

SISTEM OTOMASI DALAM PENYORTIRAN TOMAT DENGAN IMAGE PROCESSING MENGGUNAKAN METODE DETEKSI RGB

AUTOMATED SYSTEM IN TOMATO SORTING WITH IMAGE PROCESSING USING RGB DETECTION METHOD

Ihsan Nugraha Putra Mukhti¹, Drs Suwandi, M.Si.², Hertiana Bethaningtyas, S.T., M.T.³

^{1,2,3}Prodi S1 Teknik Fisika, Fakultas Teknik Elektro, Universitas Telkom
ihsannugraha.pm@gmail.com¹, suwandi@telkomuniversity.ac.id², hertiana@telkomuniversity.ac.id³

ABSTRAK

Tomat merupakan salah satu buah hortikultura yang mengalami perubahan warna menuju proses kematangannya. Perubahan warna pada tomat dapat diamati secara langsung oleh mata. Pada umumnya warna tomat muda berwarna hijau dan perlahan menjadi merah. Pakar hortikultura telah mengidentifikasi warna tomat yang matang secara tampilan visual. Image processing menjadi solusi mendapatkan nilai warna pada tomat yang matang. Nilai didapat dari proses mendeteksi nilai RGB (Red, Green and Blue) pada warna tomat. Dimulai dari mendeteksi nilai warna setelah nilai diolah menjadi keputusan matang atau tidaknya tomat yang dilihat. Faktor cahaya dan tingkat sensitivitas kamera pada webcam berpengaruh dalam pengambilan data pada tomat. Tekstur warna pada tomat dapat diidentifikasi ekstraksi cirinya dengan menggunakan metode deteksi warna. Informasi yang terdapat pada sebuah citra tomat berupa nilai - nilai ciri statistik. Berdasarkan nilai ekstraksi ciri, citra yang diakuisisi dari kamera dapat diklasifikasi dengan *euclidean distance*. Berdasarkan pengujian yang telah dilakukan, didapatkan akurasi tertinggi mencapai 88% pada saat pengambilan 20 cm menggunakan preprocessing dengan kondisi cahaya terang (70-110lux). Jika jarak tidak fokus dan menjauhi objek akurasi akan menurun. Fokus jarak objek terhadap webcam maksimal dengan jarak 20 cm dengan akurasi 88%. Perbedaan parameter resolusi kamera juga memberikan tingkat akurasi yang berbeda yaitu maksimum pada 80% dengan resolusi 5MP dan menurun jika resolusi lebih kecil lagi. Waktu komputasi rata – rata untuk PC dengan RAM 2GB adalah 4500ms.

Kata kunci: *Image Processing*, Tomat, RGB, Otomasi, Hortikultura, Matlab

ABSTRACT

Tomato is one of horticulture fruit with colour changed until its ripe. Tomato colour changing could be observe in direct. Generally unripe tomato have green colour and will be change to merah. Horticulture's expert have identification unripe tomato with visual observe. Image processing can be solution to get value of ripe tomato colour. This Value was got from detection process of RGB value (Red, Green and Blue) on tomato colour. Start from colour detection the value will get process be ripe or Un-ripe Tomato decision. This process was got from find mean value on red ,green and blue colour on tomato .Tomato colour could be identified with feature extraction using colour detection method. Information from tomato image was statistic value. Based on value of extraction, acquisitions image will be grouping with Sum of Square Difference. Based on examination, highest accuracy is 88% in taking 20cm with preprocessing and bright light. If distance failure to focus and avoid from object the accuracy will go down. Distance of object focus on webcam was optimal with 20cm with accuracy 88%. Different resolution of camera also affect accuracy till 80% in 5 MP and down if used low resolution. Average computation time for PC with RAM 2GB is 4500ms.

