

RANCANG BANGUN ALAT SORTASI OTOMATIS UNTUK BUAH TOMAT MENGUNAKAN APLIKASI *IMAGE PROCESSING*

Yultrisna¹, Andi Syofian²

¹Politeknik Negeri Padang

²Institut Teknologi Padang

e-mail: andisyofianmt@gmail.com

ABSTRAK

One of the stages of post-harvest activities were quite decisive is sorting. This process includes the separation of the product based on one or several criteria. Particularly in tomatoes, color and size parameters determine the quality of the fruit commodities. Along with the development of science and technology today, many developed with the sorting process automation systems for effectiveness in the use and accuracy of the results obtained. Tomato sorting machine which is realized using the camera as a sensor, the camera will take a picture of tomatoes through in line, and then a camera will be able to distinguish color and size of the tomatoes. Determination of color in tomatoes is doing by using the method of determining the threshold and size of tomato using edge detection method. Sorting is done by a servo motor to perform the separation of the tomatoes that have been detected in tomato set output based on the position that has been determined. Controlling which gave the order for the servo motor work is PIC16F84A microcontroller using serial data communication from the PC to the microcontroller. Testing the system as a whole to test the consistency of the performance of the machine have been conducted using seven different criteria of tomatoes with three times repetitions with successful results for 71% edge detection method, the grouping size is 100% and 95% for color readings.

Keywords: camera, image processing, edge detection, threshold, PIC16F84A microcontroller, servomotor

1. PENDAHULUAN

Potensi dan peluang pasar industri buah-buahan terus meningkat, yang disebabkan oleh bertambahnya konsumen dan meningkatnya pendapatan serta kesadaran masyarakat akan gizi. Hal ini menuntut tersedianya buah-buahan yang berkualitas baik dan mutu buah yang seragam. Buah-buahan tersebut diantaranya adalah buah tomat yang termasuk komoditas hortikultura yang bernilai ekonomi tinggi. Untuk menjaga kualitas tomat, selain pembudidayaan yang baik juga diperlukan perlakuan pascapanen yang baik pula.

Salah satu tahapan kegiatan pascapanen yang cukup menentukan adalah sortasi. Proses ini mencakup pemisahan produk berdasarkan satu atau beberapa kriteria misalnya berat, ukuran, warna dan sebagainya. Khususnya pada buah tomat, parameter warna sangat menentukan mutu buah, karena dapat menunjukkan tingkat kematangan buah dan daya tarik selera konsumen. Demikian juga parameter ukuran juga menentukan mutu komoditas tersebut karena ukuran yang seragam akan memudahkan proses pengemasan.

Pada umumnya proses sortasi tersebut masih menggunakan tenaga manusia dengan perkiraan-perkiraan berdasarkan pengalaman sehingga memerlukan banyak waktu dan tenaga serta memungkinkan hasil yang didapatkan dalam pemisahan tersebut tidak konstan. Sejalan dengan perkembangan ilmu dan teknologi dewasa ini, proses sortasi banyak dikembangkan dengan sistem otomasi dengan menggunakan perangkat elektronik dan mekanik untuk keefektifan dalam penggunaan, serta keakuratan hasil yang didapat.

Realisasi dari alat sortasi otomatis untuk buah tomat ini menggunakan aplikasi *image processing* dengan bantuan komputer dari citra buah tomat yang diambil menggunakan suatu alat perekam atau kamera dan suatu unit pengontrolan mekanik untuk melakukan proses penyortiran setelah buah tomat dikenali. Pengaplikasian *image processing* pada alat ini merupakan salah satu alternatif untuk penyederhanaan penggunaan sensor-sensor yang digunakan untuk mendukung proses otomasi dengan menggunakan beberapa metode *image processing* yaitu *thresholding* dan *edge detection* [1-8].

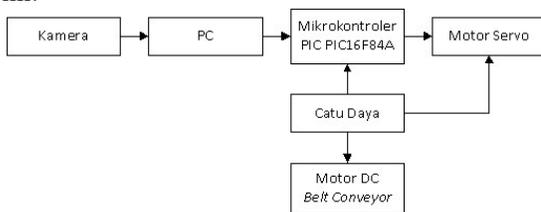
2. METODOLOGI

Alat sortasi otomatis untuk buah tomat ini dapat membedakan ukuran dan warna buah tomat dari hasil pengolahan gambar buah tomat yang diambil menggunakan kamera. Gambar tersebut menjadi input program pada PC untuk menentukan kriteria dari buah tomat sehingga ukuran dan warna dari buah tomat yang akan disortir dapat dibedakan. Penyortiran dilakukan oleh motor servo dengan melakukan pemisahan terhadap buah tomat yang sudah dideteksi dengan mengatur keluaran buah tomat berdasarkan posisi yang sudah ditentukan. Pengendali yang memberi perintah agar motor servo bekerja adalah mikrokontroler PIC16F84A, menggunakan komunikasi data serial dari PC ke mikrokontroler PIC16F84A. PC akan mengirimkan kode karakter dalam 1 byte data kepada mikrokontroler PIC16F84A, kemudian mikrokontroler PIC16F84A akan menjalankan perintah sesuai dengan kode karakter yang dikirimkan oleh PC.

