

BAB IV

HASIL DAN PEMBAHASAN

Dalam bab hasil dan pembahasan, akan dijelaskan mengenai implementasi penelitian mulai dari akusisi data citra, pre-proses citra, segmentasi citra, ekstraksi ciri citra, dan pengujian.

4.1. Akuisisi Data Citra

Contoh akuisisi data citra disajikan pada Tabel 3.3. Citra yang digunakan untuk contoh tahap-tahap penelitian ditunjukkan pada Gambar 4.1.



Gambar 4. 1 Citra padi

4.2. Pre processing Citra

Tahap *pre processing* memiliki tiga tahap, yaitu merubah citra RGB menjadi citra *Grayscale*, merubah kontras dan cahaya, dan *filtering* menggunakan teknik media *filtering*.

1. Grayscale Citra Daun Padi

Grayscale bertujuan untuk merubah citra RGB menjadi citra abu agar proses perhitungan citra lebih mudah dilakukan. Secara matematis, grayscale merupakan pencarian rerata dari ketiga komponen *red*, *green*, dan *blue*. Di bawah berikut adalah contoh perhitungan dari operasi *grayscale*. Gambar 4.2 adalah contoh matriks pada sebuah citra.

R = 90	R = 70
G = 90	G = 80
B = 90	B = 90
R = 80	R = 70
G = 50	G = 30
B = 80	B = 20

Gambar 4. 2 Contoh Matriks Citra

Misal, matriks 2 x 2 memiliki komponen *red*, *green*, *blue* seperti di atas, operasi yang dilakukan adalah menjumlah ketiga komponen tersebut, lalu membagi hasil penjumlahan dengan angka tiga (jumlah komponen warna). Operasi perhitungan *grayscale* secara rinci dijelaskan pada operasi berikut :

$$f_{0(1,1)} = \frac{f_{iR(1,1)} + f_{iG(1,1)} + f_{iB(1,1)}}{3} = \frac{90+90+90}{3} = 90$$

Hasil dari perhitungan kemudian dimasukkan ke dalam masing-masing matriks seperti pada Gambar 4.3.

90	80
70	40

Gambar 4. 3 Matriks Hasil *Grayscale*

Di bawah berikut merupakan contoh data citra yang dikonversi dari citra RGB menjadi citra *Grayscale* atau citra abu.



Gambar 4. 4 Proses *grayscale* (a) Citra daun padi (b) Citra daun padi setelah dikonversi

2. Pengaturan Kecerahan dan Kontras Citra Daun Padi

Citra yang sudah dikonversi menjadi citra *Grayscale* kemudian akan diatur kecerahan dan kontrasnya untuk menonjolkan informasi yang ingin didapatkan pada objek citra. Gambar 4.5 adalah contoh matriks *Grayscale*.

90	80
70	40

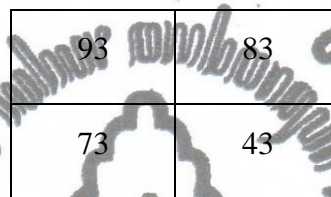
Gambar 4. 5 Contoh Matriks *Grayscale*

Matriks di atas merupakan contoh matriks *Grayscale* 2 x 2 yang akan dimodifikasi kecerahan dan kontrasnya. Misal, citra akan ditambah kecerahannya sebanyak 3 level, maka di bawah berikut merupakan contoh penghitungan modifikasi kecerahan.

$$f(x,y)' = f(x,y) + b$$

$$f'_{(1,1)} = 90 + 3 = 93$$

Setelah kecerahan ditingkatkan, matriks citra akan berubah menjadi seperti Gambar 4.6.



93	83
73	43

Gambar 4. 6 Modifikasi Kecerahan pada Matriks Citra

Selain dimodifikasi kecerahannya, citra juga akan dimodifikasi kontrasnya. Di bawah berikut merupakan contoh penghitungan perubahan kontras pada matriks citra pada gambar 4.6, dengan perubahan kontras sebanyak dua kali.

$$f(x,y)' = f(x,y) * P$$

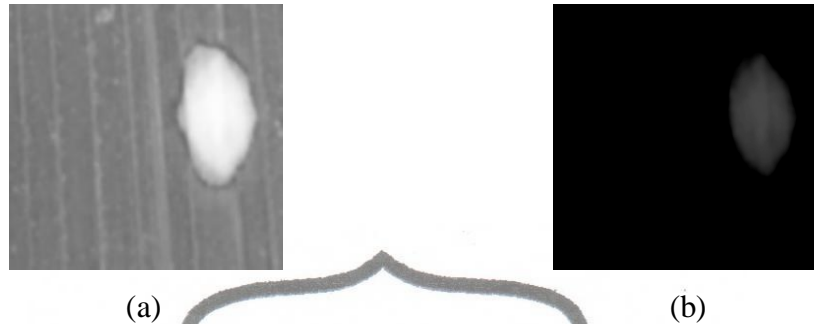
$$f'_{(1,1)} = 90 * 2 = 180$$

Dari perhitungan di atas, maka matriks citra setelah diubah kontrasnya akan menjadi seperti Gambar 4.7.

180	160
140	80

Gambar 4. 7 Modifikasi Kontras pada Matriks Citra

Gambar 4.8 adalah contoh citra *Grayscale* yang dimodifikasi kecerahan dan kontrasnya.



Gambar 4. 8 Citra hasil perubahan kontras dan kecerahan (a) Citra *Grayscale* daun padi (b) Citra daun padi yang sudah dimodifikasi kecerahan dan kontrasnya

3. Reduksi Derau dengan Median *Filtering*

Median filtering bertujuan untuk mereduksi derau yang ada pada data citra padi. Derau pada citra padi berupa objek lain yang tertangkap kamera seperti binatang kecil, serabut-serabut, atau bayangan dari padi lain. Teknik median *filtering* adalah mencari nilai tengah dari piksel-piksel tetangga, kemudian mengganti nilai piksel tengah dengan nilai median.

Gambar 4.9 adalah contoh operasi median *filtering* dengan filter 3 x 3.

