

# Pengukuran Cacat Tekstur Pada Kulit Jeruk Keprok dengan Klasifikasi *Pixel*

Helmi Fitria Nugroho – NIM: A11.2011.06492

Program Studi Teknik Informatika S1, Fakultas Ilmu Komputer,  
Universitas Dian Nuswantoro Semarang

Email: 111201106492@mhs.dinus.ac.id

## Abstrak

Persaingan pasar untuk buah-buahan saat ini menjadi semakin ketat dengan masuknya impor buah-buahan dari luar negeri ke pasar lokal. Otomatisasi dalam proses produksi diharapkan dapat mempercepat proses produksi dan meningkatkan kualitas buah-buahan lokal sehingga mampu bersaing dengan buah-buahan impor. Teknologi *computer vision* telah diterapkan di bidang produksi pangan untuk otomatisasi pemeriksaan kualitas bahan pangan. Proses ini didasarkan pada pengolahan dan analisis citra bahan pangan. Pada skripsi ini dibahas penggunaan analisis citra untuk mengukur banyaknya cacat pada permukaan kulit jeruk keprok *Citrus sinensis* dengan klasifikasi *pixel* atas citra jeruk.

Kata kunci: *computer vision*, *nearest neighbor classification*, *quality control*, pengendalian mutu bahan pangan, kualitas buah jeruk keprok

## I. PENDAHULUAN

Di tengah pasar bebas saat ini, produk pangan lokal menghadapi persaingan ketat dengan produk pangan impor. Salah satu produk yang terdampak adalah buah-buahan lokal yang terpinggirkan karena masyarakat semakin menggemari buah-buahan impor.<sup>1</sup> Oleh karena itu diperlukan peningkatan efisiensi proses produksi buah-buahan lokal agar dapat bersaing dengan buah-buahan impor di pasar dalam negeri. Banyak hal yang dapat diperbaiki dalam proses produksi antara lain dengan penerapan otomatisasi untuk meningkatkan efisiensi dan akurasi pengendalian mutu produk.

Computer vision atau visi komputer adalah konstruksi deskripsi yang eksplisit dan bermakna untuk objek dari citra objek. Teknologi ini bertujuan untuk menirukan efek penglihatan manusia dengan mempersepsikan dan memahami sebuah citra secara elektronik. Bidang produksi pangan adalah salah satu sektor industri yang memperoleh manfaat dari teknologi visi komputer. Visi komputer telah berhasil diterapkan untuk pengukuran objektif berbagai produk pertanian (He, Yang, Xue, & Geng, 1998; Li & Wang, 1999) dan makanan olahan (Sun, 2000; Wang & Sun, 2001). Industri pangan menempati peringkat sepuluh besar pengguna teknologi visi komputer (Gunasekaran, 1996).[2]

Dalam penelitian ini penulis melakukan penelitian deteksi cacat tekstur buah jeruk keprok. Deteksi dilakukan dengan klasifikasi *pixel*. Objek yang diamati adalah buah jeruk keprok (*Citrus sinensis*).

## II. TINJAUAN PUSTAKA

SNI telah menetapkan standard kualitas untuk buah jeruk keprok (*Citrus sinensis*) dalam SNI 3165-2009. Standard ini diberlakukan bagi buah jeruk keprok yang dipasarkan untuk konsumsi langsung (bukan olahan atau bahan baku industri).

Standard ini meliputi standard minimum, ketentuan kemasangan, pengemasan, pelabelan, higienis, cemaran logam berat dan residu pestisida. Setelah memenuhi standard minimum tersebut, jeruk dapat diklasifikasi mutu dan ukurannya dengan batas-batas toleransi untuk tiap kelas mutu[11].

Berdasarkan standard tersebut, buah jeruk keprok dibagi menjadi tiga kelas: kelas super yang bebas dari kerusakan, kelas A dengan total area cacat maksimum 10% dan kelas B dengan total area cacat maksimum 15%.

Citra (*image*) adalah representasi spasial dari suatu objek yang sebenarnya dalam bidang dua dimensi yang biasanya ditulis dalam koordinat kartesius  $x - y$ , dan setiap koordinat merepresentasikan satu sinyal terkecil dari objek[6]. Citra digital adalah fungsi intensitas warna dua dimensi  $f(x, y)$ , dengan  $x$  dan  $y$  mewakili koordinat lokasi suatu titik dan nilai dari fungsi yang merupakan tingkat intensitas warna atau tingkat keabu-abuan dari titik tersebut[10]. Citra digital merupakan representasi dari suatu objek nyata yang dapat dikenali oleh komputer.

Pengolahan citra (*image processing*) merupakan suatu sistem di mana proses dilakukan dengan masukan berupa citra dan hasilnya juga berupa citra[1]. Pada awalnya pengolahan citra hanya dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya teknologi informasi serta meningkatnya kapasitas kemampuan komputer, pengolahan citra mulai dilibatkan dalam pengambilan informasi dari citra digital.

Computer vision adalah bidang studi yang mempelajari metode-metode pengumpulan, pemrosesan, analisis dan penafsiran citra dari dunia nyata dengan tujuan untuk menghasilkan informasi numerik atau simbolis.

