

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Seperti yang telah dilakukan pada tahap-tahap sebelumnya, implementasi dalam penelitian ini menggunakan bahasa pemrograman PYTHON 3.08.

4.1.1 *Resize Citra*

Resize citra merupakan tahap untuk mengubah ukuran citra menjadi lebih kecil dari ukuran aslinya. Kode program *resize* citra dapat dilihat pada Gambar 4.1.

```
def jerawat(self):  
    citraAwal = cv2.imread(str(self.fname))  
    citra = cv2.imread(str(self.fname))  
    citra = imutils.resize(citra, width=500)
```

Gambar 4.1 Kode Program *Resize* Citra

4.1.2 *Threshold Citra*

Threshold citra adalah proses untuk mengatur jumlah batas keabuan yang ada pada citra wajah. Proses ini juga digunakan untuk mengatur batas keabuan yang maksimal, dari batas keabuan yang maksimal ini akan dinyatakan sebagai objek. Kode program *threshold* citra dapat dilihat pada Gambar 4.2.

```
grayCitra = cv2.cvtColor(citra, cv2.COLOR_BGR2GRAY)  
ret, otsu = cv2.threshold(grayCitra, 210, 255, cv2.THRESH_OTSU)  
ret, biner = cv2.threshold(otsu, 255, 255, cv2.THRESH_BINARY)
```

Gambar 4.2 Kode Program *Threshold* Citra

Hasil dari threshold citra dapat dilihat pada Gambar 4.3.



Gambar 4.3 Hasil *Threshold* Citra

4.1.3 Menentukan Citra Wajah

Segmentasi wajah adalah suatu proses untuk memisahkan obyek dengan latar belakangnya. Teknik ini memanfaatkan warna dengan *HSV (Hue, Saturation, and Value)*. Setelah melalui tahap *resize* citra, selanjutnya citra dikonversi ke citra biner dengan menggunakan metode *Otsu*. Kode program segmentasi terlihat pada Gambar 4.4. Hasil untuk menentukan wajah citra dapat dilihat pada Gambar 4.5.

```
hsvHasilKonturMuka = cv2.inRange(hsvHasilKonturMuka, np.array(self.rangeBawahWajah,
dtype = "uint8"), np.array(self.rangeAtasWajah,dtype = "uint8"))
closing = cv2.morphologyEx(hsvHasilKonturMuka, cv2.MORPH_CLOSE, kernelWajah)
hasilKonturMuka = cv2.bitwise_and(hasilKonturMuka, hasilKonturMuka, mask = closing)
```

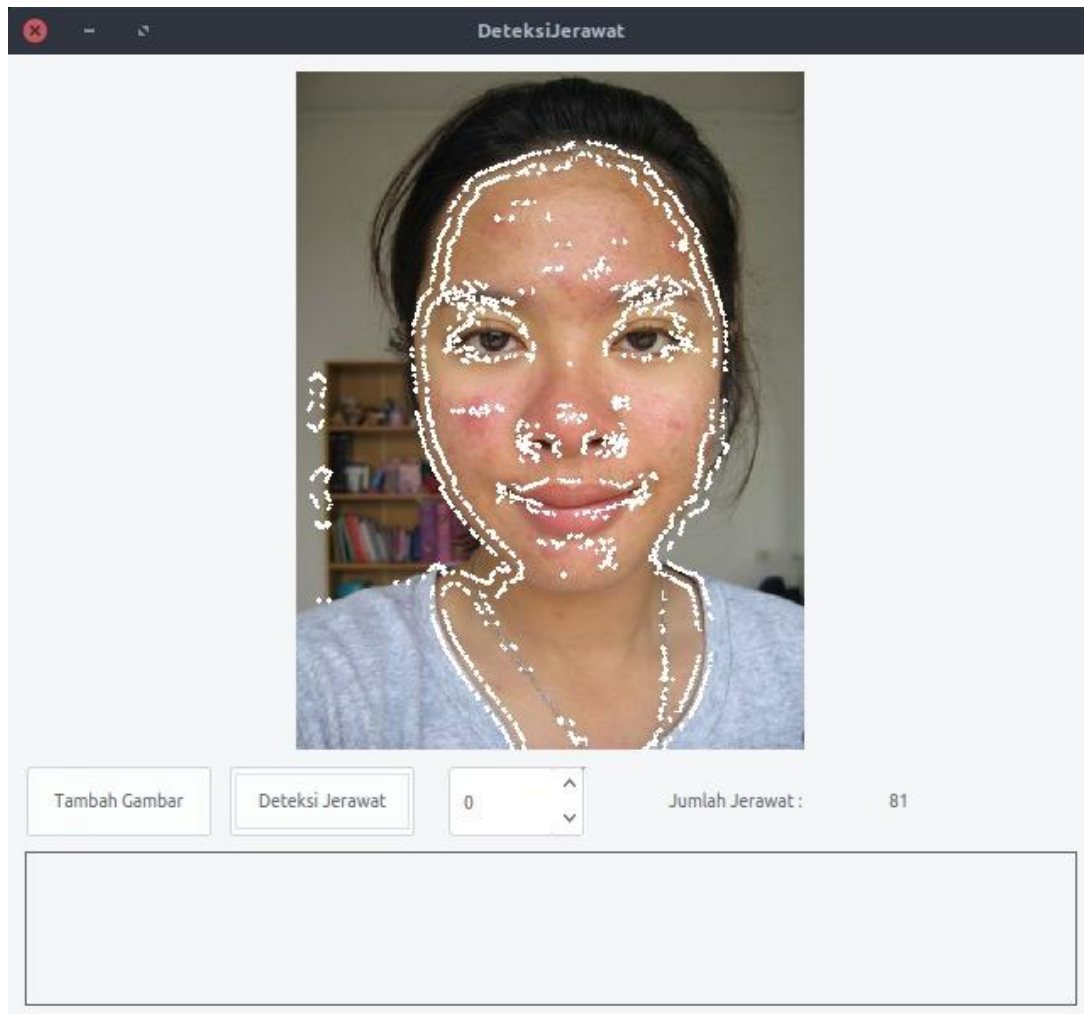
Gambar 4.4 Kode Program Segmentasi Citra



