

BAB 1V

HASIL SIMULASI DAN ANALISIS

Hasil simulasi yang didapat dari *script coding* atau program simulasi menggunakan Matlab. Hasil yang dianalisis pada Tugas akhir ini yaitu luas area objek buah semangka dan bentuk buah semangka dengan *input* gambar buah semangka yang didapat dari kamera *digital*. Hasil analisis program untuk menentukan luas area objek buah semangka akan dibuat dalam sebuah table sehingga dapat di kelompok buah semangka.

4.1 Pra proses

4.1.1 Konversi RGB ke bentuk abu-abu

Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi *gray-scale*, seperti gambar 4.2 (b), dalam gambar 4.2 (b) tidak ada lagi warna, yang ada, adalah derajat keabuan, untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing R, G, B menjadi citra *grayscale*, maka konversi dapat dilakukan dengan mengambil nilai rata-rata dari nilai R, G, dan B, sehingga dapat dirumuskan menjadi,

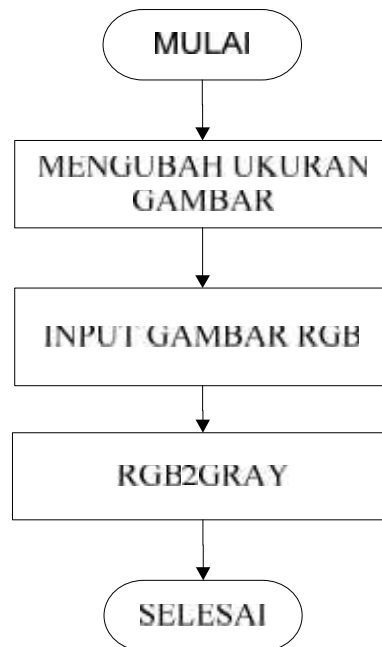
$$S = \frac{r + g + b}{3}$$

Perintah sintaks yaitu

```
I=rgb2gray % Merubah warna abu-abu (grayscale)  
imshow(I); % Menampilkan gambar (grayscale)
```

Gambar 4.1 proses kerja konversi RGB ke bentuk abu-abu, proses ini ada tiga tahapan untuk merubah warna ke abu-abu. Tahapan pertama diawali dengan mengubah ukuran gambar yang didapat dari kamera digita dengan ukuran 176x132. Proses selanjutnya gambar asli dirubah ke warna abu-abu yang terlihat pada gambar 4.2, hal ini digunakan untuk menyederhanakan model citra. Konversi gambar berwarna, menjadi *grayscale* perlu dilakukan agar lebih mudah untuk menghilangkan *noise* pada gambar. *Grayscale image* merupakan gambar

digital yang hanya memiliki satu nilai pada setiap *pixel*-nya dengan kedalaman warna 8 bit.



Gambar 4.1. Tampilan Grayscale

Gambar 4.2 hasil gambar asli menjadi *grayscale* yang dilakukan dengan cara mengambil semua *pixel* pada gambar kemudian warna tiap *pixel* akan diambil informasi mengenai 3 warna dasar yaitu merah, biru dan hijau, ketiga warna dasar ini akan dijumlahkan kemudian dibagi tiga sehingga didapat nilai rata-rata. Nilai rata-rata inilah yang akan dipakai untuk memberikan warna pada *pixel*.



(a) Original RGB *image*



(b) Konversi RGB ke bentuk *grayscale*



(a) Original RGB *image*

(b) Konversi RGB ke bentuk
grayscale

Gambar 4.2. Konversi RGB ke bentuk *grayscale*

4.2 Menghilangkan noise

Citra hasil rekaman kamera digital sering sekali terdapat beberapa gangguan yang mungkin terjadi, seperti lensa tidak fokus, muncul bintik-bintik yang disebabkan oleh proses *capture* yang tidak sempurna, pencahayaan yang tidak merata yang mengakibatkan intensitas tidak seragam, kontras citra terlalu rendah sehingga objek sulit dipisahkan dari latar belakangnya, atau gangguan yang disebabkan oleh kotoran-kotoran yang menempel pada citra dan lain sebagainya seperti gambar 4.4 (a), maka proses pengolahan citra yang akan memberikan hasil yang tidak optimal. Oleh karena itu, diperlukan suatu proses *noise filtering* untuk mengurangi noise yang terdapat, sehingga menghasilkan seperti gambar 4.4 (b) dan perintah sintaks seperti:

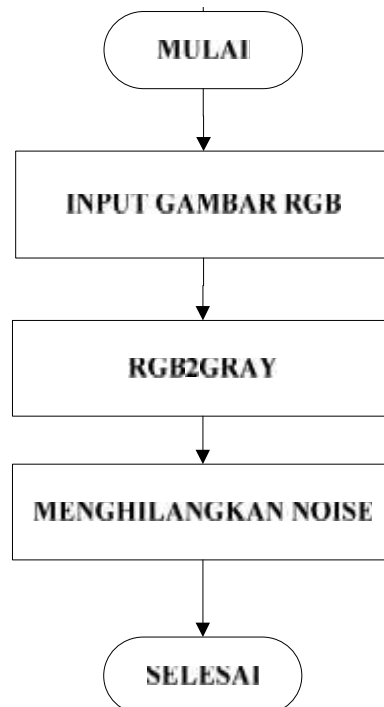
```
gambar=imread('3,5 kg.jpg'); %Memanggil gambar File
[X,map] = rgb2ind(gambar, 128);% %reconstruction merupakan proses
                                untuk perbaikan image
I = ind2gray(X,map); % proses rekonstruksi pada matla
gray=rgb2gray(gambar); %merubah dari RGB ke Graycale
thresh=graythresh(gray);%digunakan untuk medapatkan nilai ambang batas
```

`imbw=im2bw(gray,thresh);` % mengubah image dari grayscale kedalam bentuk biner threshold menghasilkan gambar hitam putih.

`Ifill = imfill (imbw,'holes');` % fungsinya menampilkan obek berwarna hitam putih pada bagian obek.menghasilkan biner.

`figure, imshow(imbw);figure, imshow(Ifill)` % menampilkan gambar/objek

Pada gambar 4.3 menerangkan proses menghilangkan *noise* ketika dilakukan proses pengambilan gambar, seringkali terdapat *noise* yang masuk ke dalam citra, sehingga menyebabkan timbulnya bercak-bercak yang tidak beraturan. Oleh karena itu, diperlukan suatu proses *noise* pengolahan citra baik proses binerisasi maupun deteksi tepi menghasilkan citra yang pada umumnya masih belum baik, oleh karena itu perlu dilakukan perbaikan citra /rekonstruksi citra kembali. Di matlab proses rekonstruksi dilakukan menggunakan fungsi `imfill`, dapat dilihat perbedaan gambar 4.4 a dan b, gambar a terdeteksi *noise* dan gambar b bebas *noise*.