### A. Nearest Neighbor Search

*Nearest Neighbor Search* (NNS) adalah optimisasi untuk menemukan titik data yang paling mirip. Ukuran kesamaan antara titik-titik data dengan  $n$ -atribut dapat dideskripsikan

<sup>1</sup>(<http://technology-indonesia.com/home/201-buah-lokal-vs-buah-impor>)

dalam fungsi ketidaksamaan (*dissimilarity function*)[5]. Pada umumnya data dimodelkan sebagai titik-titik pada ruang vektor dimensi- $n$ . Salah satu perhitungan yang digunakan untuk mengukur ketidaksamaan antardata adalah rumus jarak Euclid.

$$d = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \quad (1)$$

$a_i$  dan  $b_i$  adalah atribut ke- $i$  dari  $a$  dan  $b$ .

Untuk keperluan klasifikasi, tiap kelas data diwakili oleh sebuah titik pusat (*centroid*) yang berada di tengah sebuah kelas, dan data yang diuji diklasifikasikan ke kelas yang memiliki titik pusat terdekat dengan data tersebut. Perhitungan jarak Euclidean digunakan untuk mengklasifikasikan *pixel* pada citra jeruk untuk deteksi daerah cacat. Tiap *pixel* diukur jaraknya dari nilai intensitas rata-rata tiap kelas.

### B. Sistem Warna RGB

Sistem warna atau model warna adalah spesifikasi sistem koordinat dan subruang dalam koordinat tersebut dengan tiap warna direpresentasikan sebagai sebuah titik. Sistem warna RGB memodelkan semua warna sebagai kombinasi tiga warna primer yaitu merah (R), hijau (G) dan biru (B) dalam berbagai intensitas. Dalam model ini, sebuah koordinat kartesius 3-dimensi menunjukkan besarnya intensitas tiap komponen warna. Warna-warna digambarkan sebagai titik-titik di dalam sebuah kubus yang meliputi seluruh kombinasi nilai ketiga komponen, setiap warna dideskripsikan dengan vektor dari titik asal. Warna hitam terletak di titik asal koordinat, dan warna putih di sudut terjauh dari titik asal.[4]

Sistem warna RGB didasarkan pada pemahaman atas teori trikromatik penglihatan mata manusia. Dengan pemahaman ini, sistem warna ini telah diterapkan untuk fotografi film berwarna dan terus dipakai di peralatan elektronik yang mengolah dan menampilkan citra, baik secara analog maupun digital. Kelebihan sistem warna RGB adalah penggunaannya yang luas pada perangkat elektronik sehingga dapat dengan mudah ditransfer tanpa memerlukan konversi. Kelemahan sistem ini adalah tidak dapat digunakan dengan sempurna pada perangkat cetak karena sifat substraktif dari pigmen warna yang digunakan dalam proses percetakan.

*Pixel* pada citra digital merupakan kombinasi tiga nilai intensitas komponen warna primer RGB, maka tiap *pixel* dapat dilihat sebagai titik-titik data pada koordinat kartesius 3-dimensi dari sistem warna RGB. Sebagaimana titik pada ruang vektor lainnya, ukuran metrik fungsi ketidaksamaan berlaku untuk data *pixel* RGB, sehingga klasifikasi *nearest neighbor* dapat diterapkan.

### C. Metode Thresholding Otsu

*Thresholding* adalah suatu metode untuk memisahkan objek dan *background*-nya. Tujuan *thresholding* adalah untuk menemukan nilai *threshold* yang tepat untuk memisahkan objek dari *background*. Hasil *thresholding* berupa citra biner, dengan nilai 1 atau aras keabuan maksimum menggantikan nilai-nilai *pixel* di atas ambang dan nilai 0 menggantikan nilai-nilai *pixel* di bawah ambang, jika diasumsikan objek lebih terang daripada

latar belakang. Namun intensitas objek dapat diasumsikan berada di bawah ambang atau di antara dua ambang batas[9].

Salah satu metode *thresholding* adalah metode Otsu. Algoritma Otsu membagi data histogram menjadi dua kelas dengan mencari nilai ambang yang memaksimalkan selisih rata-rata data antarkelas, dengan demikian memperkecil selisih rata-rata data dalam satu kelas. Perhitungan tersebut diformulasikan sebagai berikut

$$\sigma_{\omega}^2(t) = \omega_1(t)\sigma_{\omega_1}^2(t) + \omega_2(t)\sigma_{\omega_2}^2(t) \quad (2)$$

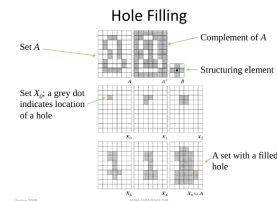
Nilai yang dihasilkan *thresholding* digunakan pada tahap klasifikasi awal *pixel* ke dalam kelas-kelas warna.

### D. Operasi Morfologi Hole Filling

Lubang (*hole*) didefinisikan sebagai daerah *background* yang dikelilingi *pixel foreground* yang saling terhubung. Pengisian lubang dilakukan dengan menandai salah satu titik pada lubang kemudian melakukan dilasi kondisional atas *structuring element B* secara iteratif, dilasi dibatasi oleh irisan dengan komplement citra biner yang diproses.

$$X_k = (X_{k-1} \oplus B) \cap A^c \quad k = 1, 2, 3, \dots \quad (3)$$

Operasi diulangi sampai  $X_k = X_{k-1}$ .  $X_k$  digabungkan dengan  $A$  untuk mendapatkan citra hasil akhir.