Gambar 4.5 Hasil Proses Menentukan Wajah Pada Citra

4.1.4 Menentukan Jumlah Jerawat

Menentukan jumlah jerawat merupakan proses menjumlah titik-titik pada citra wajah yang diidentifikasi sebagai jerawat pada citra wajah. Semua titik yang dianggap masuk sebagai jerawat akan otomatis masuk sebagai kandidat jerawat. Jadi, jcitra kandidat jerawat ini menggunakan perhtiungan dengan metode Laplacian Of Gaussian dapat dilihat pada Gambar 2.9. Namun, titik-titik tersebut belum tentu dianggap sebagai jerawat oleh pakar. Hasil dari menentukan jumlah jerawat dapat dilihat pada Gambar 4.6.



Gambar 4.6 Hasil Proses Menentukan Jumlah Jerawat Pada Citra

4.1.5 Nilai RGB Untuk Setiap Titik

Penulis melakukan perhitungan RGB manual disesuaikan dengan hasil identifikasi jenis jerawat oleh ketiga pakar yaitu dr. Jodi Visnu, MPH. , dr. Rosmelia, M.Kes., Sp.KK., dan dr.Hj. Nafiah Chusniayati, Sp.KK.,M.Sc.

Hasil perhitungan nilai RGB pada setiap citra wajah dapat dilihat pada Gambar 4.7 dan Gambar 4.8.

Citra wajah		RGB pada titik wajah			Rata-rata
Gambar Ke-	Titik pada wajah	R	G	B	
24	1	208,44	181,14	168,03	185,87
	2	202,06	172,23	158,87	177,72
	3	207,67	179,52	162,33	183,17
	4	192,01	165,57	149,09	168,89
	7	195,72	163,73	144,24	167,90
	9	202,22	178,06	161,6	180,63
	10	195,29	167,32	146,64	169,75
	11	180,85	153,16	130,35	154,79
	13	174,83	147,85	124,48	149,05
	14	174,86	148,14	124,89	149,30
	15	169,87	142,82	119,42	144,04
	16	154,98	152,78	148,92	152,23
	17	180,04	155,32	135,65	157,00
	18	150,38	144,9	140,52	145,27
	19	204,9	159,34	136,7	166,98
	25	216,69	168,47	151,73	178,96
	26	206,35	158,37	142,35	169,02
	27	219,93	174,44	156,26	183,54
	28	200,08	146,7	131,7	159,49
	29	218,11	144,35	142,7	168,39
	30	209,33	162,5	149,33	173,72
	31	198,44	147,78	121,63	155,95
	33	200,3	154,6	127,48	160,79
	34	152,75	146,71	135,14	144,87
	35	206,91	153,01	133,42	164,45
	37	223,76	177,01	158,02	186,26
	38	203,45	140,91	125,3	156,55
	39	217,36	174,03	159,39	183,59
	40	202,39	158,54	144,8	168,58

Gambar 4.7 Hasil RGB Pada Titik Bukan Jerawat

Citra wajah		RGB pada titik wajah			Rata-rata
Gambar Ke-	Titik pada wajah	R	G	B	
24	23	217,87	177,86	160,08	185,27
	24	220,81	171,01	155,08	182,30
	48	229,97	177,54	173,45	193,65
	62	205,45	156,34	135,85	165,88
	70	230,43	190,47	177,38	199,43
	73	223,35	177,52	157,75	186,21
	76	206,66	158,23	136,19	167,03
Rata-rata		219,22	172,71	156,54	