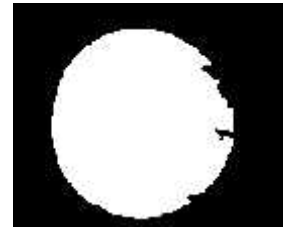
Gambar 4.3. Menghilangkan *noise*

Setelah perubahan gambar berwarna menjadi gambar *grayscale* yang hanya memiliki derajat keabuan, selanjutnya dilakukan penghalusan gambar.

Penghalusan dalam hal ini adalah menghilangkan/mereduksi *noise* yang muncul pada saat gambar diambil dari kamera dapat terlihat pada gambar 4.4. gambar 4.4 a terdeteksi *noise* dan b bebas *noise*.



(a) Terdeteksi *noise*



(b) Bebas *noise*



(a) Terdeteksi *noise*



(b) Bebas *noise*

Gambar 4.4. Hasil tampilan menghilangkan *noise*

4.3 Merubah bentuk biner

Binerisasi citra merupakan proses merubah citra ke dalam bentuk biner (0 dan 1). Dengan merubah ke bentuk biner, citra hanya akan mempunyai 2 warna yakni hitam dan putih seperti gambar 4.6 (c). Dengan proses ini, citra RGB juga akan menjadi 1 matriks penyusun saja seperti gambar 4.6 (b). Citra biner (*binary image*) adalah citra yang setiap pikselnya hanya memiliki 2 kemungkinan derajat keabuan yakni 0 dan 1. Proses pembineran dilakukan dengan membulatkan keatas atau kebawah untuk setiap nilai keabuan dari piksel. Jika intensitas warna dimulai dari 0 sampai dengan 255 maka diambil nilai tengahnya yaitu 128, jika dibawah 128 maka warna akan cenderung hitam dan diatas 128 warna akan cenderung

putih. Proses binerisasi citra *grayscale* untuk menghasilkan citra biner dapat dijelaskan dengan rumus (Candra, 2011).

$$g_{x,y} = \begin{cases} 1 & \text{iff } x,y \geq T \\ 0 & \text{iff } x,y < T \end{cases}$$

Dimana $g(x,y)$ adalah citra biner dari citra *gray scale* $f(x,y)$, dan T menyatakan nilai ambang, $T = T\{x,y, A(x,y), f(x,y)\}$ dengan T tergantung pada koordinat-koordinat *pixel*. Pada gambar 4.6 (d) adalah citra biner yang menganalisa objek buah semangka atau menghitung luas area objek citra. Citra digital ini mewakili oleh sebuah matriks yang terdiri dari M kolom N baris, dimana perpotongan antara kolom dan baris disebut piksel (*pixel* = *picture element*), yaitu elemen terkecil dari sebuah citra. Piksel mempunyai dua parameter, yaitu koordinat dan intensitas atau warna. Nilai yang terdapat pada koordinat (x,y) adalah $f(x,y)$, yaitu besar intensitas atau warna dari piksel di titik itu. Oleh sebab itu, untuk mendapatkan citra biner memiliki fungsi matematis yaitu (Candra, 2011).

$$0 \leq x \leq M - 1$$

$$0 \leq y \leq N - 1$$

$$0 \leq f(x,y) \leq G - 1$$

Dimana :

M = jumlah piksel baris (row) pada array citra

N = jumlah piksel kolom (column) pada array citra

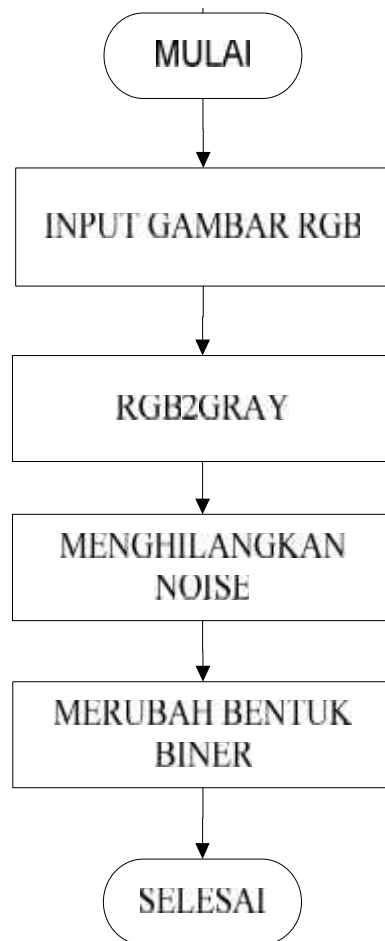
G = nilai skala keabuan (*graylevel*) (Candra, 2011).

Besarnya nilai M , N dan G pada umumnya merupakan perpangkatan dari dua seperti pada persamaan $M = 2^m$; $N = 2^n$; $G = 2^k$ Dimana nilai m , n dan k adalah bilangan bulat positif. Interval $(0,G)$ disebut skala keabuan (*grayscale*). Besar G tergantung pada proses digitalisasinya. Biasanya keabuan 0 (nol) menyatakan intensitas hitam dan 1 (satu) menyatakan intensitas putih. Untuk citra 8 bit, nilai G sama dengan $2^8 = 256$ warna (derajat keabuan), adapun sintaks program yang digunakan yaitu:

```
bw = im2bw(I, graythresh(I)); %mengubah image dari grayscale kedalam
                                bentuk biner
```

`bw2 = imfill(bw,'holes');` % fungsinya menampilkan hasil biner yang berwarna putih ditengah objek

Pada gambar 4.6 hasil tampilan yang sudah dilakukan untuk menganalisa luas area semangka, proses ini dimulai dari mengambil gambar asli selanjutnya dirubah ke bentuk *grayscale*, setelah itu mengubah gambar ke biner dan menganalisa luas area objek semangka.



Gambar 4.5. Merubah bentuk biner

Pada gambar 4.6 hasil tampilan mulai dari gambar asli sampai menganalisa luas area objek semangka. Gambar 4.6 (d) menunjukkan bahwa objek semangka untuk dianalisa luas area semangka, pada latar belakang berwarna hitam tidak terkait dalam analisa luas area objek.