Gambar 1: Ilustrasi operasi *hole filling*

### E. Metode Klasifikasi Pixel untuk Citra Jeruk

Hasil klasifikasi *pixel* membagi jeruk menjadi lapisan-lapisan rentang intensitas dari yang terendah hingga rentang tertinggi. Sesuai karakteristik penyinaran pada benda bulat, intensitas tertinggi akan berada di tengah jeruk dan intensitas-intensitas terendah di tepinya. Cacat mempunyai intensitas lebih rendah daripada daerah normal di sekitarnya. Diskontinuitas pada daerah kelas dengan intensitas tertinggi menandakan adanya cacat pada bagian itu.[7]

Salah satu problem dalam klasifikasi *nearest neighbor* adalah penentuan pusat *cluster* sebagai acuan. Metode standar klasifikasi menentukan pusat *cluster* secara iteratif dari hasil klasifikasi titik-titik pada langkah sebelumnya sampai tercapai konvergensi. Penelitian ini menggunakan pembagian kelas warna seperti pada tabel I untuk mendapatkan titik-titik awal *cluster*.

No.	Kelas Warna	R	G	B
1.	Latar belakang	0	0	0
2.	Biru	0	0	1
3.	Hijau	0	1	0
4.	Cyan	0	1	1
5.	Merah	1	0	0
6.	Magenta	1	0	1
7.	Kuning	1	1	0
8.	Putih	1	1	1

Tabel I: Tabel Kelas Warna Citra Jeruk

1) *Klasifikasi Awal dengan Kelas Warna*: Penentuan kelas warna awal untuk tiap nilai *pixel* dilakukan dengan *thresholding*. Sebuah nilai *pixel* dianggap memiliki salah satu dari ketiga komponen warna primer apabila nilai intensitas pada komponen tersebut melebihi nilai *threshold*.

$$isColor(p, c) = \begin{cases} 1, & p_c \geq t_c \\ 0, & p_c < t_c \end{cases} \quad (4)$$

$p_c$  merupakan nilai *pixel* untuk komponen warna  $c$ ,  $t_c$  merupakan nilai *threshold* Otsu untuk komponen warna tersebut.

Tiap komponen warna diberi bobot sedemikian rupa sehingga tiap kelas warna disimbolkan sebagai bilangan integer unik, dengan angka terkecil diberikan kepada kelas yang memiliki intensitas paling rendah dan angka terbesar untuk kelas dengan intensitas tertinggi.

$$pixelClass(p) = 4isColor(p, R) + 2isColor(p, G) + isColor(p, B) + 1 \quad (5)$$

Selanjutnya tiap *pixel* pada citra disubstitusikan dengan bilangan integer kelasnya sehingga untuk tiap citra yang diolah didapatkan matriks integer seukuran citra asli yang memuat data kelas-kelas *pixel* pada citra.

2) *Pembuatan Kelas Baru*: Kelas-kelas yang muncul di klasifikasi awal perlu disusun ulang untuk menggabungkan kelas-kelas yang berdekatan nilainya dan menghapus kelas yang tidak muncul pada citra. Penggabungan kelas dilakukan berdasarkan antara pusat-pusat kelas yang dihitung dari rata-rata intensitas kelas pada tiap komponen warna.

Rata-rata intensitas kelas

$$\mu_{R,G,B}(k) = \frac{\sum_k p_{R,G,B}}{n_k} \quad (6)$$

Jarak antarkelas

$$d_{a,b} = \sqrt{\frac{\sum_{R,G,B} (\mu(a) - \mu(b))^2}{3}} \quad (7)$$

Dari jarak antarkelas didapatkan tetangga untuk masing-masing kelas, yaitu kelas dengan jarak terdekat. Dua kelas yang saling bertetangga digabungkan dan rata-ratanya dihitung ulang. Tabel II dan gambar 2 menunjukkan ilustrasi ketetanggaan antarkelas dan kelas-kelas baru hasil penyusunan ulang.

3) *Klasifikasi Ulang Pixel ke Kelas Baru*: *Pixel* diklasifikasikan ulang ke kelas-kelas baru yang diperoleh dengan metode *nearest neighbor*. Perhitungan jarak yang digunakan adalah *sum of squared difference* untuk tiap komponen warna.

Kelas	K1	K2	K3	K4	K5	K6	K7	K8
K1	-	49.07	50.35	57.28	51.15	59.82	94.55	110.60
K2	49.07	-	19.95	8.51	14.51	11.02	55.70	62.96
K3	50.35	19.95	-	20.74	7.64	25.36	44.70	62.65
K4	57.28	8.51	20.74	-	15.26	5.59	49.35	54.68
K5	51.15	14.51	7.64	15.26	-	19.00	45.47	59.98
K6	59.82	11.02	25.36	5.59	19.00	-	50.90	53.41
K7	94.55	55.70	44.70	49.35	45.47	50.90	-	34.19
K8	110.60	62.96	62.65	54.68	59.98	53.41	34.19	-

Tabel II: Contoh matriks jarak antarkelas

	Kelas terdekat	Kelas baru
K1	K2	K1
K2	K4	K2
K3	K5	K3=K3+K5
K4	K6	K4=K4+K6
K5	K3	K5=K7+K8
K6	K4	
K7	K8	
K8	K7	

Gambar 2: Penggabungan Kelas

$$ssd(p, k) = \sqrt{\frac{\sum_{r,g,b} (\mu(k) - p)^2}{3}} \quad (8)$$

#### F. Deteksi dan Pengukuran Prosentase Cacat Kulit Jeruk

*Pixel* yang diklasifikasikan dalam kelas dengan rata-rata intensitas tertinggi merupakan daerah normal pada kulit jeruk. Untuk menandainya dilakukan operasi *thresholding* pada matriks kelas *pixel* dengan nilai kelas tertinggi sebagai nilai *threshold*.