Keywords: *Image Processing*, Tomato, RGB, Automation, Hortikultura, Matlab

1. PENDAHULUAN

1.1. Latar Belakang

Tomat merupakan salah satu jenis buah hortikultura. Saat tumbuh dan berkembang tomat mempunyai ciri warna yang khas. Ketika mulai berbuah dari mentah sampai matang terjadi perubahan warna pada tomat. Pemasaran panen tomat yang baik siap distribusikan ke pasar tradisional atau supermarket. Beberapa landasan penelitian yang berhubungan dengan teknik penyortiran dan sistem *image processing* yang telah dilakukan sebelumnya antara lain jurnal tentang *Klasifikasi Tingkat Kematangan Varietas Tomat Merah dengan Metode Perbandingan Kadar Warna* [4] yang menggunakan metode perbandingan kadar warna dalam varietas tomat merah telah dapat digunakan dengan baik untuk mengklasifikasikan tingkat kematangan suatu tomat. Kemudian *Otomatisasi Pemisah Buah Tomat*

Berdasarkan Ukuran dan Warna Menggunakan Webcam Sebagai Sensor [5] serta Sistem Deteksi Jenis Buah Jeruk Berbasis Pengolahan Citra [6]. Referensi sebelumnya mempunyai kelemahan saat pengambilan citra, keterbatasan jarak antara tomat dengan sensor kamera, susahnya mengatur intensitas cahaya dan data yang diambil mempunyai akurasi data 80-95

1.2. Batasan Masalah

Batasan Masalah yang dalam tugas akhir adalah:

1. Pemrosesan *Image Processing* hanya menggunakan simulasi dengan software MATLAB 2013a.
2. Pengambilan data gambar dengan *webcam* 5MP (resolusi 640 x 480) mengikuti referensi data pakar tomat.
3. Metode yang digunakan adalah *Deteksi RGB*.
4. Intensitas Cahaya ditentukan sekitar 70lx – 110lx dan jarak pengambilan 20 cm.
5. Menggunakan komputer Dell Optiplex 7010 series, Intel® Core™ i7 3770HQ Processor, RAM 4GB, HDD 250 GB dan menggunakan sistem operasi Windows 7 64-bit.
6. Pengujian citra dilakukan dengan 4 sisi tomat.
7. Jenis tomat yang diuji adalah tomat sayur

1.3. Rumusan Masalah

Rumusan masalah penelitian adalah bagaimana merancang sistem otomatis penyortiran buah tomat dengan proses *Image Processing* untuk mengetahui matang, setengah matang dan mentahnya buah tomat dilihat dari warna dengan metode Deteksi RGB.

1.4. Tujuan

Tujuan dari pengerjaan tugas akhir untuk memanfaatkan *Image Processing* dalam mengidentifikasi keadaan suatu objek (tomat) hanya dari warna citra.

2. LANDASAN TEORI

2.1. Tomat

Tomat (*Lycopersicon esculentum*) adalah tumbuhan keluarga *Solanaceae*, biasa dikonsumsi dalam kehidupan sehari-hari. Tomat dapat dikonsumsi langsung sebagai buah maupun diolah sebagai sayuran. Untuk membedakan jenis tomat buah atau sayur dapat dilihat dari bentuk dan ketebalan kulitnya. Tomat buah berbentuk agak lonjong dan memiliki kulit agak tebal sedangkan tomat sayur memiliki bentuk bulat dan berkulit tipis. Jenis tomat yang digunakan pada penelitian adalah tomat sayur[1].

2.2. Pengolahan Citra Digital

Pengolahan citra digital merupakan sebuah ilmu yang mempelajari hal - hal yang berkaitan dengan perbaikan kualitas citra. Dengan kata lain pengolahan citra adalah kegiatan memperbaiki kualitas citra agar mudah diterjemahkan oleh manusia atau mesin (komputer) [2].

2.2.1 Citra Grayscale

Citra grayscale adalah warna-warna *pixel* yang berada dalam rentang gradasi warna hitam dan putih. Citra grayscale digunakan untuk menyetarakan intensitas warna. Pada intensitas warna merah, hijau dan biru, citra grayscale mempunyai nilai *pixel* yang sama [6]

2.2.2 Citra Biner

Dalam citra biner setiap *pixel* hanya mempunyai dua kemungkinan nilai yaitu 0 dan 1. Umumnya, angka nol mewakili warna hitam dan angka satu mewakili warna putih. Citra biner diperoleh melalui proses pemisahan *pixel-pixel* berdasarkan skala keabuan yang dimilikinya. *Pixel* yang memiliki skala keabuan lebih kecil dari nilai batas yang ditentukan diberikan nilai 0, sementara *pixel* yang memiliki skala keabuan yang lebih besar dari batas diubah menjadi bernilai 1. Jadi untuk gambar yang biasanya berwarna hitam putih, nilainya 0 dan 1 [6].