Gambar 4.8 Hasil RGB Pada Titik Jerawat

4.1.6 Marking Jerawat

Marking jerawat adalah proses untuk menandai jerawat dari citra yang sebelumnya telah di segmentasi. Penandaan yaitu diberi warna garis tepi jerawat yang telah di segmentasi. Warna yang digunakan untuk adalah warna yang berbeda dengan warna kulit atau jerawat agar dapat dengan mudah dikenali. Kode program *marking* jerawat dapat dilihat pada Gambar 4.9.

```

if fiturR[titik] >= self.minMaxWhiteHeadKomedo[0] and fiturR[titik] <=
self.minMaxWhiteHeadKomedo[3] and fiturG[titik] >= self.minMaxWhiteHeadKomedo[1] and
fiturG[titik] <= self.minMaxWhiteHeadKomedo[4] and fiturB[titik] >=
self.minMaxWhiteHeadKomedo[2] and fiturB[titik] <= self.minMaxWhiteHeadKomedo[5]:
    print("jenis minMaxWhiteHeadKomedo "+str(titik))
    hasilJenisJerawat = cv2.drawContours(citra, cnt, -1,
(255,255,255), 3)
    jumlahJerawatWhiteHeadKomedo =
jumlahJerawatWhiteHeadKomedo + 1
    if whiteHeadKomedo :
        nilaiWhiteHeadKomedo = "jenis jerawat yang terjadi
karena pori-pori tersumbat oleh minyak dan sel kulit mati. Whitehead Komedo ditandai
dengan warna putih.\n"
        whiteHeadKomedo = False

    if fiturR[titik] >= self.minMaxNodul[0] and fiturR[titik] <=
self.minMaxNodul[3] and fiturG[titik] >= self.minMaxNodul[1] and fiturG[titik] <=
self.minMaxNodul[4] and fiturB[titik] >= self.minMaxNodul[2] and fiturB[titik] <=
self.minMaxNodul[5]:
        print("jenis minMaxNodul "+str(titik))
        hasilJenisJerawat = cv2.drawContours(citra, cnt, -1,
(0,255,255), 3)
        jumlahJerawatNodul = jumlahJerawatNodul + 1
        if nodul :
            nilaiNodul = "jerawat yang menyebabkan rasa sakit,
jerawat jenis ini dimulai dari pori-pori yang tersumbat dan terinfeksi oleh bakteri.
Nodul ditandai dengan warna kuning.\n"
            nodul = False

        if fiturR[titik] >= self.minMaxPustul[0] and fiturR[titik]
<= self.minMaxPustul[3] and fiturG[titik] >= self.minMaxPustul[1] and fiturG[titik]
<= self.minMaxPustul[4] and fiturB[titik] >= self.minMaxPustul[2] and fiturB[titik]
<= self.minMaxPustul[5]:
            print("jenis minMaxPustul "+str(titik))
            hasilJenisJerawat = cv2.drawContours(citra, cnt, -1,
(255,0,255), 3)
            jumlahJerawatPustul = jumlahJerawatPustul + 1
            if pustul :
                nilaiPustul = "jerawat yang memiliki benjolan, bagian
atas kulit berrnanah berwarna kemerahan yang meradang. Pustul ditandai dengan warna
ungu.\n"
                pustul = False

            if fiturR[titik] >= self.minMaxPapul[0] and fiturR[titik] <=
self.minMaxPapul[3] and fiturG[titik] >= self.minMaxPapul[1] and fiturG[titik] <=
self.minMaxPapul[4] and fiturB[titik] >= self.minMaxPapul[2] and fiturB[titik] <=
self.minMaxPapul[5]:
                print("jenis jerawat papul "+str(titik))
                hasilJenisJerawat = cv2.drawContours(citra, cnt, -1,
(0,0,255), 3)
                jumlahJerawatPapul = jumlahJerawatPapul + 1
                if papul :
                    nilaiPapul = "jerawat yang muncul di bagian bawah
permukaan kulit, jika disentuh seperti tonjolan yang padat dan menyakitkan. Papul
ditandai dengan warna merah.\n"
                    papul = False

            titik = titik + 1

        totalJerawat = jumlahJerawatPapul + jumlahJerawatPustul +
jumlahJerawatNodul + jumlahJerawatWhiteHeadKomedo
        jumlahJerawatNodul = jumlahJerawatNodul / totalJerawat * 100
        jumlahJerawatPustul = jumlahJerawatPustul / totalJerawat * 100
        jumlahJerawatPapul = jumlahJerawatPapul / totalJerawat * 100
        jumlahJerawatWhiteHeadKomedo = jumlahJerawatWhiteHeadKomedo /
totalJerawat * 100