Penandaan daerah cacat memerlukan penentuan seluruh daerah jeruk yang bisa diproses dengan metode ini. Daerah tepi dengan kelas yang intensitasnya lebih rendah tidak dapat diproses karena perbedaan perbandingan nilai intensitas antara daerah normal dan daerah cacat. Bagian tepi tersebut tidak dimasukkan ke dalam perhitungan.

Untuk mendapatkan seluruh daerah yang tercakup pemrosesan maka dilakukan operasi morfologis *hole filling* pada *mask* daerah normal.

$$p_{cacat} = \frac{L_{cacat}}{L_{normal} + L_{cacat}} \quad (9)$$

$$L_{normal} \approx Luas(M) \quad (10)$$

$$L_{cacat} \approx Luas(M' - M) \quad (11)$$

$H$  adalah kumpulan *pixel* dengan kelas yang paling tinggi intensitasnya,  $H'$  adalah hasil pengisian lubang. Cacat dideteksi dari selisih antara  $H$  dan  $H'$ .

#### G. Akurasi Deteksi

Akurasi deteksi cacat akan diukur dari parameter presisi, *recall* dan akurasi statistik jumlah *pixel* daerah cacat dan daerah normal. Hal-hal ini terkait dengan keberhasilan sistem dalam mengambil informasi yang dibutuhkan pengguna. Pada penelitian ini informasi yang dicari adalah *pixel* dari daerah cacat. Akurasi sistem biasanya dihitung dengan *confusion matrix* yang digambarkan pada gambar 3

Dalam pengukuran akurasi perlu dipertimbangkan batasan daerah yang diproses saat pendeteksian cacat karena proses

pendeteksian tidak dilakukan untuk daerah tepi yang intensitasnya lebih rendah. Karena itu perlu ditambahkan lagi ukuran prosentase deteksi yang dirumuskan sebagai perbandingan antara luas daerah yang dapat diproses dari tiap citra jeruk dengan luas keseluruhan objek jeruk yang tampak. Prosentase deteksi ini tidak akan mencapai 100% karena bagian tepi objek jeruk menerima lebih sedikit cahaya intensitas warnanya lebih rendah dan tidak terdeteksi sebagai kelas tertinggi (daerah normal).

$$deteksi = \frac{Luas(H')}{\sum_{kelas>1}^n Luas(M_{kelas})} \quad (12)$$

$H'$  merupakan hasil pengisian lubang yang menunjukkan bagian yang dapat diproses,  $M_{kelas}$  merupakan mask hasil klasifikasi akhir jeruk.

		Nilai sebenarnya	
		TRUE	FALSE
Nilai prediksi	TRUE	TP (True Positive) Correct result	FP (False Positive) Unexpected result
	FALSE	FN (False Negative) Missing result	TN (True Negative) Correct absence of result

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Gambar 3: Confusion matrix

Jika diketahui citra biner  $D_{proses}$  sebagai hasil deteksi cacat dari pengolahan data dan  $D_{proses}^c$  komplementnya, citra biner  $A$  sebagai hasil pengisian lubang, serta citra biner  $D_{manual}$  sebagai penandaan cacat yang dibuat manual dan komplement  $D_{manual}^c$ , persamaan-persamaan berikut dapat digunakan untuk mengambil keempat ukuran akurasi di atas. Perlu dilakukan operasi irisan dengan daerah batas deteksi  $A$  untuk mendapatkan hasil yang valid.

- 1) *True positive rate*: jumlah *pixel* daerah cacat yang berhasil dideteksi oleh algoritma

$$TP = n(D_{proses} \cap D_{manual} \cap A)$$

- 2) *False positive rate*: jumlah *pixel* daerah normal yang ditandai sebagai cacat.

$$FP = n(D_{proses} \cap D_{manual}^c \cap A)$$

- 3) *True negative rate*: jumlah *pixel* daerah normal yang tidak ditandai sebagai cacat,

$$TN = n(D_{proses}^c \cap D_{manual}^c \cap A)$$

- 4) *False negative rate*: jumlah *pixel* daerah cacat yang tidak terdeteksi oleh algoritma.

$$FN = n(D_{proses}^c \cap D_{manual} \cap A)$$

### III. METODE PENELITIAN

#### A. Objek Penelitian

Objek penelitian yang penulis gunakan dalam penelitian ini adalah jeruk keprok (*Citrus sinensis*). Fokus pengamatan adalah cacat pada tekstur kulit jeruk.

#### B. Jenis dan Sumber Data

Pada penelitian ini, penulis menggunakan beberapa jenis data.

- 1) Data primer, berupa foto-foto jeruk keprok dari berbagai kelas kualitas.
- 2) Data Sekunder, adalah data primer yang sudah diolah lebih lanjut dan disajikan dengan baik oleh pihak pengumpul data primer atau pihak lainnya. Data yang penulis gunakan diolah dari foto-foto jeruk yang telah dipersiapkan

Selain itu, penulis juga mengumpulkan data melalui studi literatur, yaitu dengan mempelajari jurnal dan hasil penelitian yang sudah ada untuk mendapatkan pemahaman tentang analisis tekstur.

#### C. Analisis Kebutuhan Perangkat

Tujuan dari proses analisa kebutuhan aplikasi adalah untuk mengetahui sifat dari kebutuhan sistem sehingga mempermudah dalam perancangan. Tujuan lain dari analisa ini adalah untuk mendokumentasikan sifat program tersebut. Proses analisis meliputi analisis kebutuhan perangkat lunak dan perangkat keras, termasuk analisis terhadap kebutuhan sistem.