2.2.3 Citra Warna (RGB)

Citra berwarna merupakan gabungan dari beberapa lapis citra kanal warna yang bertumpuk. Masing-masing lapisan merepresentasikan nilai intensitas warna tertentu terhadap warna gelap sehingga pada citra berwarna setiap *pixel* mempunyai informasi warna tertentu yang merupakan gabungan warna-warna dari citra kanal warna. Misalnya

RGB (Red Green Blue) mempunyai warna dasar merah, hijau dan biru digabungkan dalam membentuk suatu susunan warna yang luas.

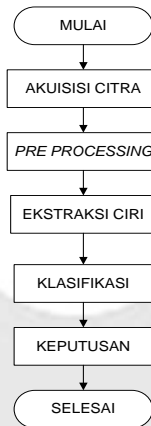
2.3 Metode Perbandingan Kadar Warna (Deteksi RGB)

Pencarian ciri mean dengan menjumlahkan semua nilai pixel terhadap ukuran dan dibagi dengan jumlah pixel yang ada. Setiap pixel akan diambil mean red, green, blue dan standar deviasi dari red, green, blue serta varian dari red, green dan blue sehingga terdapat 9 ciri yang akan dihitung. Setelah menemukan nilai dari 9 ciri yang dicari digunakan 50 tomat yang telah ditandai oleh pakar tomat tingkat kematangan tomat sehingga dijadikan sebagai *landasan* untuk melakukan pengujian. Selanjutnya data pakar ditentukan 9 ciri dari 50 tomat dan dijadikan sebagai *database*. Pencarian jarak ditentukan oleh sistem. Pada saat pengujian jarak antara ciri pengujian dan setiap ciri yang digunakan untuk *database*. Selanjutnya jarak minimum antar ciri ditemukan dan dijadikan nilai akhir. *Euclidian Distance* adalah metode perhitungan yang dapat digunakan untuk memperhitungkan kesamaan dua vektor. Euclidean distance menghitung akar kuadrat perbedaan 2 vektor (root of square differences). Penghitungan kemiripan warna *pixel* dapat dilakukan dengan menggunakan persamaan 2.3.

$$d_{i,j} = \sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.1)$$

3 PERANCANGAN SISTEM

3.1. Perancangan Sistem



Gambar 3. 1 Gambaran Umum Sistem

3.1.1 Penilaian Pakar

Pakar tomat adalah orang yang mengerti tentang tingkat kematangan buah tomat seperti orang pertanian dan perkebunan yang meneliti perkembangan buah tomat. Data citra tomat yang telah diambil diberikan kepada pakar dan dilakukan kuisioner tentang tingkat kematangan tomat pada citra. Dari hasil pengamatan pakar diperoleh nilai subjektif yang menjadi acuan untuk perbandingan akurasi pada sistem dan performansi sistem. Pakar mengamati dengan dua sisi citra tomat. Jumlah citra yang diamati sebanyak 50 buah tomat dengan koresponden berjumlah 15 orang. Setelah melakukan pengamatan, 17 tomat dinyatakan mentah, 17 tomat dinyatakan setengah matang dan 16 tomat matang. Sebelumnya tomat telah dilakukan pengambilan citra dan dicari nilai mean, standar deviasi dan variannya. Nilai tersebut akan menjadi ciri dan *database* untuk melakukan pengujian. Tomat-tomat seperti tabel 3.1 telah nyatakan kelasnya ditandai untuk setiap nilai pada cirinya.

3.2 Proses Akuisisi Citra

Akusisi citra membutuhkan kondisi webcam dan buah tomat yang berada dalam ruang lingkup pengambilan. Background dari buah tomat diberikan warna putih agar kontras dengan warna tomat. Perbedaan warna kontras antara background dan objek berguna dalam proses *thresholding*. Citra yang diambil secara *realtime* diperoleh format *JPG. Pengambilan citra dilakukan empat kali untuk satu tomat dan perputaran poros background dengan sudut 90 derajat.