113	106	146	130	140
126	124	126	127	135
118	120	130	125	154
119	115	119	123	133
111	116	110	150	120

Gambar 4. 9 Contoh Matriks 5 x 5 Citra

Dari kesembilan piksel di atas, bila ditulis dalam baris akan menjadi :

[124, 126, 127, 120, 130, 125, 115, 119, 123]

Bila diurutkan akan menjadi :

[115, 119, 120, 123, 124, 125, 126, 127, 130]

Dari urutan nilai di atas, diketahui bahwa nilai median dari kesembilan piksel tersebut adalah 124, maka nilai tengah matriks pada gambar 4.10 diubah menjadi seperti pada Gambar 4.10.

113	106	146	130	140
126	124	126	127	135
118	120	124	125	154
119	115	119	123	133
111	116	110	150	120

Gambar 4. 10 Hasil Median *Filtering*

Selanjutnya filter akan berpindah dengan perpindahan per 1 piksel untuk melakukan *filtering* pada piksel-piksel lainnya.

Karena citra daun padi pada gambar 4.1(a) tidak memiliki *noise*, maka digunakan citra daun padi lain sebagai contoh operasi median *filtering*.

Gambar 4. 11 Contoh Penerapan Median *Filtering* pada Citra Padi (a) Citra padi dengan *noise* (b) Citra padi setelah penghilangan *noise*

Dari gambar di atas, derau yang terdapat dalam lingkaran merah pada gambar 4.11(a) berkurang dan citra daun menjadi lebih halus seperti pada gambar 4.11(b).

4.3 Segmentasi Citra

Pada penelitian ini, ada beberapa tahapan dalam segmentasi citra, yaitu : deteksi tepi menggunakan algoritma deteksi tepi Canny, operasi morfologi *closing*, operasi *hole filling*.

1. Canny

Algoritma deteksi canny merupakan kombinasi dari beberapa proses, yaitu :

- Smoothing menggunakan filter gaussian
- Finding gradien menggunakan operator sobel
- Non-maksimum-suppresion

- d. Double thresholding
- e. Edge Tracking by hysteresis

Gambar 4.12(a) adalah citra padi sebelum dideteksi tepinya, sedangkan Gambar 4.12(b) adalah citra padi setelah dideteksi tepi menggunakan algoritma Canny.



Gambar 4. 12 (a) Citra padi sebelum dideteksi tepi (b) Citra padi setelah dideteksi tepi menggunakan Canny

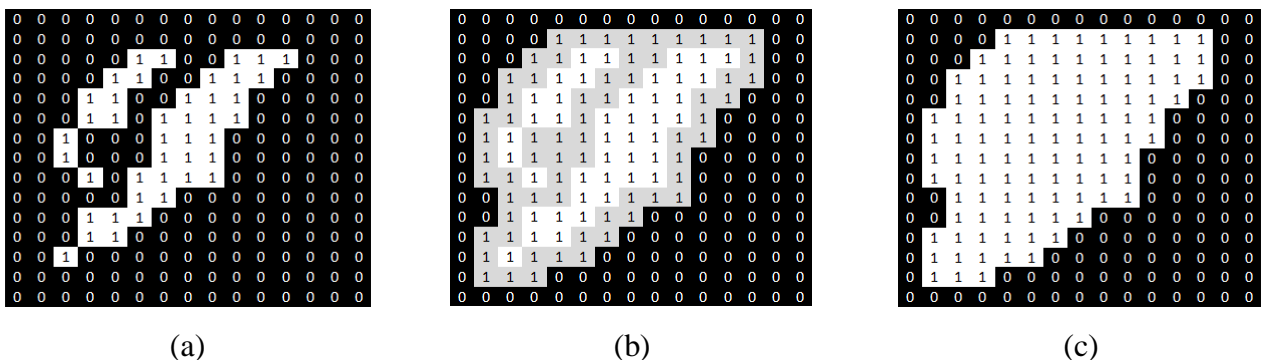
2. Closing

Operasi *closing* terdiri dari dua tahap, yaitu dilasi dan erosi. Operasi *closing* bertujuan untuk mengisi lubang kecil pada objek dan menggabungkan objek yang berdekatan. Di bawah berikut adalah contoh perhitungan operasi *closing*., sedangkan erosi berfungsi untuk mengikis tepi objek.

a. Dilasi

Dilasi bertujuan untuk menyambung titik-titik yang terputus pada citra. Operasi dilasi dilakukan dengan menggunakan Persamaan (2.11)

Gambar 4.13 (a) adalah matriks sebuah citra biner, Gambar 4.13 (b) adalah proses dilasi citra biner, dan Gambar 4.13 (c) adalah citra hasil dilasi.



Gambar 4. 13 Citra dilasi (a) Citra biner asli (b) Proses dilasi citra (c) Hasil citra setelah dilasi

Gambar 4.14 adalah *structuring element* untuk operasi dilasi.

1	1	1
1	1	1
1	1	1

Gambar 4. 14 Structuring element 3x3 operasi dilasi

Gambar 4.13(a) akan didilasi menggunakan structuring element 3x3 seperti pada gambar 4.14. Gambar 4.13(b) merupakan proses dari dilasi citra biner dengan structuring element yang kemudian menghasilkan citra seperti pada gambar 4.13(c).

Citra pada gambar 4.13(a) memiliki himpunan $f(x, y)$ seperti di bawah ini :

$$f(x, y) = \{(3,6), (3,7), (3,10), (3,11), (3,12), (4,5), (4,6), (4,9), (4,10), \dots\},$$

dengan himpunan SE seperti di bawah ini :

$$SE = \{(-1, -1), (-1, 0), (1, 1), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1)\}$$

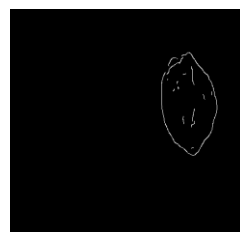
maka contoh perhitungan operasi dilasi adalah sebagai berikut :

$$g(x, y) = \{((3,6) + (-1, -1)), ((3,6) + (-1, 0)), ((3,6) + (-1, 1)), ((3,6) + (0, -1)), ((3,6) + (0, 0)), ((3,6) + (0, 1)), ((3,6) + (1, -1)), ((3,6) + (1, 0)), ((3,6) + (1, 1)), \dots\}$$

$$g(x, y) = \{(2,5), (2,6), (4,7), (3,5), (3,6), (3,7), (4,5), (4,6), (4,7), \dots\}$$

$$g(x, y) = \{(2,5), (2,6), (3,5), (3,6), (3,7), (4,5), (4,6), (4,7), \dots\}$$

Jika operasi diimplementasikan pada citra biner, maka hasilnya adalah seperti Gambar 4.15 (b).