##### 1) Kebutuhan Perangkat Lunak:

- 1) Octave dengan *package image* untuk melakukan pemrosesan citra dan pengolahan data secara numerik. Matlab dapat digunakan sebagai pengganti Octave.

##### 2) Kebutuhan Perangkat Keras:

- 1) PC yang dapat menjalankan perangkat lunak Octave
- 2) Kamera minimum 2MP
- 3) Lampu LED 5W
- 4) Backdrop kain hitam

3) *Tahapan Penelitian*: Langkah-langkah penelitian yang harus dilakukan yaitu:

- 1) mengambil foto-foto jeruk dari berbagai kualitas, Untuk akurasi data maka pengambilan foto perlu dilakukan pada kondisi yang terkontrol.
- 2) pengolahan awal foto
- 3) perhitungan *threshold* komponen warna
- 4) klasifikasi *pixel* awal
- 5) perhitungan rata-rata intensitas setiap kelas
- 6) pembuatan kelas baru
- 7) klasifikasi *pixel* akhir
- 8) pembuatan *mask* untuk menandai objek jeruk dan daerah-daerah cacat pada jeruk
- 9) perhitungan luas daerah cacat pada permukaan kulit jeruk
- 10) perhitungan akurasi hasil deteksi

Pengambilan foto dilakukan dengan perlengkapan-perengkapan berikut

- 1) Kamera DSLR Nikon D3100
- 2) Sumber cahaya lampu LED Philips 5W dan *stand*
- 3) Kain hitam

Kamera DSLR diset sebagai berikut:

- 1) sensitivitas ISO 200
- 2) mode pencahayaan *aperture priority*
- 3) format citra JPEG dengan resolusi 3406x2304 *pixel*

Kamera, sumber cahaya dan objek disusun sebagai Gambar

4. Jarak antara kamera dan objek adalah 50cm.



Gambar 4: Penataan kamera, sumber cahaya dan objek

Pengambilan foto diulang dua kali dengan objek diputar 180° untuk mengambil gambar semua sisi jeruk.

Foto di-*crop* dengan ukuran 2304x2304 *pixel* untuk membuang bagian yang tidak diperlukan. Ukuran dan resolusi foto diperkecil menjadi 72dpi dan ukuran 800x800 *pixel* untuk mengurangi beban pemrosesan sistem. Karena lingkungan pengambilan foto yang kurang ideal maka diperlukan penyuntingan manual gambar untuk menghilangkan latar belakang dan objek selain jeruk. Hasil penyuntingan disimpan dalam format PNG.



Gambar 5: Foto jeruk yang digunakan

#### IV. HASIL DAN PEMBAHASAN

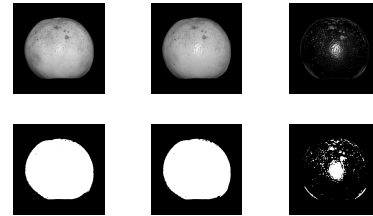
Data berupa foto jeruk yang telah diambil sesuai dengan langkah-langkah yang dijelaskan pada subbab III-C3. Pada penelitian ini dipersiapkan 12 foto dari 6 jeruk. Setiap jeruk difoto dua kali untuk mendapatkan foto bagian depan dan belakang.

#### A. Pengolahan Data

1) *Thresholding*: *Thresholding* dengan metode Otsu diterapkan pada tiap komponen warna. Nilai *threshold* untuk data contoh ditampilkan di tabel III. Hasil operasi *thresholding* untuk tiap komponen warna ditampilkan di gambar 6.

Komponen warna	Nilai <i>Threshold</i>
R	82
G	79
B	44

Tabel III: Nilai *threshold* metode Otsu



Gambar 6: Hasil operasi *thresholding* pada komponen warna citra R, G, dan B

Operasi *threshold* menghasilkan tiga citra biner untuk masing-masing komponen warna. Klasifikasi dapat dilakukan ke *pixel* data citra dengan menerapkan langsung rumus 5 pada ketiga citra biner sehingga dihasilkan matriks integer yang menyimpan nilai kelas untuk data *pixel* citra.

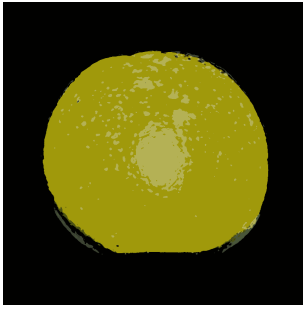
Hasil operasi *thresholding* menunjukkan bahwa komponen warna biru pada citra jeruk mempunyai intensitas yang rendah, sebaliknya komponen warna merah dan hijau mempunyai intensitas lebih tinggi dan tersebar merata pada hampir seluruh permukaan jeruk yang tampak. Hal ini dapat dipahami sebagai akibat dari karakteristik warna jeruk yang didominasi oleh warna hijau dan kuning. Komponen warna biru justru mempunyai intensitas lebih tinggi pada daerah-daerah yang secara visual tampak sebagai bagian cacat dari kulit jeruk.

2) *Klasifikasi Pixel Awal*: Klasifikasi awal dilakukan dengan *weighted sum* dari ketiga citra biner hasil *thresholding* untuk ketiga komponen warna. *Pixel* citra jeruk diklasifikasikan berdasarkan keberadaan komponen warna primer. Hasil klasifikasi adalah matriks integer dengan rentang nilai [1..8]. Dari matriks ini dihitung banyaknya *pixel* yang dimasukkan ke masing-masing kelas sehingga dapat diketahui rata-rata intensitas tiga komponen warna dari masing-masing kelas.