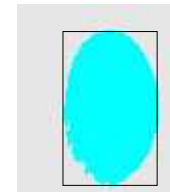
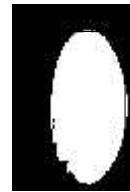
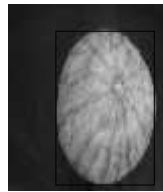


(a) Original RGB image

(b) Konversi RGB ke bentuk grayscale

(c) Merubah ke biner

(d) Citra biner



(a) Original RGB image

(b) Konversi RGB ke bentuk grayscale

(c) Merubah ke biner

(d) Citra biner

Gambar 4.6. Proses merubah bentuk biner

Pada gambar 4.7 menjelaskan tentang tahap-tahap untuk mencapai gambar 4.8 yang menujukkan hasil tampilan yang diinginkan. langkah-langkah ini dimulai dengan menentukan jenis semangka atau bentuk semangka yang didapat dari toko buah semangka. Proses ini ada lima tahapan. mulai dari gambar RGB dirubah menjadi abu-abu, menghilangkan noise, merubah bentuk biner dan menentukan jenis buah semangka, sehingga hasil tampilan dapat terlihat pada gambar 4.8.



Gambar 4.7. Menentukan jenis semangka

Gambar 4.8 menunjukan sample yang diteliti. Sample yang diteliti ada dua jenis buah semangka, yaitu sbuah semangka madu dan semangka hibrida. Gambar 4.8 (b) menunjukan bentuk atau jenis buah semangka, jika bentuk bulat semangka hibrida/lokal dan bentuk ellips semangka madu.



(a) Original RGB *image*



(b) Citra biner lingkaran



(a) Original RGB *image*

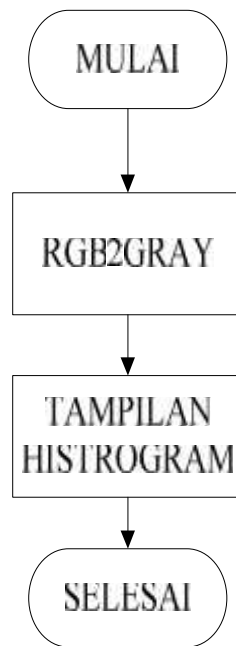
(b) Citra biner ellips

Gambar 4.8. bentuk citra

4.4. Histogram

Histogram merupakan grafik yang menunjukkan distribusi dari intensitas sebuah gambar *histogram* yang terlihat pada gambar 4.6 (b). Pada sumbu horizontal menunjukan nilai gray level sedangkan sumbu vertikal menunjukan nilai jumlah piksel gray level. Pada gambar 4.6 menunjukan bahwa citra yang gelap histogramnya cenderung ke kiri (intensitas *gray levelnya* rendah), dan citra terang cenderung kekanan (intensitas *gray levelnya* tinggi). Untuk *low contrast* agak cenderung menjahui terang dan gelap, untuk *high contrast* histogram merata pada semua *gray level*. Fungsi yang digunakan dalam matlab untuk menampilkan histogram sebuah citra adalah: $h = imhist(f, b)$. Di mana h merupakan histogram dari citra, f merupakan variabel citra dan b adalah jmlah bins yang digunakan dalam membentuk histogram. Jika tidak disebutkan maka matlab akan menggunakan nilai default, yaitu $b=256$ pada citra 8 bit.

Sintaks program yaitu: `imhist(I);`

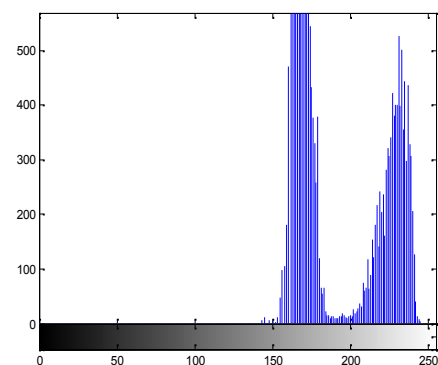


Gambar 4.9. Histogram

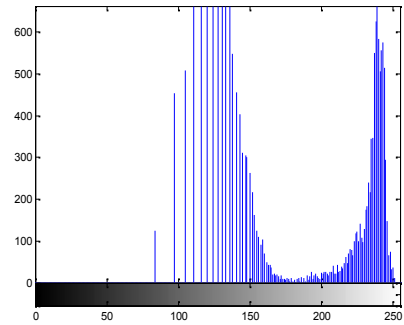
Pada gambar 4.10 tampilan hitrogram mulai dari merubah gambar grayscale dan menampilkan hitrogram setiap gambar. untuk membuat hitrogram dari gambar dengan menggunakan fungsi *imhist*. Perlu diperhatikan bahwa *imhist* hanya dapat digunakan untuk matrik gambar 1 dimensi sehingga bila diimplementasikan pada matriks gambar maka hanya berupa matriks biru.



(a) *Grayscale*



(b) Histogram



(a) *Grayscale*
















(b) *Histogram*

Gambar 4.10. Hasil Histogram

Tabel 4.1. Pengelompokan Buah Semangka Hibrida/ Lokal

Jenis buah semangka Hibrida/ Lokal


No	Jenis Semangka Hibrida/ Lokal					
	Pengelompokan Kelas Buah Semangka					
	Berat	Kelas A	Barat	Kelas B	Berat	Kelas C
1	4,0 Kg		2,2Kg		0,2Kg	
2	4,7 Kg		2,6Kg		0,3Kg	
3	4,9 Kg		2,7Kg		0,4Kg	
4	6,2 Kg		2,9Kg		0,6Kg	
5	4,8 Kg		3,0Kg		0,08Kg	

























6	5,1 Kg		3,1Kg		0,8Kg	
7	6,2 Kg		3,2Kg		0,17Kg	
8	5,8 Kg		3,4Kg		1,0Kg	
9	6,6 Kg		3,5Kg		1,6Kg	
10	7,4 Kg		3,9 Kg		1,9Kg	

Pada tabel 4.1 Pengelompokan buah semangka hibrida menurut *classification* dan data ini di dapat dari toko buah samangka.

Tabel 4.2. Pengelompokan Buah Semangka Madu

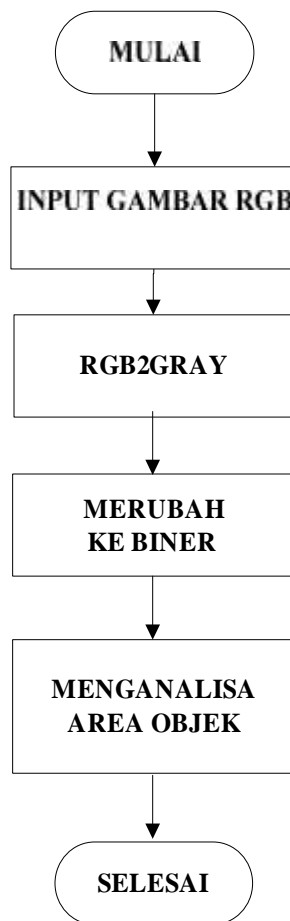
Jenis Buah Semangka Madu

No	Jenis Semangka Madu					
	Pengelompokan Kelas Buah Semangka					
	Berat	Kelas A	Barat	Kelas B	Berat	Kelas C
1	3,4 Kg		2,1 Kg		0,2 Kg	
2	4,5 Kg		2,3 Kg		0,3 Kg	

3	4,8 Kg		2,4 Kg		0,5Kg	
4	5,0 Kg		2,5 Kg		0,09 Kg	
5	5,7 Kg		2,6 Kg		0,9 Kg	
6	6,2 Kg		2,8 Kg		1,2 Kg	
7	7,5 Kg		2,9 Kg		1,4 Kg	
8	7,7 Kg		3,1 Kg		1,6 Kg	
9	8,8 Kg		3,2 Kg		1,7 Kg	
10	7,3 kg		2,4 Kg		1,8 Kg	

Pada tabel 4.2 Pengelompokan buah semangka hibrida menurut *classification* dan data ini di dapat dari toko buah samangka.