Pada perancangan alat sortasi otomatis untuk buah tomat menggunakan aplikasi *image processing* ini terdiri dari dua bagian, yaitu perancangan perangkat keras (*hardware*), serta perancangan perangkat lunak (*software*).

2.1 Perancangan Hardware

Gambar 1 berikut menunjukkan blok diagram perangkat keras pada alat sortasi buah tomat ini.

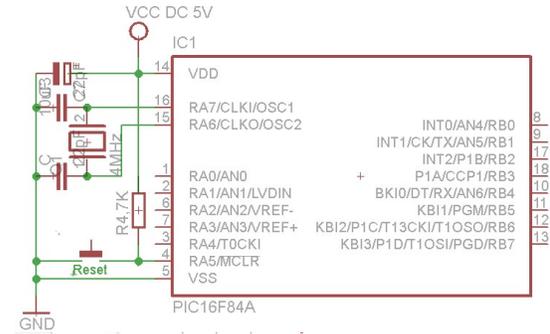


Gambar 1. Blok Diagram *Hardware*

Berikut adalah penjelasan komponen yang dipilih untuk membangun sistem pada gambar 1: Kamera yang digunakan adalah kamera *webcam* dengan resolusi 640 × 480. Kamera digunakan sebagai *input* untuk mengambil gambar dari tomat yang akan disortir. PC digunakan untuk mengolah gambar hasil pembacaan dari kamera dengan program *Visual Basic* dan mengakses perangkat keras dari komputer (misal: COM

Port dan LPT Port). Gambar tersebut nantinya akan berfungsi sebagai informasi kepada program *Visual Basic* untuk menentukan pengelompokan tomat yang dideteksi sebelum disortir.

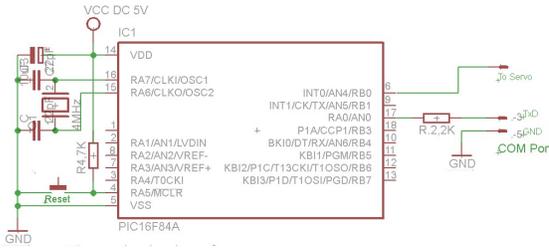
Mikrokontroler PIC16F84A ini digunakan untuk mengontrol sebuah motor servo standar 180° yang dipasang untuk menggerakkan selektor ke posisi yang ditentukan. PC akan mengirimkan perintah pada mikrokontroler PIC16F84A untuk menggerakkan motor servo berupa kode karakter dalam 1 byte data yang diatur secara *software*.



Gambar 2. Rangkaian Sistem Minimum PIC16F84A

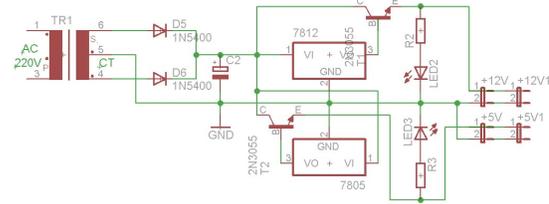
Pada Gambar 2 rangkaian sitem minimum terdapat sebuah sumber daya sebesar +5 Volt, sebuah resonator 4 Mhz sebagai sumber detak, sebuah resistor 4,7K Ohm yang dihubungkan antara pin MCLR dengan +5 Volt dan sebuah saklar sebagai saklar reset. Resistor 4,7K Ohm tersebut memang diperlukan agar pin MCLR berlogika tinggi sehingga mikrokontroler akan bekerja, jika pin MCLR berlogika rendah, maka mikrokontroler akan direset. Motor servo digunakan sebagai aktuator untuk menggerakkan selektor ke posisi keluaran masing-masing kelompok buah tomat yang telah dideteksi oleh program. Untuk menghubungkan mikrokontroler PIC16F84A ke PC digunakan port serial standar RS232 (COM1, dll). Pin RA0 pada PIC16F84A dihubungkan melalui resistor 22K langsung ke port serial “Tx”. Resistor 22K digunakan untuk membatasi arus yang melalui antarmuka langsung ke port serial PC. Mikrokontroler PIC16F84A menerima informasi dari PC untuk mengatur posisi motor servo yang terhubung ke pin RB0.