(a)



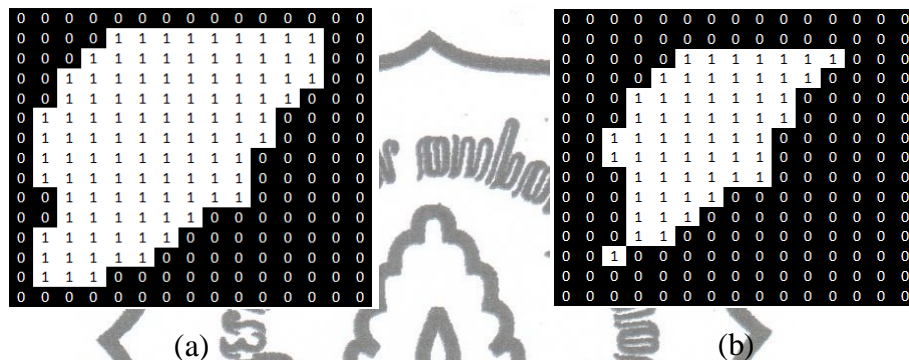
(b)

Gambar 4. 15 Citra contoh dilasi (a) Citra biner sebelum operasi dilasi (b) Citra biner setelah didilasi

b. Erosi

Erosi adalah operasi yang berkebalikan dengan dilasi. Jika dilasi bertujuan untuk mempertebal, maka operasi erosi berfungsi untuk menipiskan garis-garis pada citra. Rumus matematis yang digunakan dalam operasi erosi adalah seperti Persamaan (2.12)

Gambar 4.16 (a) dan (b) merupakan contoh operasi erosi dari gambar 4.14(b).



Gambar 4. 16 Hasil erosi (a) Citra biner sebelum dierosi (b) Citra biner setelah dierosi

Dengan structuring element seperti di Gambar 4.17.

1	1	1
1	1	1
1	1	1

Gambar 4. 17 Structuring element 3x3 operasi erosi

Gambar 4.15(a) akan dierosi menggunakan structuring element 3x3 seperti pada gambar 4.16. Gambar 4.15(b) merupakan hasil dari erosi citra biner dengan structuring element.

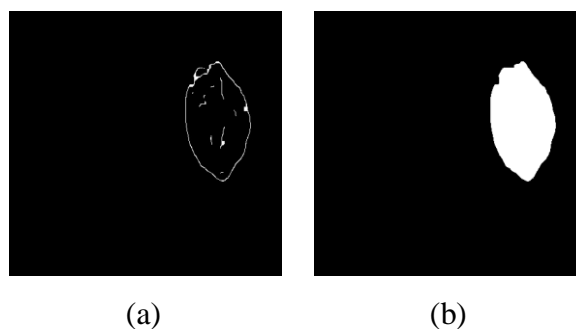
3. Hole filling

Operasi morfologi *hole filling* bertujuan untuk mengisi lubang dalam objek citra. Citra biner yang sudah melalui proses *closing* menghasilkan citra tepi yang antar titikny sudah saling terhubung, namun masih memiliki lubang di tengah objek citra. Untuk memudahkan proses ekstraksi, dibutuhkan operasi *hole filling* yang dapat merubah lubang dalam objek citra yang bernilai 0 (berwarna hitam) menjadi bernilai 1 (berwarna putih).

Operasi *hole filling* memiliki algoritma sebagai berikut :

- a. Memasukkan citra berukuran i baris dan j kolom
- b. Menentukan titik awal objek
 1. Inisiasi variabel 'awal' dengan nilai jumlah kolom citra
 2. Membuat perulangan dari variabel $j = 0$ sampai $j = \text{jumlah kolom (image width)}$
 3. Jika (j, i) bernilai hitam, maka lewatkan
 4. Jika (j, i) bernilai putih, maka awal = j
- c. Menentukan titik akhir objek
 1. Inisiasi variabel 'akhir' dengan nilai 0
 2. Membuat perulangan dari variabel $j = 0$ sampai $j = \text{jumlah kolom (image width)}$
 3. Jika (j, i) bernilai hitam, maka lewatkan
 4. Jika (j, i) bernilai putih, maka 'akhir' = j
- d. *Hole filling* (mengisi lubang) objek dengan nilai 1 (putih)
 1. Membuat perulangan $x = \text{'awal'}$ sampai $x = \text{'akhir'}$
 2. Set lubang citra dengan warna putih

Gambar 4.18 (a) dan (b) berikut merupakan contoh hasil proses operasi morfologi *hole filling* :



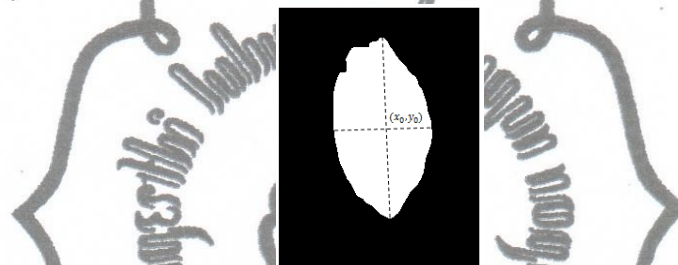
Gambar 4. 18 Citra hasil hole filling (a) Citra biner sebelum operasi *hole filling*
(b) Citra biner setelah operasi *hole filling*

4.4 Ekstraksi Ciri

Ekstraksi ciri memiliki beberapa tahap, yaitu: mencari *centroid* citra biner, melakukan erosi sebesar 1 piksel, mencari selisih dua citra biner, dan mengkonversi sudut kartesian menjadi sudut polar. Berikut adalah penjelasan ekstraksi ciri di setiap tahap.

1. Mencari *Centroid* citra biner

Centroid diperoleh dengan menggunakan nilai rerata koordinat setiap piksel yang menyusun objek. Gambar 4.19 adalah contoh centroid pada citra biner.