Pada gambar 4.11 tahapan sistem pengolahan citra untuk menganalisis area objek buah semangka dan melakukan klasifikasi buah semangka menurut jenis semangka dan kelas.



Gambar 4.11. Flowchat Sistem Program GUI

Berikut ini tabel hasil luas area buah semangka sehingga dikelompokkan satu persatu setiap kelas dan menurut jenis buah semangka. Setiap kelas terdiri dari 10 sample, oleh karena itu untuk jumlah keseluruhan sample 60 buah semangka.

Tabel 4.3. Pengelompokan Buah Semangka Hibrida/ Lokal

Jenis buah semangka Hibrida/ Lokal

Tabel 4.3 kelas A

Jenis buah semangka Hibrida			
No	Berat	panjang	Lebar
1	4,0 Kg	94 Piksel	104 Piksel
2	4,6 Kg	107 Piksel	100 Piksel
3	4,7 Kg	106 Piksel	102 Piksel
4	4,8 Kg	114 Piksel	96 Piksel
5	4,9 Kg	107 Piksel	105 Piksel
6	5,1 Kg	98 Piksel	116 Piksel
7	5,8 Kg	107 Piksel	117 Piksel
8	6,2 Kg	114 Piksel	114 Piksel
9	6,6 Kg	124 Piksel	109 Piksel
10	7,4 Kg	118 Piksel	124 Piksel

Nilai Maksimum	Panjang	Lebar
	124 Piksel	124 Piksel
Nilai Minimum	Panjang	Lebar
	94 Piksel	96 Piksel

Pada tabel 4.3 di dapat nilai maksimum buah semangka hibrida/ lokal kelas A untuk panjang 124 Piksel dan lebar sebesar 124 Piksel. Sedangkan untuk nilai Panjang minimum buah semangka hibrida kelas A sebesar 94 Piksel dan lebar 96 Piksel.

Tabel 4.4 Pengelompokan Buah Semangka Hibrida/ Lokal

Jenis buah semangka Hibrida/ Lokal

Tabel 4.4 kelas B

No	Berat	Panjang	Lebar
1	2,2 Kg	83 Piksel	80 Piksel
2	2,6 Kg	84 Piksel	87 Piksel
3	2,7 Kg	85 Piksel	88 Piksel
4	2,8 Kg	91 Piksel	84 Piksel
5	2,9 Kg	88 Piksel	90 Piksel
6	3,0 Kg	92 Piksel	88 Piksel
7	3,1 Kg	91 Piksel	91 Piksel
8	3,2 Kg	98 Piksel	85 Piksel
9	3,4 Kg	92 Piksel	95 Piksel
10	3,5 Kg	98 Piksel	92 Piksel

	Panjang	Lebar
Nilai Maksimum	98 Piksel	95 Piksel
	Lebar	Panjang
Nilai Minimum	83 Piksel	80 Piksel

Pada tabel 4.4 didapat nilai maksimum buah semangka hibrida/ lokal kelas B untuk panjang sebesar 98 Piksel dan lebar sebesar 95 Piksel. Sedangkan untuk nilai panjang minimum buah semangka hibrida kelas B adalah 83 Piksel dan lebar 80 Piksel.

Tabel 4.5 Pengelompokan Buah Semangka Hibrida/ Lokal

Jenis buah semangka Hibrida/ Lokal

Tabel 4.5 kelas C

Jenis buah semangka Hibrida			
No	Berat	Panjang	Lebar
1	0,2 Kg	41 Pikel	42 Pikel
2	0,3 Kg	48 Pikel	42 Pikel
3	0,4 Kg	49 Pikel	44 Pikel
4	0,6 Kg	59 Pikel	50 Pikel
5	0,08 Kg	24 Pikel	30 Pikel
6	0,8 Kg	62 Pikel	56 Pikel
7	0,1 Kg	35 Pikel	35 Pikel
8	1,9 Kg	121 Pikel	33 Pikel
9	1,0 Kg	60 Pikel	66 Pikel
10	1,6 Kg	78 Pikel	70 Pikel

	Panjang	Lebar
Nilai Maksimum	121 Pikel	70 Pikel
	Lebar	Panjang
Nilai Minimum	24 Pikel	30 Pikel

Pada tabel 4.5 didapat nilai maksimum buah semangka hibrida/ lokal kelas C untuk panjang adalah 121 Pikel dan lebar 70 Pikel. Sedangkan untuk nilai panjang minimum buah semangka hibrida kelas C adalah 24 Pikel dan lebar 30 Pikel.

Tabel 4.6 Pengelompokan Buah Semangka Madu

Jenis buah semangka Madu

Tabel 4.6 kelas A

No	Berat	Panjang	Lebar
1	3,9 Kg	127 Piksel	69 Piksel
2	4,5 Kg	135 Piksel	71 Piksel
3	4,8 Kg	67 Piksel	143 Piksel
4	5,0 Kg	135 Piksel	78 Piksel
5	5,7 Kg	152 Piksel	70 Piksel
6	6,2 Kg	155 Piksel	73 Piksel
7	7,3 Kg	92 Piksel	149 Piksel
8	7,5 Kg	149 Piksel	94 Piksel
9	7,7 Kg	120 Piksel	126 Piksel
10	8,8 Kg	132 Piksel	125 Piksel

	Panjang	Lebar
Nilai Maksimum	155 Piksel	149 Piksel
	Panjang	Lebar
Nilai Minimum	67 Piksel	69 Piksel

Pada tabel 4.6 didapat nilai maksimum buah semangka Madu kelas A untuk panjang adalah 155 Piksel dan lebar 149 Piksel. Sedangkan untuk nilai minimum buah semangka Madu kelas A panjang 67 Piksel dan lebar 69 Piksel.