```

```
# print("Nodul : "+ str(int(jumlahJerawatNodul)) + " Pustul : "+
str(int(jumlahJerawatPustul)) + " Papul : " + str(int(jumlahJerawatPapul)) + " WHK
: " + str(int(jumlahJerawatWhiteHeadKomedo)))
self.rgbNilai.setText("Whitehead      Komedo      (" +
str(jumlahJerawatWhiteHeadKomedo) + "%) : " + nilaiWhiteHeadKomedo + "Papul (" +
str(jumlahJerawatPapul) + "%) : " + nilaiPapul+ "Pustul (" + str(jumlahJerawatPustul)
+ "%) : " + nilaiPustul+ "Nodul : " + str(jumlahJerawatNodul) + "%) : " + nilaiNodul)
```

Gambar 4.9 Kode Program *Marking Jerawat*

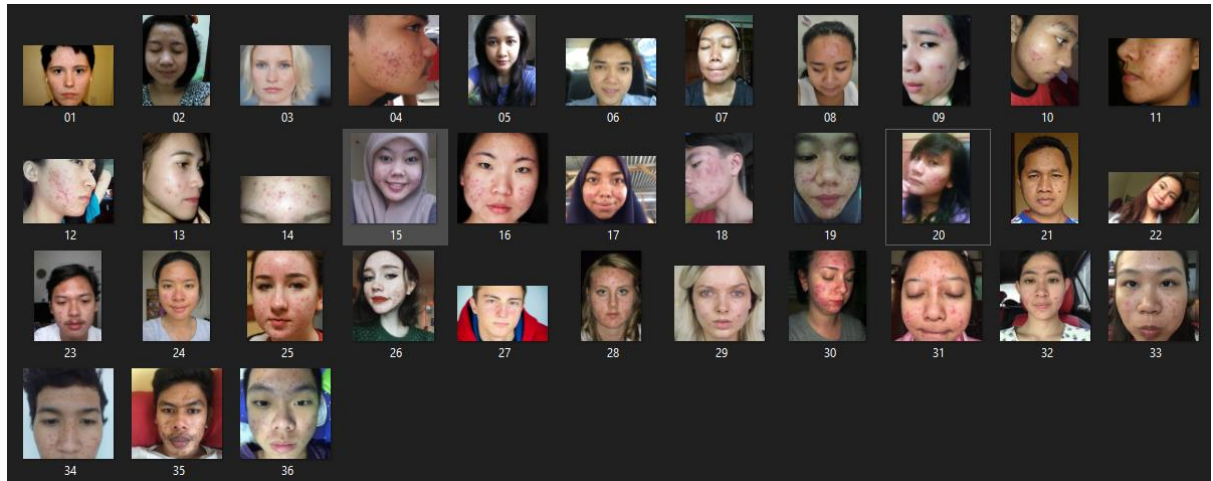
Penandaan jerawat pada citra menggunakan metode *Edge Detection*. *Edge Detection* adalah salah satu teknik pemrosesan gambar untuk mendapatkan atau mengetahui titik terjauh yang terletak pada objek, tetapi juga dapat digunakan untuk menganalisis bentuk objek yang terdeteksi karena teknik ini hanya akan menampilkan bentuk dalam bentuk titik-titik yang mengikuti bentuk objek dan menampilkan tepi dari objek yang terdeteksi. Salah satu metode yang dapat digunakan untuk detektor tepi adalah *laplacian*. Hasil dari *Edge Detection* dapat dilihat pada Gambar 4.10.

Gambar 4.10 Hasil Proses *Marking Jerawat*

4.1.7 Identifikasi Jenis Jerawat

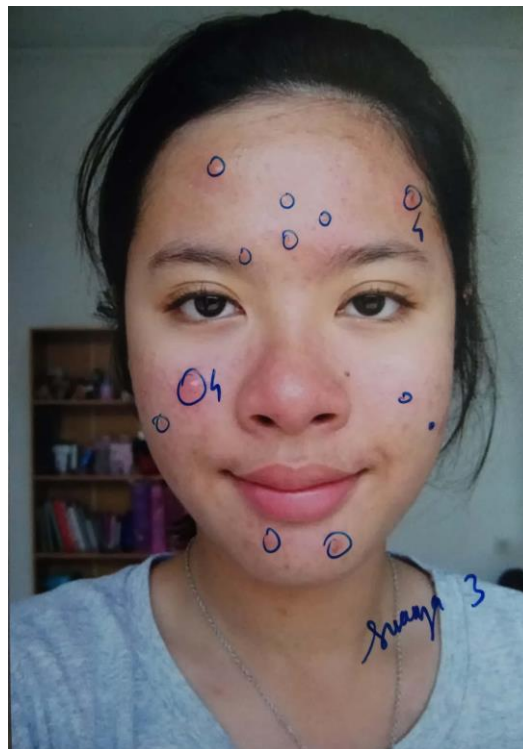
Identifikasi jenis jerawat pada penelitian ini penulis menggunakan 36 gambar wajah jerawat sebagai data pelatihan. Data ini didapatkan dari AVO. Penulis mengambil 36 foto yang semuanya merupakan wajah berjerawat dengan wajah berjerawat dengan karakteristik wajah close-up tunggal menghadap depan atau ke arah kamera untuk dijadikan sampel. Sampel foto yang akan diidentifikasi dari individu-individu yang berbeda dan latar belakang (*background*) foto yang berbeda juga. Sampel foto tersebut juga memiliki varian jenis kelamin dan warna kulit yang berbeda.