Gambar 4. 19 Centroid pada citra biner

Algoritma untuk mencari *centroid* sebagai berikut :

- a. Memasukkan citra yang berukuran m baris dan n kolom
- b. Membuat variabel $pusatx$, $pusaty$, dan $luas$
- c. Inisialisasi ketiga variabel tersebut dengan nilai 0
- d. Membuat perulangan $q = 1$ sampai m (baris) dan $p = 1$ sampai n (kolom)
- e. Jika $f(q, p) = 1$, maka:

$$Luas = luas + 1;$$

$$Pusatx = pusatx + p;$$

$$Pusaty = pusaty + q$$
- f. Jika perulangan selesai, maka :

$$Pusatx = pusatx / luas;$$

$$Pusaty = pusaty / luas$$

Dengan algoritma di atas, sebagai contoh, sebuah objek citra memiliki *luas* sebesar 7164 piksel dengan jumlah kordinat piksel pada sumbu x sebanyak 1589976 dan jumlah koordinat piksel pada sumbu y sebanyak 1590160 piksel.

Maka, *centroid* (x, y) adalah sebagai berikut :

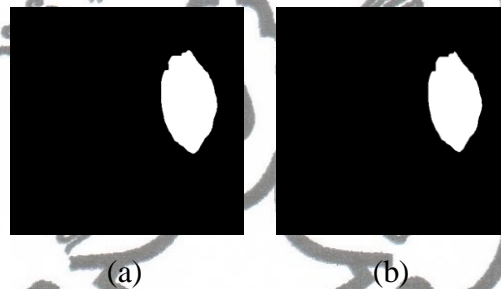
$$x_0 = \frac{\text{jumlah koordinat piksel sumbu } x}{\text{luas}} = \frac{1589976}{7164} = 221.9 = 222$$

$$y_0 = \frac{\text{jumlah koordinat piksel sumbu } y}{\text{luas}} = \frac{1590160}{7164} = 221.9 = 222$$

2. Erosi citra biner sebesar 1 piksel

Tujuan operasi erosi pada tahap ekstraksi ciri adalah untuk menghasilkan citra tererosi yang akan dicari selisihnya dengan citra asli. Citra asli dierosi sebesar 1 piksel.

Gambar 4.17(a) merupakan citra biner sebelum dierosi, sedangkan gambar 4.17(b) merupakan citra setelah dierosi 1 piksel.



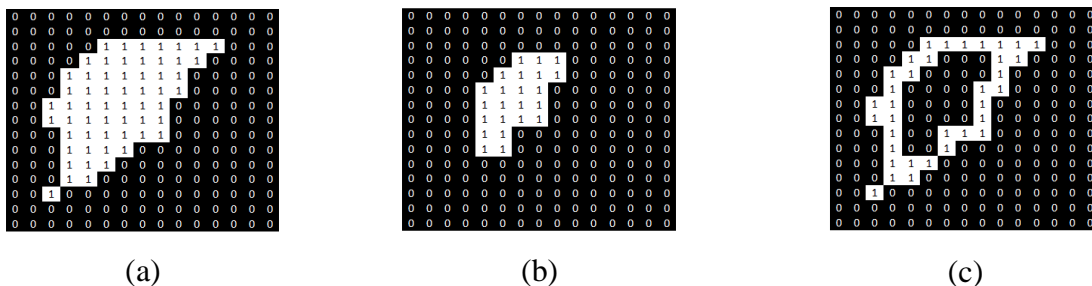
Gambar 4. 20 Citra erosi 1 piksel (a) Citra biner sebelum dierosi (b) Citra biner setelah dierosi

3. Mencari selisih dua citra biner

Selisih dua citra biner diperoleh dengan menggunakan rumus :

$$Q(i, j) = P_1(i, j) - P_2(i, j)$$

Selisih dicari untuk mendapatkan tepi dari citra biner. Gambar 4.21 (a) adalah citra biner, Gambar 4.21(b) adalah citra biner (a) setelah dierosi 1 piksel, dan Gambar 4.21 (c) adalah hasil selisih citra biner (a) dan (b).



Gambar 4. 21 Proses mencari selisih citra (a) Citra biner (b) Citra biner setelah dierosi 1 piksel (c) Hasil selisih citra (a) dan (b)

Jika dinotasikan dalam operasi matematika, maka :

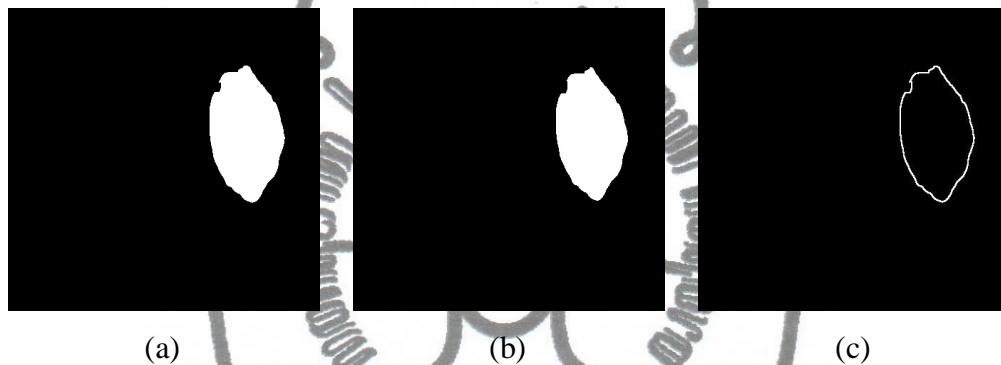
$$P_1(i, j) = \{(3,6) \dots, (4,5), (4,6), (4,7), (4,8), (4,9), (4,10), (4,11), \dots\}$$

$$P_2(i, j) = \{(4,7), (4,8), (4,9), \dots\}$$

$$Q(i, j) = P_1(i, j) - P_2(i, j)$$

$$Q(i, j) = \{(3,6), \dots (4,5), (4,6), (4,10), (4,11), \dots\}$$

Gambar 4.22 (a), (b), adalah dua citra biner yang dicari selisihnya. Selisih dari kedua citra adalah Gambar 4.22 (c).