Hasil klasifikasi awal membagi jeruk menjadi 8 kelas berdasarkan segmentasi komponen-komponen warna primer dengan metode Otsu. Pada contoh ditunjukkan visualisasi hasil klasifikasi pada gambar 7, nilai-nilai intensitas asli *pixel* digantikan dengan rata-rata intensitas kelas masing-masing *pixel* tersebut. Tabel rata-rata intensitas IV menunjukkan bahwa seluruh kelas warna muncul pada citra yang diuji.

3) *Pembuatan Kelas Baru*: Kelas-kelas *pixel* selanjutnya disusun ulang berdasarkan jarak antarkelas yang dihitung dari *mean squared distance* rata-rata intensitas. Kelas yang saling bertetangga digabungkan kemudian rata-rata intensitas baru dihitung untuk tiap kelas. Tabel V menunjukkan hasil perhitungan jarak antarkelas untuk data contoh, tabel VI



Gambar 7: Visualisasi hasil klasifikasi *pixel* awal

Kelas	Rata-rata intensitas		
	R	G	B
1	1.67	0.72	0.75
2	61.11	68.35	51.13
3	76.11	82.57	21.15
4	73.85	87.14	57.80
5	84.01	76.80	25.06
6	82.90	77.18	49.90
7	160.50	154.17	11.82
8	180.61	177.75	85.59

Tabel IV: Nilai rata-rata intensitas tiap kelas

menunjukkan kelas baru yang dibuat. Tabel VII memuat rata-rata intensitas kelas baru.

Kelas	K1	K2	K3	K4	K5	K6	K7	K8
K1	-	59.20	64.79	72.49	65.85	70.03	127.27	153.05
K2	59.20	-	21.18	13.66	20.62	13.60	79.14	95.68
K3	64.79	21.18	-	21.39	5.80	17.26	63.84	89.52
K4	72.49	13.66	21.39	-	20.67	9.01	68.59	82.46
K5	65.85	20.62	5.80	20.67	-	14.36	63.28	88.00
K6	70.03	13.60	17.26	9.01	14.36	-	66.83	83.59
K7	127.27	79.14	63.84	68.59	63.28	66.83	-	46.39
K8	153.05	95.68	89.52	82.46	88.00	83.59	46.39	-

Tabel V: Jarak antarkelas

Kelas	Kelas Terdekat	Kelas Baru
K1	K2	K1
K2	K6	K2
K3	K5	K3 = K3 + K5
K4	K6	K4 = K4 + K6
K5	K3	-
K6	K4	-
K7	K8	K5 = K7 + K8
K8	K7	-

Tabel VI: Penggabungan kelas

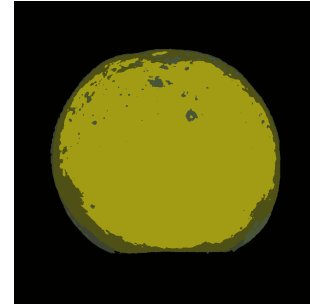
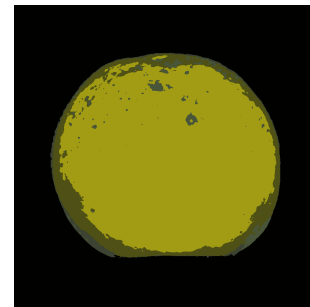
4) *Reklasifikasi Pixel*: *Pixel* pada citra asli diklasifikasi ulang berdasarkan jarak terdekat dari nilai rata-rata intensitas dengan perhitungan *sum squared distance*. Gambar 8 menunjukkan hasil klasifikasi ulang, nilai intensitas asli tiap *pixel* diganti dengan rata-rata intensitas kelas untuk memvisualisasikan matriks kelas *pixel*.

5) *Operasi Morfologi*: Dari matriks data klasifikasi akhir dilakukan *thresholding* untuk mengambil titik-titik *pixel* dengan kelas yang mempunyai rata-rata intensitas tertinggi sebagai sebuah citra biner yang menunjukkan bagian teksur normal. Operasi morfologi penutupan lubang dilakukan atas citra biner ini lalu diambil selisihnya untuk mendeteksi bagian-bagian cacat.

Gambar 9 menunjukkan bahwa bagian tepi masuk ke dalam kelas dengan intensitas yang lebih rendah. Hal ini disebabkan

Kelas	Rata-rata intensitas		
	R	G	B
1	1.67	1.72	0.75
2	61.11	68.35	51.13
3	78.62	81.10	22.14
4	74.50	86.43	57.24
5	163.00	157.10	21.04

Tabel VII: Rata-rata intensitas kelas baru

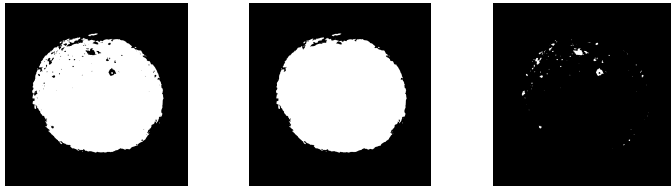
Gambar 8: Hasil klasifikasi ulang *pixel*Gambar 9: Operasi *thresholding* pada matriks klasifikasi akhir dan citra biner yang dihasilkan

oleh penyinaran yang diterima bagian tepi tidak sebanyak bagian tengah karena perbedaan sudut datang sinar yang mengenai bagian tersebut. Sementara itu di bagian tengah dapat dilihat bahwa variasi warna yang cukup mencolok pada daerah normal tidak mempengaruhi hasil klasifikasi sehingga perbedaan warna di daerah normal tidak menimbulkan kesalahan deteksi daerah cacat, sebagaimana ditunjukkan oleh gambar 10.