Posisi buah yang tertangkap kamera relatif sama, artinya posisi buah selalu berada ditengah-tengah frame. Saat akuisisi dibutuhkan waktu pengambilan gambar yang akan dipengaruhi oleh *pixel* kamera dan sistem operasi dari komputer.

3.3 Pre-Processing

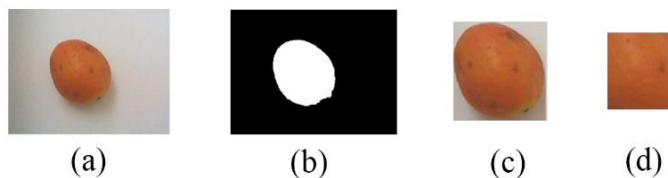
Pre-processing merupakan sebuah proses transisi yang dilakukan pada suatu citra digital sebelum dilakukan ekstraksi citra. Citra tomat akan diubah menjadi biner kemudian dipotong pada bagian tertentu. Tujuan *thresholding* dan *cropping* untuk memperoleh nilai ciri RGB yang dimiliki oleh setiap *pixel* pada citra tomat agar dengan toleransi nilai *pixel* tertentu *background* dan objek pada citra dapat dipisahkan. *Pre-processing* yang dirancang agar proses menuju ekstraksi lebih maksimal.

3.3.1 Thresholding

Pada citra tomat diambil citra RGB dengan menggunakan *cropstat*, yaitu meng-*crop* bagian tertentu pada citra tomat. Penentuan *thresholding* dengan mengubah nilai *pixel* citra RGB yang masuk akan menghasilkan nilai 1 (putih) dan 0 (hitam) karena citra *threshold* mengubah citra RGB menjadi citra biner. Pada setiap citra RGB dihitung setiap nilai *pixel* red dikurangi green dan nilai *pixel* green dikurangi blue. Hasil pengurangan akan menemukan batas beda *pixel* maksimal 30. Nilai hasil pengurangan nilai *pixel* Red-Green dan Green-Blue yang mempunyai batas beda dibawah 30 akan dianggap sebagai *background* (nilai 0) sedangkan objek (nilai 1) adalah hasil dari pengurangan yang mempunyai batas beda diatas 30. Selanjutnya hasil pengurangan red-green dan green-blue dijumlahkan jika nilai pengurangan adalah nol maka menjadi *background* dan jika nilai besar dari 0 maka dianggap sebagai objek atau *foreground*.

3.3.2 Cropping

Setelah citra *dithresholding* selanjutnya dilakukan *cropping*. *Cropping* pertama menentukan nilai batas kiri-kanan dan atas-bawah dari objek. Menentukan batas kiri-kanan nilai biner dijumlah dari atas kebawah. Batas kiri adalah hasil penjumlahan nilai biner yang pertama bernilai 1 dan batas kanan kita tentukan nilai penjumlahan citra biner yang terakhir bernilai 1. Penentuan batas atas-bawah juga berlaku hal yang sama yaitu penjumlahan citra biner dari kiri kekanan, penjumlahan yang paling pertama mendapatkan nilai 1 akan menjadi batas atas dan yang terakhir bernilai 1 akan menjadi batas bawah. Setelah melalui proses *cropping* pertama selanjutnya citra diubah dalam bentuk sesuai dengan bentuk *cropping akhir* yang mengikuti bentuk terdalam dari buah tomat. Proses didapatkan dengan memberikan margin *crop* 20% dari citra tomat. Jadi setiap batas dari objek berkurang 20% sehingga hasil *cropping* mendapatkan bagian terdalam dari tomat. Berikut tampilan citra yang sudah mengalami proses *preprocessing*.



