan histogram acak pada axes di panel 'Hue Citra Acuan dan Histogram'. Kemudian akan tampil hasil "Bunga Tidak Terdaftar" pada panel 'Hasil Klasifikasi', pada 'Persentase Kemiripan' akan tampil nilai '0', dan pada 'Error' akan tampil nilai '100'.



Gambar 18. Tampilan Proses Penguji Citra untuk Bunga Tidak Terdaftar

Perhitungan persentase keberhasilan pengujian secara umum serta persentase keberhasilan klasifikasi berdasarkan S_{max} dan H_{max} diberikan sebagai berikut :

1. Persentase keberhasilan pengujian secara umum.

Dari 25 kali pengujian, terdapat 24 kali pengujian valid, dan terdapat 1 kali pengujian tidak valid.

$$\% \text{ keberhasilan klasifikasi pengujian} = \frac{24}{25} \times 100 = 96 \%$$

$$\% \text{ kegagalan klasifikasi pengujian} = \frac{1}{25} \times 100 = 4 \%$$

2. Dari 24 pengujian yang valid, persentase keberhasilan klasifikasi berdasarkan S_{max} dan H_{max} adalah sebagai berikut :

a. Klasifikasi berdasarkan S_{max} terdapat sebanyak 8 kali dari 24 kali pengujian yang valid.

$$\% \text{ klasifikasi berdasarkan } S_{max} = \frac{8}{24} \times 100 = 33,33 \%$$

b. Klasifikasi berdasarkan H_{max} terdapat sebanyak 6 kali dari 24 kali pengujian yang valid.

$$\% \text{ klasifikasi berdasarkan } H_{max} = \frac{6}{24} \times 100 = 25 \%$$

c. Klasifikasi berdasarkan $H_{max} = S_{max}$ terdapat sebanyak 10 kali dari 24 kali pengujian yang valid.

$$\% \text{ klasifikasi berdasarkan } (H_{max} = S_{max})$$

$$(H_{max} = S_{max}) = \frac{10}{24} \times 100 = 41,67 \%$$

6. KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan pengujian dan analisis terhadap aplikasi Klasifikasi Jenis Bunga Kamboja Jepang (*Adenium sp.*) Berdasarkan Citra Mahkota Menggunakan Ekstraksi Fitur Warna dan Deteksi Tepi yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut :

1. Nilai histogram hue dari citra mahkota bunga didapatkan dari konversi citra RGB menjadi citra HSV, yang mana citra H akan digunakan dalam pembuatan histogram Hue, sedangkan untuk citra hasil deteksi tepi sobel didapatkan dari konversi citra RGB (*input*) menjadi citra *grayscale*. Tahap selanjutnya terjadi proses pengembangan untuk memisahkan citra obyek dan latar (menjadi citra biner), kemudian adanya proses pengaturan ulang citra, yaitu untuk *pixel* bernilai 1 diganti dengan nilai citra G dari citra *input*, untuk nilai 0 diatur tetap bernilai 0. Tahapan terakhir yaitu citra dikonversi dengan deteksi tepi sobel, dan menghasilkan citra hasil deteksi tepi sobel.
2. Koefisien korelasi pada penelitian ini digunakan untuk menghitung nilai kemiripan histogram hue. Apabila nilai koefisien korelasi mendekati 1, maka histogram hue masukan semakin sama (mirip) dengan histogram hue acuan, dan apabila nilai koefisien korelasi mendekati -1, maka histogram hue masukan semakin berbeda (tidak mirip) dengan histogram hue acuan.
3. Dari 7 jenis bunga *Adenium sp.* (dimana masing-masing jenis digunakan 3 sampel pengujian) menghasilkan 21 pengujian, ditambah 4 sampel pengujian bunga *Adenium sp.* (selain 7 jenis obyek bunga yang telah dilatih). Persentase keberhasilan sistem dalam mengklasifikasikan jenis bunga *Adenium sp.* adalah sebesar 96%, dan persentase kegagalan sistem dalam mengklasifikasikan jenis bunga *Adenium*

sp. adalah sebesar 4%, sedangkan persentase keberhasilan untuk klasifikasi *Adenium sp.* (dari 25 pengujian, didapatkan 24 pengujian yang dinyatakan valid/berhasil) berdasarkan fitur gabungan yaitu sebagai berikut :

- a. Berdasarkan fitur gabungan dengan nilai kemiripan histogram hue tertinggi (H_{max}) sebagai acuan adalah sebesar 25%.
- b. Berdasarkan fitur gabungan dengan nilai kemiripan citra hasil deteksi tepi sobel tertinggi (S_{max}) sebagai acuan adalah sebesar 33,33%.
- c. Berdasarkan fitur gabungan dengan nilai kemiripan histogram hue tertinggi (H_{max}) = nilai kemiripan citra hasil deteksi tepi sobel tertinggi (S_{max}) adalah sebesar 41,67%.

Sesuai persentase keberhasilan yang telah disebutkan, maka klasifikasi menggunakan fitur gabungan dengan metode sobel dan histogram hue dapat digunakan di dalam penelitian ini.

6.2. Saran

Adapun saran untuk penelitian kedepannya agar lebih baik adalah sebagai berikut :

1. Penelitian klasifikasi ini menggunakan bahasa pemrograman Matlab, pada penelitian selanjutnya dapat dicoba menggunakan bahasa pemrograman yang berbeda.
2. Proses pengambilan citra obyek menggunakan latar tertentu dengan alat bantu kotak dan pencahayaan yang telah diatur, oleh sebab itu untuk penelitian lebih lanjut dapat dicoba pengambilan citra yang dilakukan dengan latar sembarang dan tanpa penggunaan kotak.

DAFTAR PUSTAKA

- [1] Sugih, O., 88 *Variasi Adenium Agar Rajin Berbunga*. Depok: Niaga Swadaya, 2002.
- [2] Fahmi, "Perancangan Algoritma Pengolahan Citra Mata Menjadi Citra Polar Iris Sebagai Bentuk Antara Sistem Biometrik," *Karya Ilmiah Teknik Elektro, Universitas Sumatra Utara Medan*, 2007.
- [3] Rakhmawati, R. P., "Sistem Deteksi Jenis Bunga Menggunakan Nilai Hsv Dari

Citra Mahkota Bunga," *Tugas Akhir Teknik Informatika Universitas Stikubank Semarang*, 2013.

- [4] Putra, D., *Pengolahan Citra Digital*. Yogyakarta: Andi Offset, 2010.
- [5] Kalo A. Y., "Information Retrieval Berbasis Citra Berdasarkan Logo Perusahaan Dengan Menggunakan Metode Edge Matching Dan Euclidean Distance," *Tugas Akhir Teknik dan Ilmu Komputer UNIKOM Bandung*, 2014.
- [6] Sigit, R., Basuki, A., Ramadijanti, N., & Pramadihanto, D., *Step By Step Pengolahan Citra Digital*. Yogyakarta: Andi Offset, 2007.
- [7] Yusuf, M., & Wijaya, R., *Matematika Kelompok Sosial, Administrasi Perkantoran, dan Akuntansi Kelas XI*, 1st ed. Bandung: Grafindo Media Pratama, 2008.
- [8] Bowo, S. A., Hidayatno, A., & Isnanto, R. R., "Analisis Deteksi Tepi Untuk Mengidentifikasi Pola Daun," *Thesis Teknik Elektro Fakultas Teknik Universitas Diponegoro*, 2011.