Tabel 4.7 Pengelompokan Buah Semangka Madu

Jenis buah semangka Madu

Tabel 4.7 kelas B

No	Berat	Panjang	Lebar
1	2,1 Kg	112 Piksel	49 Piksel
2	2,3 Kg	105 Piksel	59 Piksel
3	2,4 Kg	101 Piksel	66 Piksel
4	2,5 Kg	103 Piksel	67 Piksel
5	2,6 Kg	59 Piksel	112 Piksel
6	2,8 Kg	84 Piksel	92 Piksel
7	2,9 Kg	121 Piksel	56 Piksel
8	3,1 Kg	111 Piksel	71 Piksel
9	3,2 Kg	107 Piksel	77 Piksel
10	3,5 Kg	113 Piksel	76 Piksel

	Panjang	Lebar
Nilai Maksimum	121 Piksel	112 Piksel
	Panjang	Lebar
Nilai Minimum	59 Piksel	49 Piksel

Pada tabel 4.7 didapat nilai maksimum buah semangka Madu kelas B untuk panjang adalah 121 Piksel dan lebar 112 Piksel. Sedangkan untuk nilai panjang minimum buah semangka Madu kelas B 59 Piksel dan lebar 49 Piksel.

Tabel 4.8 Pengelompokan Buah Semangka Madu

Jenis buah semangka Madu

Tabel 4.8 kelas C

No	Berat	Panjang	Lebar
1	0,2 Kg	51 Piksel	31 Piksel
2	0,3 Kg	51 Piksel	37 Piksel
3	0,5 Kg	63 Piksel	39 Piksel
4	0,09 Kg	30 Piksel	27 Piksel
5	0,9 Kg	82 Piksel	38 Piksel
6	1,2 Kg	76 Piksel	56 Piksel
7	1,4 Kg	81 Piksel	59 Piksel
8	1,6 Kg	93 Piksel	55 Piksel
9	1,7 Kg	66 Piksel	83 Piksel
10	1,8 Kg	127 Piksel	26 Piksel

Nilai Maksimum	Panjang	Lebar
	127 Piksel	83 Piksel
Nilai Minimum	Panjang	Lebar
	127 Piksel	26 Piksel

Pada tabel 4.8 didapat nilai maksimum buah semangka hibrida/ lokal kelas C untuk panjang 127 Piksel dan lebar 83 Piksel. Sedangkan untuk nilai panjang minimum buah semangka Madu kelas C, 127 Piksel dan lebar 26 Piksel.

4.5 Pelacakan ke Depan (*Forward Chaining*)

Secara umum, sistem pakar merupakan sistem yang mengadopsi pengetahuan manusia ke dalam komputer sehingga komputer dapat digunakan untuk menyelesaikan suatu masalah sebagaimana yang dilakukan oleh seorang pakar. Sistem pakar dibuat pada wilayah pengetahuan tertentu dan untuk suatu keahlian tertentu yang mendekati kemampuan manusia di salah satu bidang khusus. Sistem pakar mencoba mencari solusi yang memuaskan sebagaimana

yang dilakukan seorang pakar dan dapat memberikan penjelasan terhadap langkah yang diambil serta memberikan alasan atas kesimpulan yang diambil. Berikut ini sintaks *rule* pada setiap pengelompokan buah semangka.

Tabel: 4.9 Rule Buah Semangka Hibrida

Kelas	Panjang		Lebar	
	Maxsimum	Minimum	Maxsimum	Minimum
A	124 piksel	94 piksel	124 piksel	96 piksel
B	98 piksel	83 piksel	95 piksel	80 piksel
C	121 piksel	24 piksel	70 piksel	30 piksel

If Panjang $\leq 124 \geq 94$ And Lebar $\geq 124 \leq 96$ Then Class A

Else Panjang $\leq 98 \geq 83$ And Lebar $\geq 95 \leq 80$ Then Class B

Else Panjang $\leq 121 \geq 24$ And Lebar $\geq 70 \leq 30$ Then Class C

```
if tinggi  $\geq 94$  && panjang  $\leq 124$  && lebar  $\geq 96$  && lebar  $\leq 124$ 
```

```
    set(handles.edit5,'String','A');
```

```
elseif tinggi  $\geq 83$  && tinggi  $\leq 98$  && lebar  $\geq 80$  && lebar  $\leq 95$ 
```

```
    set(handles.edit5,'String','B');
```

```
elseif panjang  $\geq 24$  && tinggi  $\leq 121$  && lebar  $\geq 30$  && lebar  $\leq 70$ 
```

```
    set(handles.edit5,'String','C');
```

```
else
```

Tabel: 4.10 Rule buah Semangka Madu

Kelas	Panjang		Lebar	
	Maxsimum	Minimum	Maxsimum	Minimum
A	155 piksel	67 piksel	149 piksel	69 piksel
B	121 piksel	59 piksel	112 piksel	49 piksel
C	127 piksel	30 piksel	83 piksel	26 piksel

```
If Panjang <= 155 >= 67 And Lebar >= 149 <= 69 Then Class A
Else Panjang <= 121 >= 59 And Lebar >= 112 <= 49 Then Class B
Else Panjang <= 127 >= 30 And Lebar >= 83 <= 26 Then Class C
```

```
if tinggi >= 67 &&panjang<= 155 && lebar >= 69 && lebar <= 149
    set(handles.edit5,'String','A');
elseif tinggi >= 59 &&panjang<= 121 && lebar >= 49 && lebar <= 112
    set(handles.edit5,'String','B');
elseif tinggi >= 30 &&panjang<= 127 && lebar >= 26 && lebar <= 83
    set(handles.edit5,'String','C');
else
    set(handles.edit5,'String','Undefined');
end
```

4.6. Hasil Tampilan GUI (*Graphical User Interface*)

Aplikasi pengolahan citra untuk menganalisis dan melakukan klasifikasi buah semangka pada tugas akhir ini ditampilkan di GUI (*Graphical User Interface*). Tampilan GUI melakukan analisis dan mengelompokkan buah semangka hibrida/ lokal dan buah semangka madu, dengan jumlah keseluruhan sampel 60 buah semangka. Jenis semangka yang diteliti yaitu hibrida/ lokal dan madu. Setiap kelas masing- masing 10 buah semangka untuk mewakili kelompok kelas A, B, C, yang diimplementasikan menggunakan matlab 7.8. Hasil tampilan GUI menunjukkan bahwa parameter mutu yang sesuai untuk melakukan klasifikasi buah semangka adalah berat buah semangka. Tampilan GUI menampilkan gambar RGB merubah ke abu-abu, merubah ke biner menentukan jenis semangka, menampilkan hitrogram dan tombol-tombol yang ada.