Gambar 3.2 (a) Citra Hasil Capturing; (b) Citra Hasil Thresholding (c) Citra Hasil Cropping awal (d) Citra Hasil Cropping Akhir

3.4 Ekstraksi Ciri

Proses ekstraksi ciri bertujuan untuk mengambil ciri citra yang berisi informasi tertentu. Ciri citra menjadi faktor identifikasi yang menentukan kondisi dari citra berupa nilai *pixel*. Nilai *pixel* yang dicari adalah nilai mean RGB, standar deviasi RGB dan varian RGB sehingga terdapat sembilan ciri yang diamati. Pada proses ekstraksi citra telah melewati proses *preprocessing*. Pencarian nilai mean dilakukan dengan menjumlahkan nilai *pixel* R dibagi dengan 256. Nilai *pixel* tersebut didapat dari mean, standar deviasi dan varian R, G dan B dari citra. Nilai-nilai tersebut mewakili *pixel-pixel* citra yang dijadikan sebagai acuan untuk proses klasifikasi. Kadar RGB didapat dari rata-rata citra RGB yang dipecah per-*pixel*nya. Jadi untuk satu citra akan terdapat sembilan ciri berupa nilai yang nantinya menjadi pembandingan untuk melakukan klasifikasi. Kadar setiap citra red, green dan blue ditentukan dari rata-rata RGB dibagi dengan jumlah mean RGBnya

3.5 Klasifikasi

Klasifikasi bertujuan untuk menentukan kelas yang tepat dari suatu citra uji berdasarkan ciri yang telah diekstraksi sebelumnya. Pada sistem hasil dari proses ekstraksi ciri disimpan dalam bentuk nilai *pixel* yang mewakili setiap ciri dari citra tomat yang telah diekstraksi. Ketika proses pengujian dilakukan hasil ekstraksi ciri dari citra uji dicocokkan dengan hasil ekstraksi ciri dari data citra pakar yang telah disimpan sebelumnya dengan mencari jarak minimum antara ciri citra uji dengan ciri pada database pakar yang telah dibuat kelas-kelas yang tersedia sehingga pengklasifikasian dapat dilakukan. Pengklasifikasian menggunakan *euclidean distance* menghitung jarak paling terdekat antara citra uji dengan jarak pada database yang telah dimasukkan kedalam kelas kelasnya. Jika citra uji mendekati jarak ciri pada tomat mentah maka citra uji dinyatakan sebagai tomat mentah.

3.6 Implementasi Interface pada sistem

Pada Desain GUI (*Graphical User Interface*) pada matlab ada 9 hasil yang ditampilkan yaitu nilai fitur dari mean, varian, standar deviasi, nilai kadar RGB, keputusan dan waktu komputasi dari sistem. Tombol capture pada tampilan digunakan mengaktifkan webcam untuk langsung melakukan pengambilan citra.



Gambar 3. 3 Pendeteksi Kematangan Tomat

3.7 Hasil Akhir Sistem dan parameter performansi

Setelah citra uji dikelompokkan ke dalam kelas klasifikasinya, kesimpulan akhir dengan memunculkan nilai *pixel* untuk mentah, setengah matang dan matang. Setiap alur pengujian dianalisa performansi kerjanya. Menentukan keberhasilan dari sistem dibutuhkan nilai performansi yang dijadikan sebagai penentu keberhasilan dari sistem yang dirancang jumlah data benar dan salah akan dibandingkan dengan data pakar dibagi total pengujian yang dilakukan. Parameter performansi dapat diketahui dengan cara:

$$Akurasi = \frac{\text{Jumlah data benar}}{\text{Jumlah data keseluruhan}} \times 100 \%$$

3.8 Waktu Komputasi

Waktu komputasi merupakan lamanya waktu yang dibutuhkan sebuah sistem dalam melakukan proses komputasi. Semakin singkat waktu komputasi yang dibutuhkan akan semakin baik. Perolehan waktu komputasi sistem digunakan program matlab pada setiap pengambilan empat sisi tomat. Total waktu pengambilan saat tomat sisi pertama diakuisisi, sampai diklasifikasi pada citra sisi ke empat dari penjumlahan waktu.