Gambar 4. 22 Contoh hasil selisih citra (a) Citra biner padi (b) Citra biner padi setelah dierosi 1 piksel (c) Hasil selisih citra (a) dan (b)

4. Mengkonversi sudut kartesian menjadi polar

Pada tahap ini, setiap titik (x, y) dicari jarak (*distance*), dan sudut *thetanya* dari titik *centroid* (x_0, y_0) , dengan menggunakan rumus seperti di bawah berikut :

$$distance = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

$$\theta = \tan^{-1} \frac{(y_1 - y_0)}{(x_1 - x_0)}$$

Sebagai contoh, ada sebuah fungsi $f(x_1, y_1)$ yang memiliki nilai (19 , 10) dengan nilai titik *centroid* $f(x_0, y_0)$ sebesar (7 , 5), maka :

$$\begin{aligned} distance &= \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \\ &= \sqrt{(19 - 7)^2 + (10 - 5)^2} \\ &= \sqrt{12^2 + 5^2} = \sqrt{144 + 24} = \sqrt{169} = 13 \end{aligned}$$

$$\theta = \tan^{-1} \frac{5}{12} = 22.6^\circ$$

Dari contoh perhitungan di atas, maka koordinat kartesian yang awal mula bernilai (19, 10), setelah dikonversi mejadi koordinat polar memiliki nilai (22.6 , 13). Sudut yang digunakan pada penelitian ini adalah sudut dengan kelipatan lima (0°, 5°, 10°, ..., 350°, 355°).

5. Normalisasi nilai *distance* pada koordinat polar menggunakan normalisasi min-max

Proses normalisasi min-max menggunakan rumus :

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

Contohnya, dalam sebuah kumpulan data koordinat polar :

Tabel 4. 1 Koordinat polar citra biner

No	Sudut <i>theta</i>	<i>Distance</i> dari centroid ke titik tepi citra
1	0°	25
2	5°	27
3	10°	30
4	15°	24
5	20°	21

Dimana nilai max(x) adalah 30 dan nilai min(x) adalah 21, maka contoh perhitungan normalisasinya adalah :

$$\begin{aligned}
 z_0 &= \frac{x_0 - \min(x)}{\max(x) - \min(x)} \\
 &= \frac{25 - 21}{30 - 21} = \frac{4}{9} = 0.44
 \end{aligned}$$

Dari proses normalisasi di atas, maka akan didapatkan nilai normalisasi seperti pada tabel di bawah :

Tabel 4. 2 Koordinat polar citra biner setelah proses normalisasi nilai *distance*

No	Sudut <i>theta</i>	<i>Distance</i> dari centroid ke titik tepi citra
1	0°	0.44
2	5°	0.89
3	10°	1
4	15°	0.33
5	20°	0

6. Menyimpan sudut polar

Tahap selanjutnya adalah menyimpan koordinat polar pada file berekstensi .csv (*comma seperated value*). Nilai *distance* dan *theta* disiman dengan memisahkan masing-masing nilai dengan tanda koma. Koordinat polar disimpan untuk diproses pada tahap ekstraksi ciri. Dalam penelitian ini, tidak seluruh koordinat polar disimpan, tapi hanya kelipatan lima saja (0° , 5° , 10° , ...).

Contoh dari koordinat polar yang disimpan dalam bentuk file.csv adalah seperti pada Tabel 4.3.

Tabel 4. 3 Contoh *file comma seperated value* yang berisi nilai *theta* dan *distance*

No	Theta	Distance
1	0.0	0.13677317212544002
2	5.0	0.09257530461823762
3	10.0	0.05939627800419474
4	15.0	0.041778522355934065
5	20.0	0.02221349556612378
6	25.0	0.016845800953429933
7	30.0	0.025152236020043173
8	35.0	0.03385334650773606
9	40.0	0.04925413118706153
10	45.0	0.07902374393896673
11	50.0	0.11929967243126748
...(lihat lampiran 5)		

4.5 Klasifikasi Gejala Penyakit Padi

Pada penelitian ini, terdapat dua klasifikasi gejala penyakit padi, yaitu; klasifikasi bentuk gejala penyakit dan klasifikasi warna gejala penyakit padi.

1. Klasifikasi bentuk gejala penyakit padi

Klasifikasi bentuk penyakit menggunakan file csv sudut polar. Terdapat empat file csv yang menjadi acuan klasifikasi bentuk, yaitu : *elis.csv*, *diamond.csv*, *rectangle.csv*, *circle.csv*.

Tabel 4.4 adalah tabel yang berisi nilai *distance* keempat bentuk acuan dan nilai *distance* citra inputan, beserta selisih *distance* citra inputan dengan keempat bentuk acuan.

Tabel 4. 4 Selisih citra inputan dengan keempat bentuk acuan

No	Theta	Distance					Selisih			
		diamond	Elips	Circle	Rectangle	blast	selisih diamond	selisih elips	selisih circle	selisih rectangle
1	0	0,136773172	0,008562016	0,721515075	0,006754562	0,083581761	0,053191411	0,075019745	0,637933315	0,076827199
2	5	0,092575305	0,024278201	0,966706932	0,011442532	0,118917144	0,026341839	0,094638943	0,847789788	0,107474612
3	10	0,059396278	0,041575819	0,849996945	0,021327465	0,126017415	0,066621137	0,084441595	0,72397953	0,104689949
4	15	0,041778522	0,066136025	0,992076775	0,036568773	0,128704495	0,086925972	0,062568469	0,86337228	0,092135722
5	20	0,022213496	0,106532204	0,753470249	0,059056346	0,134299742	0,112086246	0,027767538	0,619170507	0,075243396
6	25	0,016845801	0,147928991	0,732137708	0,089237632	0,137392955	0,120547154	0,010536037	0,594744753	0,048155323
7	30	0,025152236	0,20372335	0,717784062	0,126919179	0,161196624	0,136044388	0,042526726	0,556587438	0,034277445
8	35	0,033853347	0,267532559	0,736292793	0,176900868	0,193041037	0,15918769	0,074491522	0,543251756	0,016140168
9	40	0,049254131	0,338192004	0,60901138	0,241757728	0,258451841	0,20919771	0,079740163	0,350559539	0,016694113
10	45	0,079023744	0,417428243	0,466415317	0,322724278	0,349068085	0,270044341	0,068360158	0,117347232	0,026343808
11	50	0,119299672	0,511680218	0,634483185	0,429090141	0,420663536	0,301363863	0,091016683	0,21381965	0,008426605
12	55	0,162188185	0,599790224	0,445198135	0,571701904	0,475308812	0,313120627	0,124481413	0,030110676	0,096393092
13	60	0,224616122	0,688493066	0,509189192	0,771050992	0,522609457	0,297993336	0,165883609	0,013420266	0,248441535
14	65	0,312310643	0,776374725	0,471540036	0,974032901	0,591020399	0,278709757	0,185354326	0,119480364	0,383012502
15	70	0,392586097	0,852119733	0,385015281	0,979260339	0,699796769	0,307210672	0,152322964	0,314781488	0,279463569
...(lihat lampiran 50)										
72	355	0,114150422	0,012566368	0,852415991	0,007299941	0,078492932	0,03565749	0,065926564	0,773923059	0,071192991
Jumlah Selisih							10,76432226	8,068152483	29,17923348	8,765615289