Gambar 4.12 sampai 4.17 untuk menganalisa luas area objek buah semangka, membedakan jenis semangka, menentukan bentuk semangka dan menampilkan histogram dari gambar. Tampilan histogram untuk menunjukkan bahwa citra yang gelap histogramnya cenderung ke kiri (intensitas *gray level*nya rendah), dan citra terang cenderung kekanan (intensitas *gray level*nya tinggi).

Pada gambar 4.12 tampilan GUI (*Graphical User Interface*) untuk semangka hibrida/lokal dengan berat > 4 Kg, dengan kondisi fisik sempurna, tidak terlalu masak.



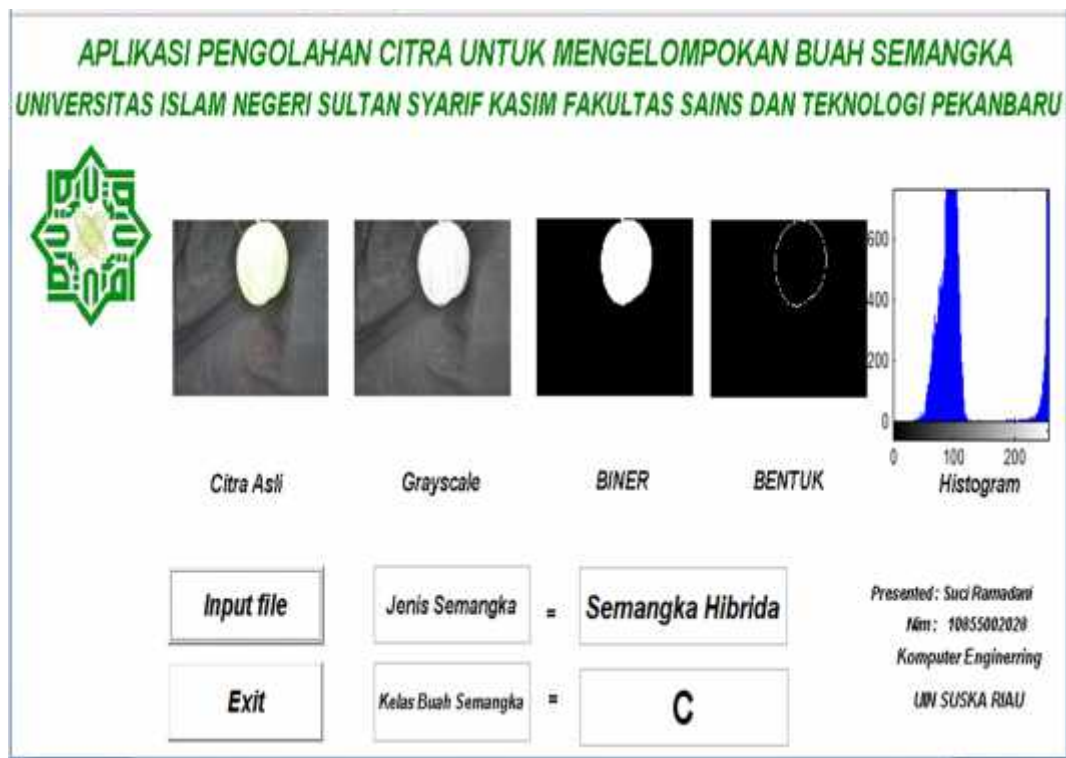
Gambar 4.12. Tampilan GUI kelas A buah semangka Hibrida/ Lokal.

Pada gambar 4.13 tampilan GUI (*Graphical User Interface*) untuk semangka hibrida/lokal Kelas B, berat $\pm 2 - 4$ kg, kondisi fisik sempurna, tidak terlalu masak.



Gambar 4.13. Tampilan GUI kelas B buah semangka Hibrida/ Lokal.

Gambar 4.14 tampilan GUI (*Graphical User Interface*) untuk semangka hibrida/lokal Kelas B, berat < 2 kg, kondisi fisik sempurna, tidak terlalu masak.



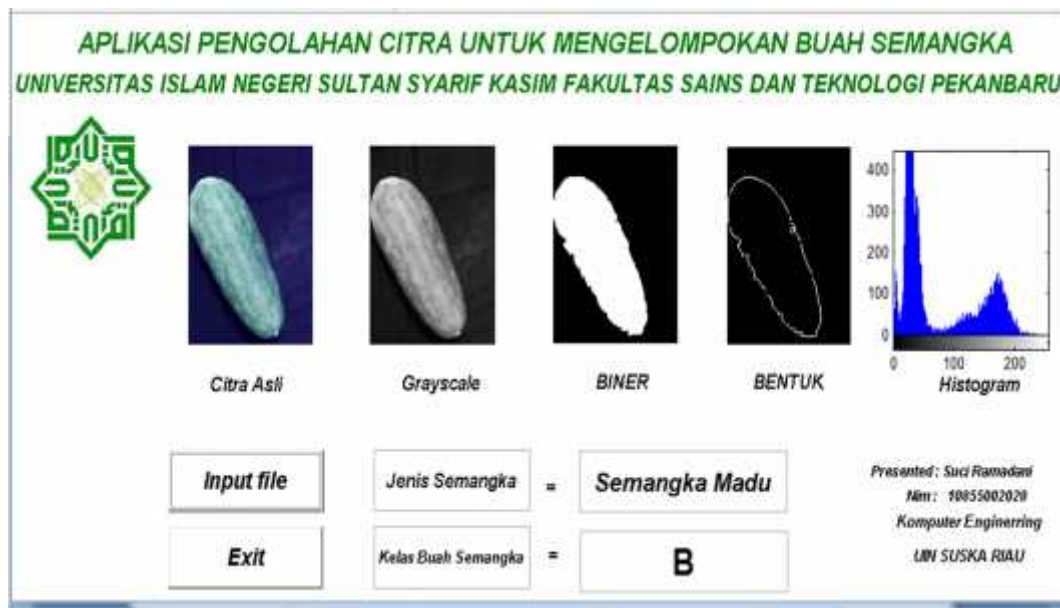
Gambar 4.14. Tampilan GUI kelas C buah semangka Hibrida/ Lokal

Pada gambar 4.15 tampilan GUI (*Graphical User Interface*) untuk semangka madu dengan berat > 4 Kg, dengan kondisi fisik sempurna, tidak terlalu masak.



Gambar 4.15. Tampilan GUI kelas A buah semangka Madu

Pada gambar 4.16 tampilan GUI (*Graphical User Interface*) untuk semangka madu Kelas B, berat $\pm 2 - 4$ kg, kondisi fisik sempurna, tidak terlalu masak.



Gambar 4.16. Tampilan GUI kelas B buah semangka Madu

Gambar 4.17 tampilan GUI (*Graphical User Interface*) untuk semangka madu Kelas B, berat < 2 kg, kondisi fisik sempurna, tidak terlalu masak.