4. ANALISIS DATA DAN PEMBAHASAN

4.1 Analisis Data

Pembuatan, perancangan sistem dan implementasi program memerlukan Matlab 2013a dengan algoritma pemrograman yang telah dirancang. Pengguna sistem harus mengenal kegunaan aplikasi yang dioperasikan pada

program matlab dan terbiasa menggunakan aplikasi yang dioperasikan pada Matlab dan mengerti cara mengambil gambar yang tepat untuk aplikasi identifikasi tomat. *Webcam* yang digunakan untuk pengambilan data agar program dapat berjalan, dibutuhkan perangkat dengan spesifikasi dengan resolusi layar 640 x 480 *pixel*. Sebelum melakukan pengujian dilakukan percobaan untuk mengatur intensitas cahaya yang baik dan kondisi *pixel* kamera yang baik untuk digunakan. Percobaan dilakukan dengan 10 Tomat.

Tabel 4. 1 Akurasi Hubungan Pencahayaan dan Pixel Kamera

Intensitas Cahaya	Pixel Kamera			
	2 megapixel	3,2 megapixel	5 Megapixel	14 Megapixel
30-70 lux	43.33%	70 %	73.33%	73.33%
70-110 lux	53.33%	76.67%	80%	80%
110 -150 lux	73.33%	83.33%	86,67%	90 %

Semakin baiknya kualitas *pixel* kamera berbanding lurus dengan akurasi pada sistem. Percobaan intensitas cahaya yang dilakukan sudah baik dengan 70 lux sampai dengan 110 lux. Faktor cahaya mempengaruhi *noise* gelap dan terang pada citra tomat. Program pendeteksian pada proyek akhir ini memiliki beberapa fitur yang dapat diakses oleh pengguna. Dengan adanya fitur-fitur tersebut user dapat melakukan capturing tomat sehingga diperoleh citra tomat dan mengidentifikasi data tomat berdasarkan warna.

4.2. Alur Pengujian

Terdapat tiga dari jenis pengujian yang dilakukan menggunakan metode deteksi RGB :

1. Pengujian untuk mengetahui akurasi dan waktu komputasi sistem yang didapat dari pengaturan kondisi cahaya dan jarak. Tahap pertama dilakukan dengan jarak pengambilan citra 20 cm pada kondisi intensitas cahaya 30-70lx dan 70-110lx
2. Pengujian untuk mengetahui akurasi dan waktu komputasi sistem yang didapat dari pengaturan kondisi cahaya dan jarak. Tahap kedua dilakukan dengan jarak pengambilan citra 15,20,25 cm pada kondisi cahaya terang.
3. Pengujian untuk mengetahui akurasi dan waktu komputasi yang diperoleh melalui proses *cropping* dan tanpa *cropping* pada *pre-processing*-nya. Tahap ketiga dilakukan dengan mengambil citra pada kondisi pencahayaan terang dan jarak pengambilan 20 cm. Bagian yang di buang (*crop*) adalah *background*.

4.3 Hasil Pengujian dan Analisis

4.3.1 Hasil pengujian sistem terhadap kondisi pencahayaan

Pengujian dilakukan berdasarkan kondisi pencahayaan yang dilakukan saat pengambilan data uji dan pengaturan jarak *capture*. Adapun dua kondisi pencahayaan yang berbeda yaitu terang dan gelap dengan menggunakan *background* putih dengan jarak pengambilan citra sejauh 15 cm dari objek. Hasil pengujian diperoleh dengan menguji 50 citra uji untuk setiap kondisi pencahayaan.

Tabel 4. 2 Akurasi sistem berdasarkan kondisi pencahayaan

Pengujian	Intensitas Cahaya	Jumlah Citra Uji	Benar	Akurasi
1	70-110lx	50	44	88%
2	30-70lx	50	18	36%

Berdasarkan **Tabel 4.2**, terlihat bahwa akurasi paling baik terjadi pada kondisi pencahayaan terang (70-110lx) dengan angka 88%. Sedangkan pada kondisi redup (30-70lx) diperoleh akurasi yang kurang baik yaitu 36%. Perubahan kondisi pencahayaan tersebut mempengaruhi warna dari citra yang diambil. Kondisi pencahayaan yang mempengaruhi hasil *pre-processing* berdampak pada ekstraksi ciri yang dilakukan dengan metode RGB dan berubah-ubah mempengaruhi komponen warna dari citra. Semakin besar cahaya yang masuk semakin tinggi nilai pixel.