Data yang didapatkan dari AVO berjumlah 36 foto wajah berjerawat dapat dilihat pada Gambar 4.11.



Gambar 4.11 Data Identifikasi Jenis Jerawat

Contoh hasil identifikasi dari ketiga dokter spesialis kecantikan dan kelamin terlihat pada Gambar 4.12.



(a)



(b)



(c)

Gambar 4.12 Data Citra ; (a) Hasil Identifikasi dr. Jodi. (b) Hasil Identifikasi dr. Rosmelia.
(c) Hasil Identifikasi dr. Nafiah.

Kode jenis jerawat yang telah diidentifikasi oleh ketiga pakar terlihat pada Tabel 4.1.

Tabel 4.1 Tabel Kode Jenis Jerawat

Hasil Identifikasi Dokter Berdasarkan Kode Jenis Jerawat

Kode Tipe Jerawat	Keterangan
1	<i>Blackhead</i> Komedo
2	<i>Whitehead</i> Komedo
3	Papul
4	Pustul
5	Nodul

Hasil nilai RGB yang terdapat pada citra wajah dengan masing-masing titik koordinat sebagai jenis jerawat berdasarkan hasil identifikasi dokter seperti yang terlihat pada Gambar 4.13.

Citra wajah		RGB pada titik wajah			Rata-rata	Identifikasi Jenis Jerawat			Kesimpulan Jenis Jerawat	Kesimpulan	Keterangan
Gambar Ke-	Titik pada wajah	R	G	B		Dokter 1	Dokter 2	Dokter 3			
24	23	217,9	177,9	160,1	185,27	3	3	3	3	Pada image acne face data ke-24 jenis jerawat pada wajah paling banyak ditemui yaitu Papul dan Nodul.	Terdapat 1 titik dengan pendapat dokter yang berbeda, yaitu pada titik 73.
	24	220,8	171	155,1	182,30	3	3	3	3		
	48	230	177,5	173,5	193,65	4	5	5	5		
	62	205,5	156,3	135,3	165,88	3	-	3	3		
	70	230,4	190,5	177,4	199,43	3	-	3	3		
	73	223,4	177,5	157,8	186,21	3	2	-	-		
	76	206,7	158,2	136,2	167,03	3	3	4	3		
Rata-rata		219,2	172,7	156,5							

Gambar 4.13 Hasil RGB Setiap Titik Koordinat

Hasil RGB tersebut didapatkan dari ketiga dokter spesialis kecantikan dan kelamin yang menjadi indikator untuk membedakan antara jenis jerawat yang satu dengan yang lainnya. Rentang nilai RGB setiap jenis jerawat yang didapat dari ketiga dokter terlihat pada Tabel 4.2 dan Tabel 4.3.

Tabel 4.2 Tabel Rentang Nilai Minimal RGB Jenis Jerawat Dari Dokter

No	Jenis Jerawat	Nilai <i>Red</i>	Nilai <i>Green</i>	Nilai <i>Blue</i>
1.	<i>Whitehead</i> Komedo	204	178	155
2.	Papul	160	122	107
3.	Pustul	160	115	98
4.	Nodul	184	113	110

Tabel 4.3 Tabel Rentang Nilai Maksimal RGB Jenis Jerawat Dari Dokter

No	Jenis Jerawat	Nilai <i>Red</i>	Nilai <i>Green</i>	Nilai <i>Blue</i>
1.	<i>Whitehead</i> Komedo	217	194	170
2.	Papul	249	222	215
3.	Pustul	237	221	215
4.	Nodul	239	206	200

Pada Tabel 4.2 dan Tabel 4.3 nilai yang didapatkan dari hasil identifikasi ketiga dokter spesialis kulit dan kelamin jika dimasukkan pada koding sistem hasilnya belum mendekati pendapat pakar yaitu tiga dokter spesialis kulit dan kelamin, maka nilai batasnya pada koding sistem menjadi seperti yang terlihat pada Tabel 4.4 dan Tabel 4.5.