Jika dilihat dari tabel di atas, selisih jarak yang paling kecil antara data blas dengan data keempat bentuk acuan adalah selisih jarak antara blas dengan bentuk elips, oleh karena itu, data blas yang menjadi inputan diklasifikasikan ke dalam bentuk elips.

2. Klasifikasi warna gejala penyakit padi

Klasifikasi selanjutnya adalah klasifikasi warna menggunakan histogram. Sebelum data inputan dimasukkan, identifikasi warna pada citra terlebih dahulu dilakukan dengan menggunakan 19 data *training* seperti pada Tabel 3.3. Hal ini bertujuan untuk mendeteksi di kelas RGB apa, warna-warna tersebut ada. Warna-warna yang akan dimasukkan ke dalam klasifikasi pada penelitian ini adalah; hijau keputihan, hijau kekuningan, coklat keputihan, dan coklat kemerahan.

Dengan keempat klasifikasi di atas, maka sebaran RGB yang harus diidentifikasi adalah warna hijau, kuning, coklat, putih, dan merah. Berikut adalah *channel* histogram masing-masing warna. *Channel* histogram didapatkan dari pembagian nilai RGB, seperti yang sudah dibahas pada bab Tinjauan Pustaka, Tabel 2.4. Tabel 4.5 adalah daftar warna pada klasifikasi gejala penyakit padi.

Tabel 4. 5 Daftar warna pada klasifikasi gejala penyakit padi

No	Warna Klasifikasi	Histogram	Warna
1	Putih	[3 3 1]	
		[3 3 2]	
		[3 3 3]	
2	Coklat	[2 2 1]	
		[2 1 0]	
		[2 1 1]	
3	Merah	[1 0 0]	
4	Hijau	[0 1 0]	
		[1 2 1]	
		[1 2 0]	
		[2 3 3]	
		[1 1 0]	
		[1 2 2]	
		[2 2 2]	
5	Kuning	[3 2 1]	
		[2 3 1]	
		[2 2 0]	

Gambar 4.23 merupakan contoh data training yang digunakan untuk menentukan *rule* klasifikasi warna gejala penyakit padi.



Gambar 4. 23 Data training untuk klasifikasi warna

Untuk menentukan *rule* klasifikasi warna, diperlukan beberapa langkah, yaitu :

- a. Mencari jumlah piksel pada masing-masing *channel* histogram

Gambar 4.23 memiliki jumlah piksel pada masing-masing *channel* histogram, seperti pada Tabel 4.6.

Tabel 4. 6 Persebaran histogram pada citra gambar 4.23

No	Warna Klasifikasi	Histogram	Jumlah Piksel yang Terdapat pada Histogram
1	Putih	[3 3 1]	19
		[3 3 2]	70
		[3 3 3]	199
2	Coklat	[2 2 1]	43
		[2 1 0]	183
		[2 1 1]	20
3	Merah	[1 0 0]	124
4	Hijau	[0 1 0]	16
		[1 2 1]	1
		[1 2 0]	199
		[2 3 3]	0
		[1 1 0]	64
		[1 2 2]	0
		[2 2 2]	2
5	Kuning	[3 2 1]	48
		[2 3 1]	102
		[2 2 0]	201

- b. Menghitung jumlah piksel pada masing-masing warna klasifikasi

Berdasarkan Tabel 4.6, jumlah piksel pada masing-masing warna klasifikasi dihitung seperti di bawah berikut.

$$\sum \text{piksel putih} = [3\ 3\ 1] + [3\ 3\ 2] + [3\ 3\ 3] = 19 + 70 + 199 = 288$$

$$\sum \text{piksel coklat} = [2\ 2\ 1] + [2\ 1\ 0] + [2\ 1\ 1] = 43 + 183 + 20 = 246$$

$$\sum \text{piksel merah} = [1\ 0\ 0] = 124$$

$$\begin{aligned} \sum \text{piksel hijau} &= [0\ 1\ 0] + [1\ 2\ 1] + [1\ 2\ 0] + [2\ 3\ 3] + [1\ 1\ 0] + [1\ 2\ 2] + [2\ 2\ 2] \\ &= 16 + 1 + 199 + 0 + 64 + 0 + 2 = 282 \end{aligned}$$

$$\sum \text{piksel kuning} = [3\ 2\ 1] + [2\ 3\ 1] + [2\ 2\ 0] = 48 + 102 + 201 = 351$$

c. Menghitung proporsi masing-masing warna klasifikasi pada citra padi

Proporsi piksel warna diperoleh dengan membagi jumlah piksel pada histogram dengan jumlah seluruh piksel dalam citra. Di bawah ini adalah perhitungan proporsi masing-masing warna, dengan total piksel citra sebesar 2294 piksel.

$$\begin{aligned} \text{proporsi putih} &= \frac{\text{total piksel putih}}{\text{total piksel citra}} \\ &= \frac{288}{2294} = 0.12554 \end{aligned}$$

$$\begin{aligned} \text{proporsi coklat} &= \frac{\text{total piksel coklat}}{\text{total piksel citra}} \\ &= \frac{246}{2294} = 0.1072 \end{aligned}$$

$$\begin{aligned} \text{proporsi merah} &= \frac{\text{total piksel merah}}{\text{total piksel citra}} \\ &= \frac{124}{2294} = 0.054 \end{aligned}$$

$$\begin{aligned} \text{proporsi hijau} &= \frac{\text{total piksel hijau}}{\text{total piksel citra}} \\ &= \frac{282}{2294} = 0.054 \end{aligned}$$

$$\begin{aligned} \text{proporsi kuning} &= \frac{\text{total piksel kuning}}{\text{total piksel citra}} \\ &= \frac{351}{2294} = 0.153 \end{aligned}$$

d. Mencari batas atas dan batas bawah masing-masing warna klasifikasi

Untuk menentukan rule pada klasifikasi warna, diperlukan training data. Data yang digunakan seperti pada Tabel 3.1. Misalnya, untuk melatih warna coklat putih, proporsi pada masing-masing warna diidentifikasi terlebih dahulu, seperti pada Tabel 4.7.