4.3.2 Hasil pengujian sistem terhadap perubahan parameter jarak

Pengujian dilakukan terhadap 50 citra uji yang masing-masing jarak memiliki 35 citra uji. Parameter jarak yang digunakan yaitu 15 cm, 20 cm dan 25 cm dengan kondisi pencahayaan terang. Penentuan jarak uji maksimal dipilih berdasarkan ukuran optimal. Sedangkan jarak minimum didapat dari percobaan pengambilan citra oleh kamera webcam yang mampu menampung ukuran citra tomat rata-rata yaitu 15 cm.

Tabel 4. 3 Akurasi sistem berdasarkan jarak

Pengujian	Jarak	Jumlah Citra Uji	Benar	Akurasi
1	15 cm	50	35	70 %
2	20 cm	50	44	88 %
3	25 cm	50	37	74%

Berdasarkan **Tabel 4.3**, terlihat bahwa akurasi paling baik terjadi pada jarak 20 cm yaitu 88%. Sedangkan pada jarak 15 cm dan 25 cm diperoleh akurasi yang kurang baik yaitu 70% dan 76% sehingga mempengaruhi ukuran citra yang diambil. Sistem identifikasi buah tomat menentukan ukuran citra yang diambil berdasarkan ukuran tomat pada saat proses deteksi berlangsung. Pada proses pengambilan citra dari pakar, ukuran citra yang diperoleh bermacam-macam disesuaikan dengan *database* yang telah diambil. Berdasarkan data pengujian diketahui bahwa pada jarak 20 cm diperoleh ukuran citra mendekati nilai citra pada data dari pakar yang juga diambil pada jarak 20 cm. Sedangkan untuk jarak 15 cm diperoleh ukuran citra yang lebih besar dibanding citra dari pakar dan untuk jarak 25 cm diperoleh ukuran citra diperoleh ukuran citra yang lebih kecil lagi dibanding citra dari pakar.

4.3.3 Hasil pengujian sistem terhadap *pre-processing*.

Pengujian dilakukan dengan 70 citra uji dengan 35 citra untuk masing-masing kondisi yaitu dengan *cropping* dan *non-cropping*. Parameter tetap dari pengujian yaitu kondisi pencahayaan terang dan pengambilan citra sejauh 20 cm dari objek.

Tabel 4. 4 Akurasi sistem berdasarkan *pre-processing*

Pengujian	<i>Pre-processing</i>	Jumlah Citra Uji	Benar	Akurasi
1	Cropping	30	26	86.67 %
2	Tanpa Cropping	30	20	66.67 %

Berdasarkan **Tabel 4.4**, terlihat bahwa akurasi paling baik terjadi pada pengujian *cropping* yaitu 86.67%. Sedangkan dengan menggunakan proses *non-cropping* diperoleh akurasi yang kurang baik yaitu 66.67%. Pada saat proses *pre-processing*, output citra berubah ukurannya sesuai dengan teknik *cropping*. Proses tersebut menghasilkan citra yang lebih kecil jika dibandingkan dengan data tanpa *pre-processing* karena proses *cropping* yang dilakukan secara otomatis menghasilkan output *cropping* dari setiap sistem. Proses *cropping* memudahkan untuk perhitungan *mean* pada warna tomat yang telah di ambil dengan *grayscale* data dan memaksimalkan *mean* pada warna objek.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Pengunaan metode perbandingan kadar warna RGB yang dilakukan. Dari hasil pengujian yang telah dilakukan, serta hubungan dari setiap pengujian. Maka penelitian tugas akhir yang berjudul SISTEM OTOMASI DALAM PENYORTIRAN TOMAT DENGAN IMAGE PROCESSING MENGGUNAKAN METODE DETEKSI RGB, hasil pengujian dan analisis yang dilakukan pada sistem, dapat diambil beberapa kesimpulan sebagai berikut:

1. Memanfaatkan image processing pada tomat dapat diidentifikasi dari kadar warna yang terdapat dalam citra. Tingkat akurasi sistem terbaik terdapat pada pengujian dengan jarak 50 lux dan 150 lux dengan kondisi terang dan *preprocessing* dengan nilai akurasinya adalah 88%, dan waktu komputasi rata-rata adalah. 4.5s.
2. Penggunaan *preprocessing* cukup mempengaruhi akurasi sistem, dibuktikan dengan perbedaan yang cukup signifikan dari hasil perhitungan akurasi penentuan kadar warna dengan *preprocessing* sebesar 88.67% dan menggunakan *preprocessing* sebesar 66.67%.
3. Sistem sangat rentan terhadap kondisi pencahayaan terutama ketika gelap, terbukti dari hasil pengujian gelap yang mencapai nilai 36%.
4. Berdasarkan pengujian sistem, tingkat akurasi sesuai dengan hasil yang diharapkan yakni mencapai diatas 80%.

5.2. Saran

Dalam setiap penelitian, pasti ditemukan kendala-kendala yang butuh kajian lebih mendalam dalam penyelesaiannya. Pada penelitian yang berjudul SISTEM OTOMASI DALAM PENYORTIRAN TOMAT DENGAN IMAGE PROCESSING MENGGUNAKAN METODE DETEKSI RGB, masih butuh dikembangkan dan perbaikan

dalam penelitian lanjutan. Saran-saran yang berkaitan dengan penelitian untuk ke depannya. Berikut dijelaskan dalam beberapa poin:

1. Pengkondisian cahaya harus pada rentang yang lebih kecil karena sedikit perbedaan intensitas cahaya mempengaruhi hasil akhir.
2. Melakukan *preprocessing* dengan metode lain yang lebih efektif.
3. Kondisi terang dibatasi lebih detil, untuk penelitian selanjutnya disarankan pada ruang tertutup dengan jumlah intensitas cahaya yang sama.
4. Melakukan pengujian pada mesin otomatisasi dan aplikasi android untuk mendapatkan akurasi yang lebih baik.

DAFTAR PUSTAKA

- [1] Saraswati, Yulia. (2007). *Sistem Klasifikasi Jenis dan Kematangan Buah Tomat Berdasarkan Bentuk dan Ukuran Serta Warna Permukaan Kulit Buah Berbasis Pengolahan Citra Digital*. Bandung : Institut Teknologi Telkom.
- [2] Deswari, Dila. (2010). *Identifikasi Kematangan Buah Tomat Menggunakan Metoda Backpropagation*. Padang : Politeknik Negeri Padang.
- [3] Kastaman, Roni & Fadhil Abdullah Wasi. (2009). *Analisis Kinerja Perangkat Pengolah Citra dengan Menggunakan Beberapa Metode Klasifikasi untuk Menentukan Kualitas Buah Manggis*. Bogor, 6-7 Agustus. Diakses melalui <http://repository.ipb.ac.id/handle/123456789/31881> pada 20 Mei 2013
- [4] Noviyanto, Ary. (2009). *Klasifikasi Tingkat Kematangan Varietas Tomat Merah dengan Metode Perbandingan Kadar Warna*. Jurusan Ilmu Komputer, UGM, Yogyakarta. Diakses melalui www.yovian.web.ugm.ac.id/system/application/.../RC_JM.pdf pada 20 Mei 2013.
- [5] Thiang, Leonardus Indrotanoto. (2008). *Otomatisasi Pemisah Buah Tomat Berdasarkan Ukuran dan Warna Menggunakan Webcam Sebagai Sensor*. Surabaya, 27 November 2008. Diakses melalui <http://portfolio.petra.ac.id/user.../Thiang-Pemisah%20Buah%20Tomat.pdf> pada 20 Mei 2013.
- [6] Rani, Novita. (2012). *Sistem Deteksi Jenis Buah Jeruk Berbasis Pengolahan Citra*. Semarang 16 Agustus 2012. Diakses melalui <http://eprints.unisbank.ac.id/1382/1/lengkap.pdf> pada 20 Mei 2013
- [7] Putra, Darma (2010). *Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi Offset.
- [8] Hery, Mauridhi Prurnomo (2010). *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta: Penerbit Ghara Ilmu
- [9] Jaenuri, Ahmad. *Noise pada Citra* <http://jayputracikgu.blogspot.com/2011/05/noise-pada-citra.html> (Diakses tanggal 8 Juli 2013)