Tabel 4.4 Tabel Rentang Nilai Minimal RGB Jenis Jerawat Pada Sistem

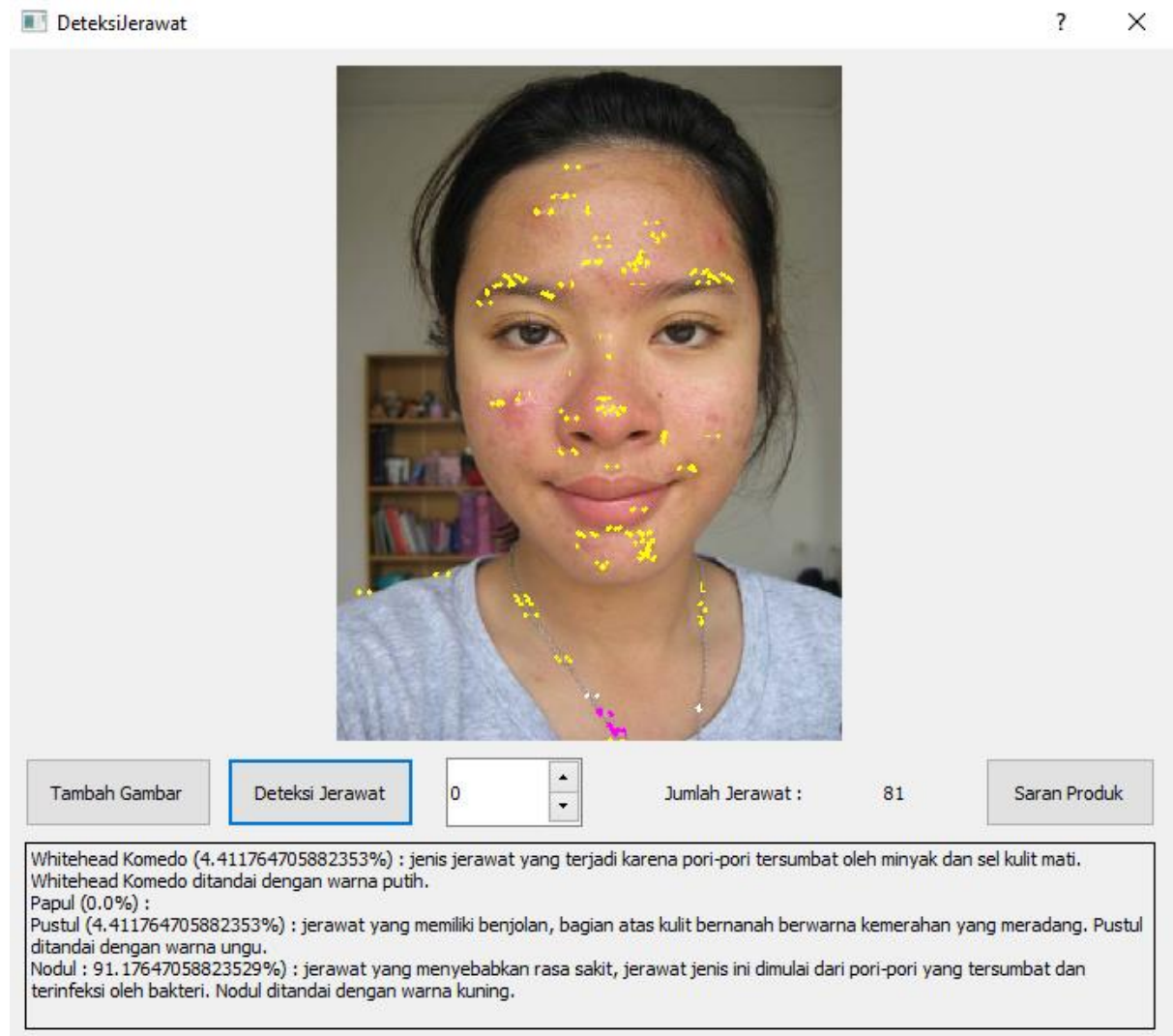
No	Jenis Jerawat	Nilai <i>Red</i>	Nilai <i>Green</i>	Nilai <i>Blue</i>
1.	<i>Whitehead</i> Komedo	204	178	154
2.	Papul	176	132	111
3.	Pustul	208	115	98
4.	Nodul	183	113	109

Tabel 4.5 Tabel Rentang Nilai Maksimal RGB Jenis Jerawat Pada Sistem

No	Jenis Jerawat	Nilai <i>Red</i>	Nilai <i>Green</i>	Nilai <i>Blue</i>
1.	<i>Whitehead</i> Komedo	217	193	170
2.	Papul	207	161	139
3.	Pustul	227	197	188
4.	Nodul	238	176	165