Gambar 4.17. Tampilan GUI kelas C buah semangka Madu.

4.7 Pengujian Sistem

Pengujian sistem ini untuk mendapatkan keakuratan sistem yang digunakan, berapa persen tingkat akurasi setiap klasifikasi buah semangka. Sample untuk pengujian sistem ini berjumlah 60. Data ini diuji satu persatu dari setiap sample dan dikelompokkan menurut klasifikasi.

Tabel 4.11. Pengujian Sistem

No	Diagnosis sistem	Keakuratannya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	1
5	Gambar 5	1
6	Gambar 6	1

7	Gambar 7	1
8	Gambar 8	1
9	Gambar 9	1
10	Gambar 10	1

$$P_{akurat1} = \frac{\text{jumlah sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

Pakurat1 = persentase keakuratan semangka hibrida A

$$P_{akurat1} = \frac{10}{10} \times 100 = 90\%$$

Tabel 4.12. Pengujian Sistem

No	Diagnosis sistem	Keakuratanya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	1
5	Gambar 5	1
6	Gambar 6	1
7	Gambar 7	0
8	Gambar 8	1
9	Gambar 9	1
10	Gambar 10	1

$$P_{akurat2} = \frac{\text{jumlah sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

P akurat 2 = persentase keakuratan semangka hibrida B

$$P_{akurat2} = \frac{9}{10} \times 100 = 90\%$$

Tabel 4.13. Pengujian Sistem

No	Diagnosis sistem	Keakuratanya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	1
5	Gambar 5	1
6	Gambar 6	1
7	Gambar 7	1
8	Gambar 8	1
9	Gambar 9	1
10	Gambar 10	1

$$P_{akurat\ 3} = \frac{\text{jumlah sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

P akurat 3= persentase keakuratan semangka hibrida C

$$P_{akurat\ 3} = \frac{10}{10} \times 100 = 100\%$$

Tabel 4.14. Pengujian Sistem

No	Diagnosis sistem	Keakuratanya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	1
5	Gambar 5	1
6	Gambar 6	1
7	Gambar 7	1
8	Gambar 8	1
9	Gambar 9	0
10	Gambar 10	0

$$P_{\text{akurat 4}} = \frac{\text{jumlah sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

P akurat 4= persentase keakuratan semangka Madu A

$$P_{\text{akurat 4}} = \frac{8}{10} \times 100 = 80\%$$

Tabel 4.15. Pengujian Sistem

No	Diagnosis sistem	Keakuratanya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	1
5	Gambar 5	1
6	Gambar 6	0
7	Gambar 7	1
8	Gambar 8	1
9	Gambar 9	0
10	Gambar 10	0

$$P_{\text{akurat 5}} = \frac{\text{jumlah sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

P akurat 5 = persentase keakuratan semangka Madu B

$$P_{\text{akurat 5}} = \frac{7}{10} \times 100 = 70\%$$

Tabel 4.16. Pengujian Sistem

No	Diagnosis sistem	Keakuratanya
1	Gambar 1	1
2	Gambar 2	1
3	Gambar 3	1
4	Gambar 4	0

5	Gambar 5	1
6	Gambar 6	1
7	Gambar 7	0
8	Gambar 8	1
9	Gambar 9	0
10	Gambar 10	1

$$P_{\text{akurat 6}} = \frac{\text{jumla}^{\square} \text{ sample yang akurat}}{\text{Total sample pengujian}} \times 100$$

Pakurat 6 = persentase keakuratan semangka Madu C

$$P_{\text{akurat 6}} = \frac{7}{10} \times 100 = 70\%$$

4.8 Tingkat Akurasi dan Kepuasan Terhadap Sistem

Pada gambar 18 dan 19 menggambarkan sistem, terhadap responden bagaimana pendapat tentang sistem yang telah dibuat. Pada tabel 4.17 dan 4.18 menjumlahkan pendapat responden terhadap sistem. Sehingga memberikan informasi yang jelas kepada responden yang berupa, *usability*, *simplicity* dan *interactivity*. Setiap pertanyaan dan jawaban berdasarkan skaladi bawah ini:

Very pool				Very
Good				
1	2	3	4	5

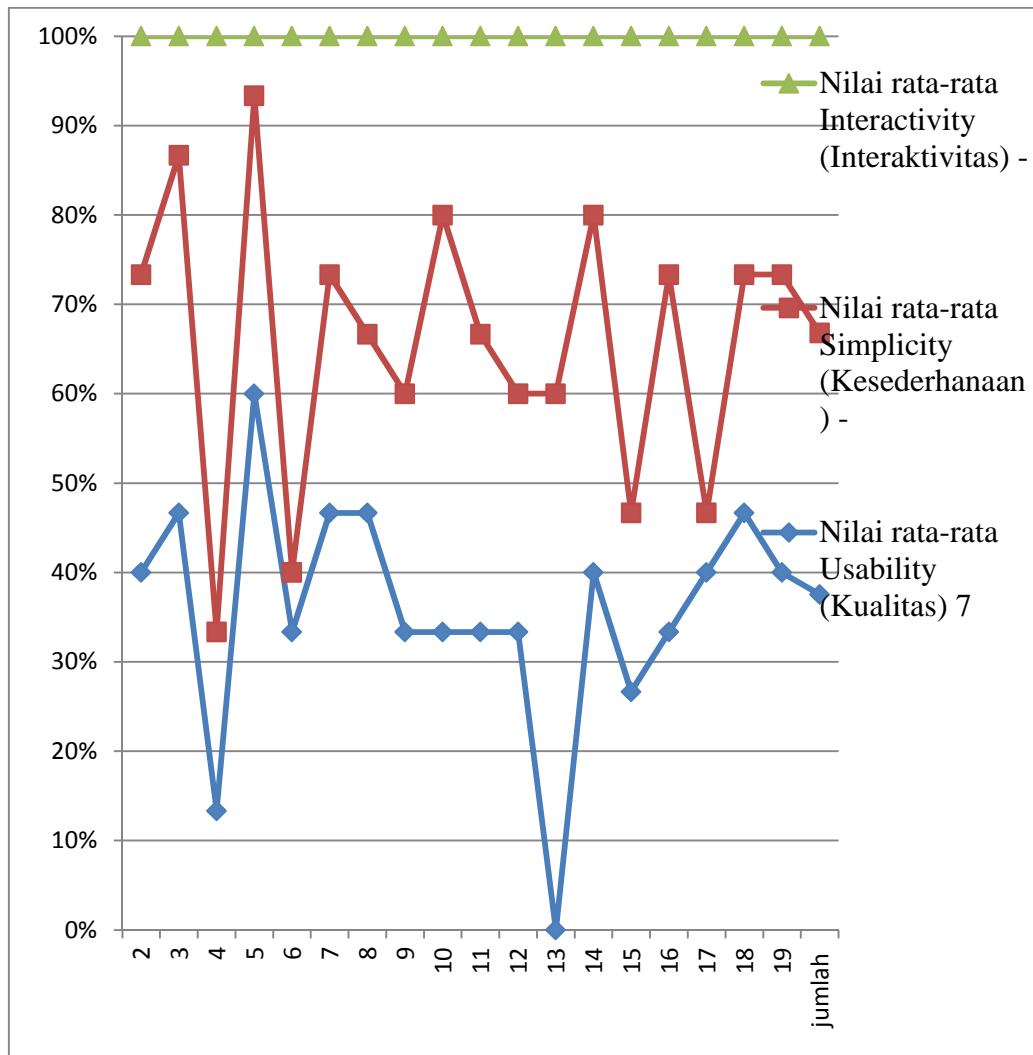
Tabel 4.17. Nilai rata-rata *Usability*, *Simplicity*, dan *Interactivity* dalam setiap pertanyaan pada responden.