Tabel 4. 7 Proporsi masing-masing warna klasifikasi pada citra training

No	Nama File	putih	coklat	merah	kuning
1	elips 6	0,33	0,23	0,03	0,19
2	circle 12	0,02	0,49	0,004	0,256
3	circle 5	0,42	0,45	0	0,0525
4	diamond 11	0,126	0,11	0,05	0,153
5	diamond 13	0,11	0,07	0,02	0,083
6	elips 9	0,13	0,3	0,12	0,1
7	diamond 1	0,2	0,11	0,116	0,346
Nilai Rata-rata		0,190	0,251	0,048	0,169
Standar Deviasi		0,138	0,169	0,050	0,104
Batas bawah		-0,081	-0,080	-0,05	-0,035
Batas Atas		0,463	0,583	0,147	0,373

Kemudian, proporsi warna pada masing-masing data training dicari reratanya, seperti di bawah ini.

$$\text{Rata - rata proporsi putih} = \frac{0,03+0,02+0,42+0,126+0,11+0,13+0,2}{7} = 0,19$$

Selanjutnya, dicari nilai standar deviasi dengan menggunakan Persamaan (2.18).

Selanjutnya, digunakan distribusi Z untuk menentukan batas atas dan batas bawah klasifikasi warna dengan persamaan (3.10)

Nilai batas atas dan batas bawah akan menjadi rule yang digunakan dalam klasifikasi warna.

Batas atas dan batas bawah masing-masing komponen warna klasifikasi menjadi acuan untuk menentukan *rule* klasifikasi warna. Dari Tabel 4.7 di atas, maka dapat dibuat *rule* klasifikasi sebagai berikut :

- a. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.46, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.583, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.147, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0,373, maka citra terklasifikasi **coklat putih**
- b. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.421, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.209, dan proporsi merah sama dengan 0, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.018, maka citra terklasifikasi **hijau putih**
- c. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.246, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.97, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.61, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.233, maka citra terklasifikasi **coklat merah**
- d. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.468, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.479, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.0059, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.546, maka citra terklasifikasi **hijau kuning**

4.6 Pengujian

Pengujian dilakukan dua kali dengan menggunakan *rule* klasifikasi warna yang berbeda pada tiap kali pengujian. Terdapat lima kelas klasifikasi pada klasifikasi bentuk dan klasifikasi warna. Kelas yang terdapat pada masing-masing klasifikasi dapat dilihat pada Tabel 3.3.

A. Confusion matrix

Tabel 4.8 adalah *confusion matrix* klasifikasi gejala penyakit padi.

Tabel 4. 8 *Confusion matrix* pengujian pertama

aktual	prediksi																	Recall	
		d-cp	d-hk	d-hp	d-cm	d-u	e-hk	e-cm	e-hp	r-hk	r-hp	r-cm	r-u	c-cp	c-hp	c-cm	u-cm		u-u
	d-cp	14				1													0,933
	d-hk					1													0
	d-hp	2	1																0
	d-cm							1											0
	d-u																		0
	e-hk									1									0
	e-cm							5				1					1		0,714
	e-hp																		0
	r-hk						1												0
	r-hp		1							1	2								0,5
	r-cm																		0
	r-u																		0
	c-cp		1						1				1	2				1	0,333
	c-hp										1								0
	c-cm				1	1										1			0,333
	u-cm																		0
	u-u																	5	1
Presisi		0,875	0	0	0	0	0	0,833	0	0	0,667	0	0	1	0	1	0	0,833	

Keterangan :

d-cp : Diamond-coklat putih	e-hp : Elips-hijau putih
d-hk : Diamond-hijau kuning	e-hk : Elips-hijau kuning
d-hp : Diamond-hijau putih	e-cm : Elips-coklat merah
d-cm : Diamond-coklat merah	e-u : Elips-unidentified
d-u : Diamond-unidentified	u-u : Unidentified-unidentified
r-cp : Rectangle-coklat putih	u-cp : Unidentified-coklat putih
r-hk : Rectangle-hijau kuning	u-hk : Unidentified-hijau kuning
r-cm : Rectangle-coklat merah	c-u : Circle-unidentified
r-hp : Rectangle-hijau putih	c-cm : Circle-coklat merah
r-u : Rectangle-unidentified	c-hk : Circle-hijau kuning
c-cp : Circle-coklat putih	c-hp : Circle-hijau putih

Dari tabel di atas, akurasi yang didapatkan adalah sebagai berikut :

$$akurasi = \frac{total\ TP}{total\ data} * 100\% = \frac{29}{48} * 100\% = 60.42\%$$

Sedangkan, presisi dan recall didapatkan dari perhitungan di bawah berikut :

$$recall = \frac{TP}{TA} * 100\% = 63,6\%$$

$$presisi = \frac{TP}{FP} * 100\% = 86,8\%$$

Pada klasifikasi bentuk, tabel confusion matrix yang didapatkan seperti pada Tabel 4.9.

Tabel 4. 9 *Confusion matrix* klasifikasi bentuk pada pengujian pertama

	prediksi					
		diamond	rectangle	elips	circle	unidentified
aktual	diamond	14		1		
	rectangle	2	3	1		
	elips	1	2	7		1
	circle	3	2	1	4	1
	unidentified					5
	presisi	0,7	0,428571429	0,7	1	0,714285714

Nilai akurasi pada pengujian klasifikasi bentuk adalah :

$$akurasi\ bentuk = \frac{total\ TP}{total\ data} * 100\% = \frac{33}{48} * 100\% = 68.8\%$$

Sedangkan, pada pengujian klasifikasi warna, confusion matrix yang terbentuk seperti pada Tabel 4.11.