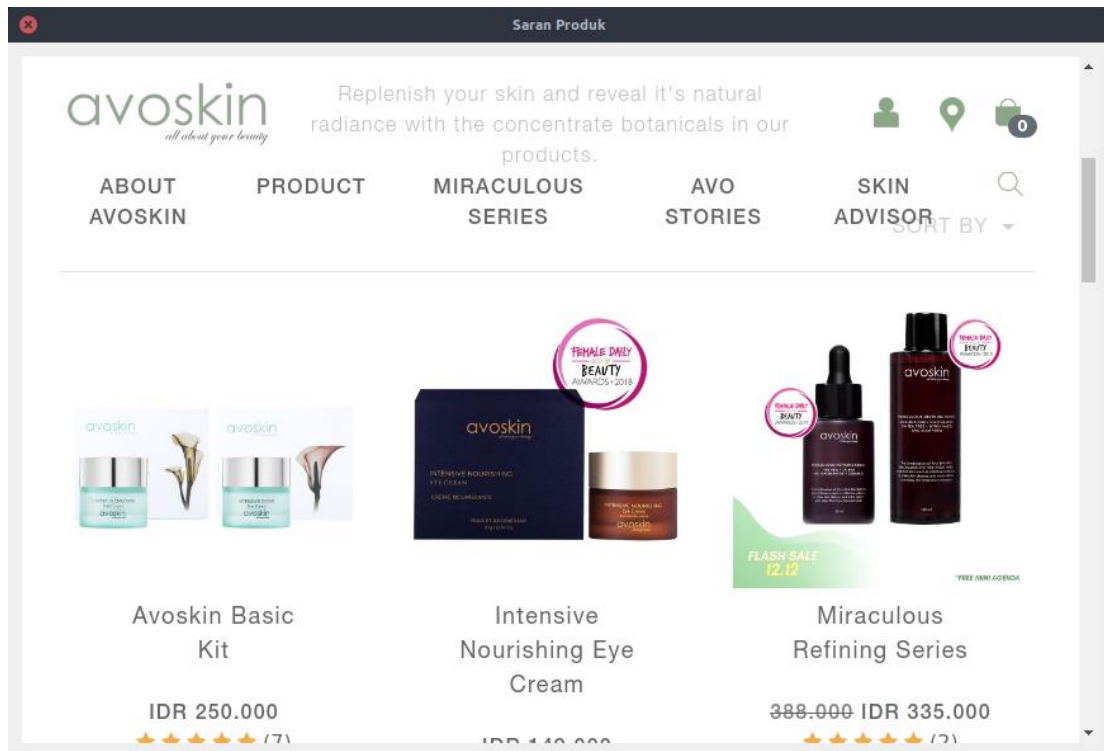
4.2 Implementasi Antarmuka

Antarmuka merupakan jembatan antara pengguna dengan sistem untuk memudahkan pengguna berinteraksi dengan sistem. Implementasi antarmuka pada penelitian ini memiliki dua halaman yaitu halaman utama dan halaman saran produk. Untuk halaman utama dapat dilihat pada Gambar 4.14.



Gambar 4.14 Halaman Utama Antarmuka

Hasil dari halaman utama antarmuka terdapat kotak keterangan yang berisi informasi mengenai jenis jerawat yang terdapat pada wajahnya dengan keterangan warna penandaan serta berisi informasi jumlah persentasi jenis jerawat terhadap jumlah jerawat. Contohnya pada kotak tersebut terdapat angka 4,411764705882353% angka tersebut merupakan jumlah persen jenis jerawat Whitehead Komedo pada total jumlah jerawat yang terdapat pada wajah. Untuk halaman saran produk AVO seperti yang terlihat pada Gambar 4.15.



Gambar 4.15 Saran Produk

Hasil dari implementasi antarmuka, terdapat penjelasan mengenai tampilan kedua halaman tersebut antara lain:

1. Tambah Gambar: Digunakan untuk mengambil citra yang akan diidentifikasi dari data yang telah tersimpan pada laptop.
2. Deteksi Jerawat: Tombol deteksi digunakan untuk mendeteksi citra yang telah diambil dari data yang tersimpan pada laptop.
3. Jumlah Jerawat: Berfungsi untuk menampilkan total jumlah jerawat pada setiap citra wajah berjerawat.
4. Kotak Status: Berfungsi untuk memunculkan penjelasan dari jenis jerawat yang terdeteksi oleh sistem.
5. Saran Produk: Berfungsi untuk memberi informasi produk dari AVO kepada pengguna.

4.3 Pengujian Sistem

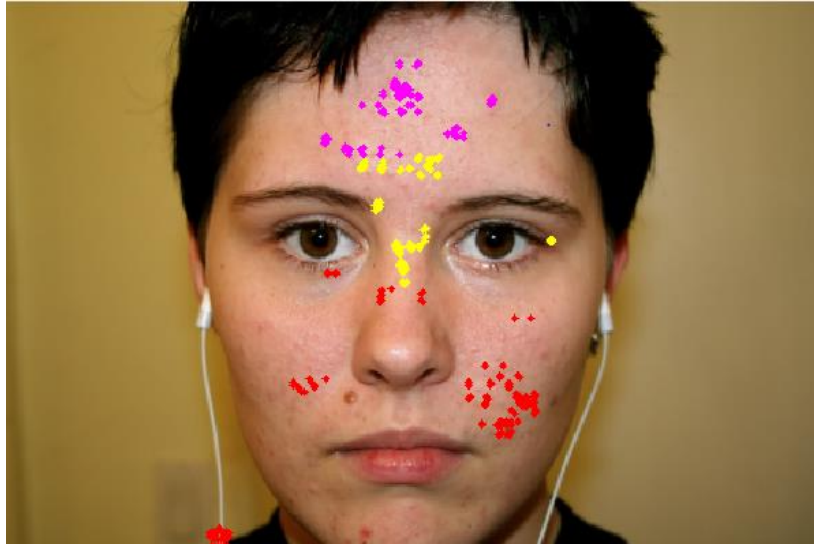
Pengujian sistem dilakukan dengan menggunakan metode *Black Box Testing* dan *Single Decision Threshold*.