No	Nilai rata-rata		
	<i>Usability</i> (Kualitas)	<i>Simplicity</i> (Kesederhanaan)	<i>Interactivity</i> (Interaktivitas)
1	(1 5)	(4)	-
2	(1 8)	(1)	-

3	(1 6)	(2)	(1)
4	(5)	(4)	(10)
5	(1)	(3)	(15)
6	(1)	(5)	(13)
7	(1)	(7)	(10)
8	(8)	(5)	(7)
9	(6)	(9)	(4)
10	(4)	(9)	(6)
11	(6)	(5)	(7)
12	(4)	(5)	(1)
13	(7)	(17)	(5)
14	(2)	(9)	(8)
15	(9)	(4)	(4)
Jumlah	54	89	91

Pada tabel 4.17 adalah menjumlahkan nilai rata-rata pertanyaan responden terhadap *usability*, *simplicity*, dan *interactivity* sehingga terdapat jumlah *usability* adalah 54, *simplicity* yaitu 89, dan *interactivity* adalah 91. Nilai rata-rata ini diambil dari jumlah pertanyaan untuk responden yang terdiri dari 15 pertanyaan, sehingga responden mengisi kolom yang tersedia dalam pendapatnya masing-masing terhadap pernyataan dan sistem yang telah dibuat.

Gambar 4.18 grafik yang menunjukkan nilai rata-rata responden setiap pertanyaan sehingga menunjikan berapa banyak rensponden yang memberikan pendapat terhadap *Usability*, *Simplicity*, dan *Interactivity*, sehingga terlihat nilai rata-rata terhadap grafik 4.18.



Gambar 4.18. Grafik Tingkat akurasi dan kepuasan terhadap sistem

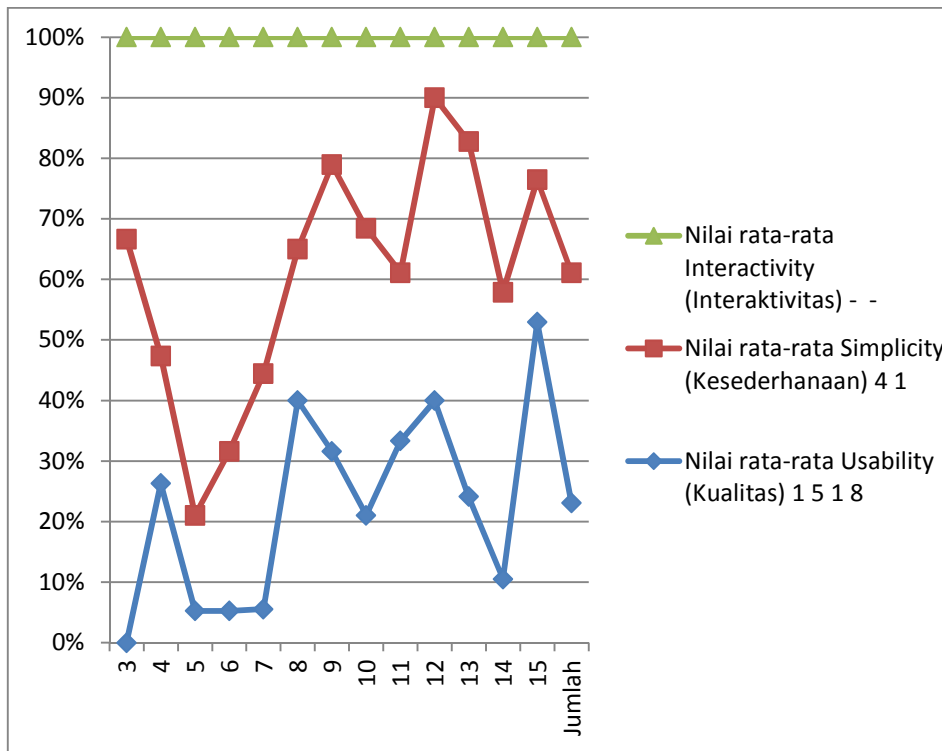
Tabel: 4.18. Nilai rata-rata responden terhadap usability, simplicity dan interactivity.

No	Nilai rata-rata		
	<i>Usability</i> (Kualitas)	<i>Simplicity</i> (Kesederhanaan)	<i>Interactivity</i> (Interaktivitas)
1	7	-	-
2	6	5	4
3	7	6	2
4	2	3	10

5	9	5	1
6	5	1	9
7	7	4	4
8	7	3	5
9	5	4	6
10	5	7	3
11	5	5	5
12	5	4	6
13	-	9	6
14	6	6	3
15	4	3	8
16	5	6	4
17	6	1	8
18	7	4	4
19	6	5	4
jumlah	104	81	92

Pada tabel 4.18 adalah menjumlahkan nilai rata-rata responden terhadap *Usability*, *Simplicity*, dan *Interactivity* sehingga jumlah *Usability* adalah 104, *Simplicity* yaitu 81, dan *Interactivity* adalah 92. Nilai rata-rata ini di ambil dari jumlah responden yang terdiri dari 19 mahasiswa/i Universitas Negeri Sultan Syarif Kasim Riau Pekanbaru, fakultas Sains dan Teknologi jurusan Teknik Elektro.

Grafik 4.19. Menunjukkan scale rata – rata pendapatresponden yang di berikan oleh *Usability*, *Simplicity*, dan *Interactivity* dalam setiap pertanyaan kepada 19 responden,



Grafik 4.19. Grafik nilai rata-rata tingkat akurasi dan kepuasan terhadap sistem