Tabel 4. 10 *Confusion matrix* klasifikasi warna pada pengujian pertama

	Prediksi						recall
		coklat putih	hijau putih	coklat merah	hijau kuning	unidentifikasi	
aktual	coklat putih	14			1	3	0,778
	hijau putih	2	6		3		0,545
	coklat merah			10		1	0,909
	hijau kuning				2	1	0,667
	unidentifikasi					5	1
	presisi	0,875	1	1	0,333	0,5	

Nilai akurasi pada pengujian klasifikasi warna adalah :

$$akurasi\ warna = \frac{total\ TP}{total\ data} * 100\% = \frac{37}{48} * 100\% = 77\%$$

Rule yang digunakan pada pengujian pertama klasifikasi pertama adalah :

- Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.46, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.583, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.147, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0,373, maka citra terklasifikasi **coklat putih**
- Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.421, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.209, dan proporsi merah sama dengan 0, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.018, maka citra terklasifikasi **hijau putih**

- c. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.246, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.97, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.61, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.233, maka citra terklasifikasi **coklat merah**
- d. Jika proporsi putih lebih besar sama dengan 0, dan lebih kecil sama dengan 0.468, dan proporsi coklat lebih besar sama dengan 0 dan lebih kecil sama dengan 0.479, dan proporsi merah lebih besar sama dengan 0 dan lebih kecil sama dengan 0.0059, dan proporsi kuning lebih besar sama dengan 0, dan lebih kecil sama dengan 0.546, maka citra terklasifikasi **hijau kuning**

B. Analisis Hasil Pengujian

Dari hasil pengujian di atas, ada beberapa hal yang dapat dianalisa, yaitu :

1. Pada klasifikasi bentuk, sistem bekerja baik pada bentuk belah ketupat. Hal ini ditunjukkan dengan tingginya angka *recall* (kemampuan sistem untuk mengingat sebuah informasi) sebesar 93.3% pada bentuk belah ketupat. Sebaliknya, pada bentuk bulat, sistem belum dapat mengingat bentuk bulat dengan baik. Hal ini ditunjukkan dengan angka *recall* bentuk bulat yang hanya sebesar 36.4%. Sistem hanya mampu mendeteksi bentuk bulat sebanyak 4 buah data citra bentuk bulat dari total 11 data. Sebagian besar data citra aktual pada bentuk bulat terdeteksi persegi panjang di dalam sistem. Hal ini dapat disebabkan karena bentuk acuan belum representatif dengan bentuk penyakit pada data citra asli sehingga banyak objek yang tidak terdeteksi (gambar terlampir). Sedangkan pada bentuk persegi panjang, sistem dapat mengenali bentuk persegi panjang sebesar 50%, atau 3 dari 6 data persegi panjang. 2 di antara data persegi panjang terdeteksi sebagai bentuk belah ketupat. Hal ini dapat disebabkan oleh kemiripan yang terdapat pada bentuk persegi panjang dan belah ketupat, baik dari bentuk bangunnya, maupun sudut pada kedua bentuk tersebut. Sedangkan pada bentuk elips, sistem berhasil mendeteksi bentuk elips sebanyak 7 data dari total 11 data citra. 4 data bentuk elips yang lain terdeteksi sebagai belah ketupat, persegi panjang, dan objek *unidentified*. Data yang terdeteksi sebagai objek *unidentified*, tidak

berhasil tersegmentasi bentuknya oleh sistem, hal ini disebabkan karena kualitas citra yang tidak bagus, *noise* yang tidak berhasil dihilangkan, dan objek citra yang tidak dapat dipisahkan dengan latar (*background*).

2. Klasifikasi bentuk dapat dioptimalkan dengan menggunakan bentuk acuan yang berasal dari bentuk gejala penyakit padi yang asli, sebab data inputan dan data acuan memiliki kemiripan yang tinggi. Hal ini memungkinkan sistem untuk mendapatkan nilai akurasi yang lebih tinggi.
3. Dalam klasifikasi warna, sistem memiliki akurasi sebesar 77%. Sistem dapat mengenali warna coklat putih dengan baik, hal ini dibuktikan dengan nilai *recall* sebesar 77.78%. Sebanyak 14 data dari 18 data aktual coklat putih dapat terdeteksi sebagai coklat putih, sedangkan 1 data terdeteksi coklat kuning, dan 3 data tidak dapat terdeteksi oleh sistem. Sistem juga mampu memberikan jawaban yang presisi sesuai dengan informasi yang diminta oleh pengguna, hal ini ditunjukkan dengan nilai presisi klasifikasi warna coklat putih sebesar 87.5%. Pada klasifikasi warna hijau putih data yang terdeteksi benar dengan data aktual sebanyak 6 data dari total 11 data, 3 data terdeteksi warna hijau kuning, dan 2 data terdeteksi coklat putih. Klasifikasi warna hijau putih dalam sistem, memiliki tingkat ketepatan yang tinggi antara jawaban yang diberikan oleh sistem dan informasi yang diminta oleh pengguna. Hal ini ditunjukkan dengan nilai presisi pada klasifikasi warna hijau putih sebesar 100 %. Sama halnya dengan klasifikasi coklat merah, sistem dapat memberikan jawaban yang presisi sesuai dengan informasi yang diminta oleh pengguna, hal ini ditunjukkan dengan nilai presisi klasifikasi coklat merah sebesar 100%. Pada warna hijau kuning, kemampuan sistem dalam mengenali kembali warna bernilai 66.67%, sedangkan kemampuan sistem untuk memberikan jawaban yang sesuai hanya 33.3%, hal ini disebabkan, ada 4 data yang seharusnya masuk ke dalam warna klasifikasi lain, namun terdeteksi sebagai hijau kuning.
4. Metode klasifikasi yang digunakan dalam penelitian ini adalah metode klasifikasi if-else yang sederhana, dan parameter yang digunakan sebagai *rule* masih tergolong sedikit. Hal ini menyebabkan beberapa data tidak dapat

terklasifikasi sesuai harapan. Oleh karena itu, dibutuhkan metode klasifikasi lain, yang meyertakan parameter klasifikasi yang lebih banyak, agar mendapatkan hasil klasifikasi yang lebih akurat.