4.3.1 Pengujian Sistem *Black Box Testing*

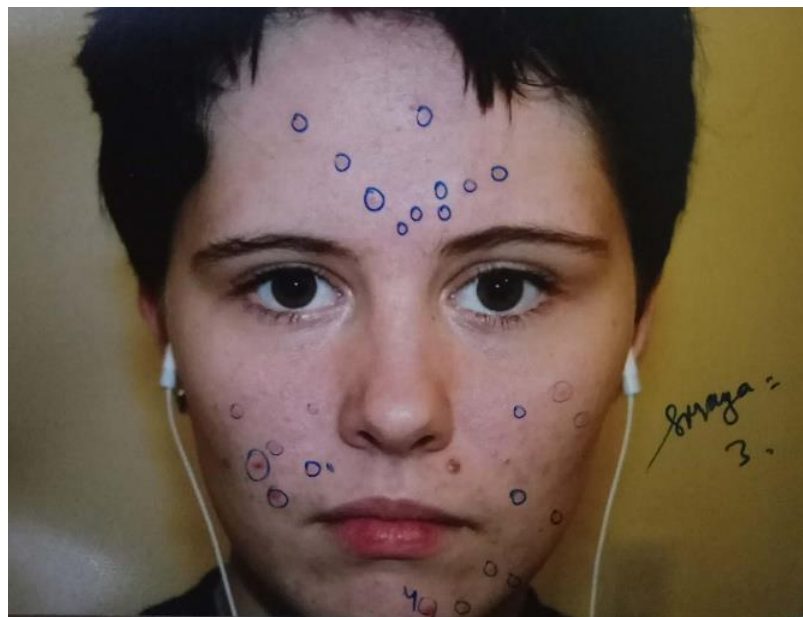
Metode *Black Box* dilakukan oleh penulis secara langsung untuk menguji sistem yang telah dibuat. Hasil pengujian sistem dapat dilihat pada Lampiran 4.1.

4.3.2 Pengujian Sistem *Single Decission Threshold*

Pengujian sistem dilakukan dengan cara membandingkan hasil deteksi sistem dengan citra yang telah ditandai oleh pakar, seperti pada Gambar 4.16.



(a)



(b)

Gambar 4.16 Perbandingan Penandaan; (a) Sistem. (b) Pakar

Pengujian sistem pada penelitian ini menggunakan *Single Decission Threshold* dengan menggunakan 3 parameter yaitu *sensitivity*, *specificity*, dan *accuracy*. Berikut table hasil uji sistem dengan membandingkan hasil penandaan oleh pakar.

Metode *Single Decission Threshold* dilakukan oleh penulis secara langsung untuk menguji validitas sistem yang telah dibuat. Hasil pengujian sistem dapat dilihat pada Lampiran 4.2.

4.4 Perbandingan Hasil Sistem Dengan Pakar

Perbandingan *output* yang didapat dari sistem dengan pendapat beberapa pakar akan menentukan bahwa sistem sudah mampu menentukan analisis terhadap jenis jerawat secara tepat. Dari 36 data yang didapatkan jenis jerawat yang sering muncul yaitu tiga jenis jerawat antara lain papul, pustul, dan nodul. Pada sistem jenis jerawat papul ditandai dengan warna merah, jenis jerawat pustul ditandai dengan warna ungu, dan jenis jerawat nodul ditandai dengan warna kuning. Hasil perbandingan dapat dilihat pada Lampiran 4.3.

4.5 Kelebihan dan Kekurangan Sistem

a. Kelebihan Sistem

- Sistem dapat menentukan jumlah titik yang dianggap sebagai jerawat pada setiap citra wajah berjerawat.
- Sistem dapat memproses citra dengan ukuran dimensi yang besar.
- Deteksi jenis jerawat dengan jumlah jerawat sedikit sudah bagus.
- Sistem dapat menentukan jenis jerawat pada setiap titik koordinat jerawat.

b. Kekurangan Sistem

- Jika pencahayaan pada citra tidak merata, hasil yang didapatkan kurang bagus.
- Identifikasi sistem terhadap jenis jerawat masih rendah.
- Sistem belum mampu mendeteksi jerawat secara akurat.