Tabel VIII adalah data-data jumlah *pixel* yang didapatkan pada gambar di atas.



Gambar 10: Pengaruh variasi warna pada deteksi



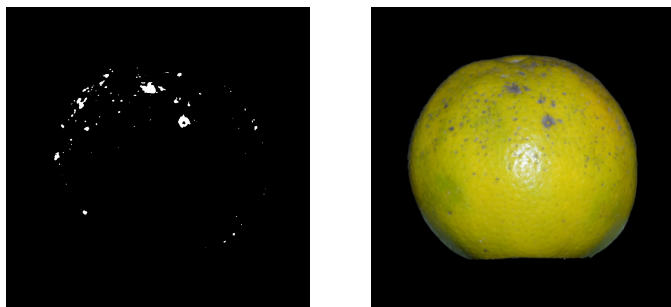
Gambar 11: Mask tekstur normal, hasil penutupan lubang dan cacat yang dideteksi

Data	Jumlah <i>pixel</i>
Daerah normal	221386
Daerah cacat	3615

Tabel VIII: Data perhitungan jumlah *pixel*

**B. Perhitungan Luas Cacat dan Akurasi**

Sebagai acuan akurasi deteksi, digunakan citra biner cacat tekstur kulit yang digambar secara manual dengan menggunakan perangkat *fuzzy select* dan *color select* pada aplikasi pengolahan citra Gimp. Luas daerah yang dapat diproses dibandingkan dengan keseluruhan luas objek jeruk dalam citra untuk mendapatkan prosentase luas deteksi.

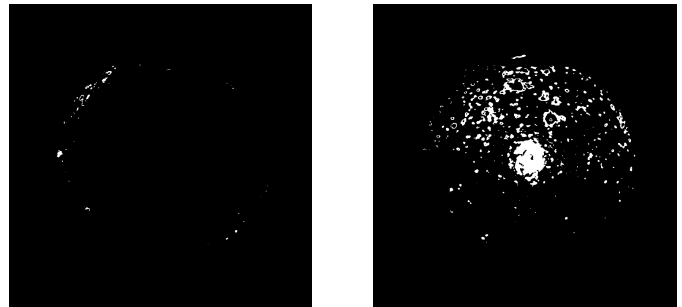
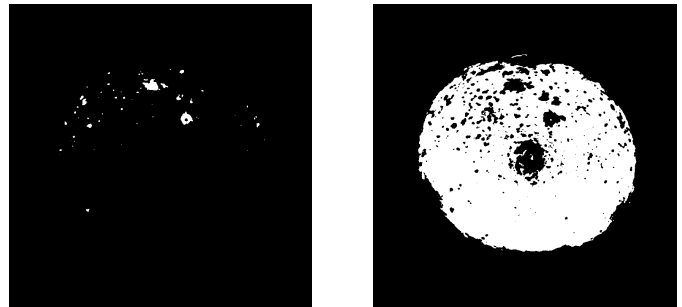


Gambar 12: Hasil deteksi dengan pemrosesan dan hasil penandaan manual

Data	Jumlah <i>pixel</i>
<i>TP</i>	2731
<i>FP</i>	884
<i>TN</i>	200468
<i>FN</i>	20918
Area deteksi	225001
Total area objek	271992

Tabel IX: Tabel perhitungan akurasi deteksi

Dari data Tabel IX didapatkan ukuran presisi, akurasi dan *recall* untuk pemrosesan data sampel.



Gambar 13: Mask pengujian hasil deteksi: *true positive*, *true negative*, *false positive*, *false negative*



Gambar 14: Perbandingan bagian jeruk yang dapat dideteksi dengan keseluruhan permukaan jeruk pada citra

$$presisi = \frac{2731}{2731 + 884} = 0.76$$

$$akurasi = \frac{2731 + 200468}{2731 + 200468 + 884 + 20918} = 0.90$$

$$recall = \frac{2731}{2731 + 20918} = 0.12$$

Prosentase permukaan yang dapat dideteksi.

$$deteksi = \frac{221386}{271992} * 100\% = 82.72\%$$

Prosentase hasil deteksi dan akurasi dari semua data eksperimen ditunjukkan pada tabel X.

Perbandingan akurasi dengan hasil penandaan manual ditunjukkan pada tabel XI.

**C. Analisis dan Interpretasi Hasil**

Berikut ini adalah beberapa hal yang diamati dari hasil pemrosesan data.

Data	<i>n pixel normal</i>	<i>n pixel hole fill</i>	<i>n pixel cacat</i>	% cacat
1a	241788	243449	1661	0.682
1b	37995	38716	721	1.862
2a	221386	225001	3615	1.607
2b	229222	230548	1326	0.575
3a	175995	182726	6731	3.684
3b	121087	130677	9590	7.339
4a	163142	169382	6240	3.684
4b	173986	176931	2945	1.664
5a	181115	184190	3075	1.669
5b	171486	173386	1900	1.096
6a	163276	166109	2833	1.706
6b	184366	186877	2511	1.344

Tabel X: Hasil deteksi *pixel* daerah cacat

Data	% Presisi	% Akurasi	% Recall	% deteksi
1a	30.283	78.233	0.776	80.975
1b	0.555	75.278	0.006	13.401
2a	75.546	85.957	6.820	82.723
2b	77.451	87.186	2.888	84.825
3a	9.330	60.080	0.654	71.901
3b	64.140	36.804	3.787	51.713
4a	94.936	56.232	5.318	70.087
4b	80.747	62.370	2.556	72.993
5a	67.675	55.040	1.750	70.289
5b	27.526	51.000	0.420	67.719
6a	89.728	62.025	2.539	64.453
6b	72.322	68.723	2.240	73.115

Tabel XI: Akurasi dan luas deteksi hasil pemrosesan

- 1) Jeruk kuning menunjukkan variasi intensitas yang relatif rendah dengan kontras antara daerah normal dan daerah cacat tinggi. Cakupan daerah yang terproses luas karena intensitas tinggi yang merata meliputi sebagian besar permukaan jeruk. Kontras yang tinggi antara daerah cacat dan daerah normal menyebabkan akurasi deteksi relatif tinggi di antara data lainnya.
- 2) Jeruk hijau mempunyai variasi warna yang lebih banyak daripada jeruk kuning. Daerah intensitas tinggi yang dapat dideteksi sedikit lebih sempit daripada jeruk kuning, akurasinya sebanding dengan data jeruk kuning. Dari contoh data terlihat bahwa cacat di bagian tengah tidak terdeteksi karena adanya kilap dari pantulan cahaya.
- 3) Data citra jeruk 1b memperlihatkan adanya kilap di tengah objek. Perbedaan intensitas kilap ini cukup tinggi dibandingkan dengan daerah lain sehingga membentuk kelas tersendiri yang sebagian besar terdiri dari *pixel-pixel* di daerah kilap. Akibatnya, bagian selain kilap tidak masuk area deteksi. Hanya 13% dari keseluruhan luas objek jeruk di dalam citra yang terproses, sedangkan pemrosesan tidak dapat bekerja dengan akurat untuk bagian yang berkilap.

Data menunjukkan bahwa metode deteksi menghasilkan hasil yang bervariasi, dengan akurasi tinggi, presisi yang cukup baik, namun *recall* sangat rendah. Tingkat *recall* yang rendah ini disebabkan karena hanya sebagian dari permukaan jeruk yang dapat dideteksi, dan terutama karena citra biner yang digunakan untuk referensi kurang akurat.

Metode ini selain tidak dirancang untuk mendeteksi cacat pada bagian-bagian gelap pada tepi jeruk, juga tidak dapat mendeteksi cacat pada bagian tengah jeruk yang berkilap

karena sorotan sumber cahaya. Perlu dilakukan penyesuaian pencahayaan untuk mengurangi atau menghindari munculnya kilap di tengah objek.

## V. KESIMPULAN DAN SARAN

### A. Kesimpulan

Metode klasifikasi *pixel* dapat mendeteksi letak cacat pada kulit jeruk dengan baik. Kelemahan metode ini adalah hasil deteksi tidak mencakup seluruh permukaan jeruk yang ditampilkan pada citra jeruk. Selain itu metode ini juga tidak dapat mendeteksi cacat pada daerah yang berkilap. Diperlukan pencahayaan yang baik untuk mendapatkan hasil deteksi yang optimum.

### B. Saran

Setelah mengadakan penelitian ini, penulis menyampaikan beberapa saran untuk penelitian selanjutnya:

- 1) Penelitian lebih lanjut mengenai pengaruh pemrosesan awal citra dengan penyesuaian histogram untuk meningkatkan efektivitas deteksi dengan metode klasifikasi *pixel* ini.
- 2) Pengolahan bagian tepi jeruk yang tidak dimasukkan ke dalam proses deteksi cacat dengan *clustering* atau metode klasifikasi lainnya untuk mendeteksi daerah cacat tambahan pada bagian tepi.

## DAFTAR PUSTAKA

- [1] Basuki, A. Palandi, F. J. Fatchurrochman (2005). *Pengolahan Citra Digital Menggunakan Visual Basic*. Graha Ilmu: Yogyakarta.
- [2] Brosnan, Tadhg, Da-Wen Sun (2004). "Improving quality inspection of food products by computer vision review," *Journal of Food Engineering*, 61, 3-16.
- [3] Dila Deswari, Hendrick, Derisma. (2013). *Identifikasi Kematangan Buah Tomat Menggunakan Metode Backpropagation*. Universitas Andalas: Padang.
- [4] Gonzales, Rafael C., Richard E. Woods (2002). *Digital Image Processing*. Prentice Hall: New Jersey.
- [5] Knuth, Donald. (1973). *The Art Of Computer Programming Volume 3*. Addison Wesley: Boston.
- [6] Kulkarni, Arum D. (2001). *Computer Vision and Fuzzy-Neural Systems*. Prentice Hall: New Jersey.
- [7] Liu, Ming Hui. (2009). *Navel orange blemish identification for quality grading system*. Magister Thesis. Massey University.
- [8] Sezgin, Mehmet, Bulen Sankur (2004). "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging* 13(1), 146165.
- [9] Linda G. Shapiro, Stockman, George C (2002). *Computer Vision*. Prentice Hall: New Jersey.
- [10] Schalkoff, Robert J (1989). *Digital Image Processing and Computer Vision*. John Wiley & Sons: New York.
- [11] Standard Nasional Indonesia (2009). *SNI 3165:2009 Jeruk Keprok*. BSN: Jakarta.