Berikut gambar rangkaian PIC16F84A untuk pengendali servo:



Gambar 3. Rangkain PIC16F84A untuk Pengendali Servo

Catu daya digunakan untuk kebutuhan mikrokontroler PIC16F84A, motor servo yang digunakan sebagai penggerak selektor dan motor *power window* sebagai penggerak *belt conveyor*. Output catu daya yang dihasilkan adalah tegangan DC 12V untuk motor *power window* dan tegangan DC 5V untuk mikrokontroler PIC16F84A dan motor servo.



Gambar 4. Rangkaian Catu Daya

Tegangan AC yang masuk ke penyearah jembatan telah diturunkan menjadi 12V AC terlebih dahulu dengan menggunakan transformator *step down*. Tegangan keluaran dari penyearah akan difilter dengan menggunakan kapasitor dan kemudian untuk mendapatkan tegangan 5V dan 12V yang stabil, maka digunakanlah IC regulator 7805 untuk keluaran 5V dc dan IC regulator 7812 untuk keluaran 12V dc. Arus keluaran dari IC regulator ini masih lemah, sehingga diperlukan suatu pelewat arus (*Current Bypass*), yaitu dengan menggunakan transistor 2N3055. Transistor tersebut membutuhkan tegangan basis-emitor (VBE) sebesar 0,7V untuk aktif, sehingga tegangan keluaran pada emitor terjadi penurunan sebesar 0,7 V. Selanjutnya tegangan akan di-filter oleh kapasitor untuk mendapatkan tegangan

keluaran catu daya dengan *ripple* yang lebih rata.

Belt conveyor digunakan untuk proses pengantaran buah tomat mulai dari peletakan awal, proses pengambilan gambar, pendeteksian ukuran dan warna buah tomat hingga sampai pada proses sortasi untuk menyortir buah tomat yang sudah dikenali. Kamera diletakan di bagian kanan dengan tinggi 20 cm diatas *belt conveyor* sehingga dapat menangkap gambar buah tomat secara utuh dan perbedaan ukuran dan warna dari buah tomat dapat terlihat dengan jelas. Sebagai penggerak *belt conveyor* digunakan motor *power window* yang mempunyai torsi yang cukup besar.

2.2 Perancangan Software

Perangkat lunak (*software*) yang digunakan pada alat sortasi buah tomat ini ada dua, yaitu Visual Basic 6.0 dan PICBasic *compiler*. Visual Basic 6.0 digunakan untuk membuat program tampilan webcam, program *image processing* dalam mendeteksi ukuran dan warna buah tomat, dan mengirimkan perintah pada mikrokontroler PIC16F84A untuk menggerakkan motor servo berupa kode karakter dalam 1 *byte* data, sedangkan PICBasic *compiler* digunakan untuk membuat program pengendali pergerakan motor servo pada mikrokontroler PIC16F84A.

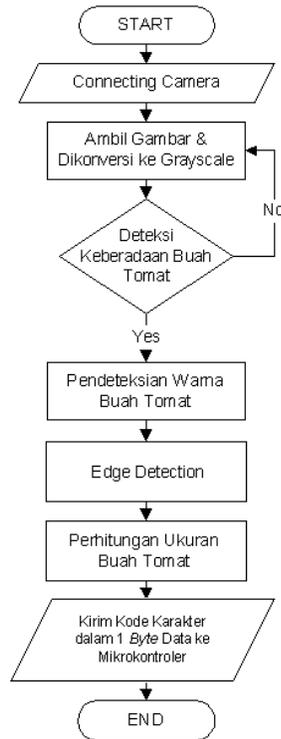
Proses yang akan dilakukan adalah pengambilan gambar dari kamera untuk mendeteksi adanya buah tomat yang lewat yang sebelumnya gambar tersebut dikonversi terlebih dahulu dari bentuk RGB ke dalam bentuk *grayscale* untuk menunjukkan tingkat intensitas keabuannya. Gambar *grayscale* didapatkan dengan merata-ratakan setiap komponen warna RGB menggunakan persamaan berikut:

$$Grayscale = \frac{R+G+B}{3} \tag{1}$$

Buah tomat akan terdeteksi dari perubahan piksel *grayscale* dari gambar yang ditangkap oleh kamera. Pengenalan objek (buah tomat) akan dilakukan oleh komputer yang sudah diprogram yaitu pengenalan warna merah dan hijau dari penentuan nilai *threshold* yang secara umum proses *thresholding* citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut:

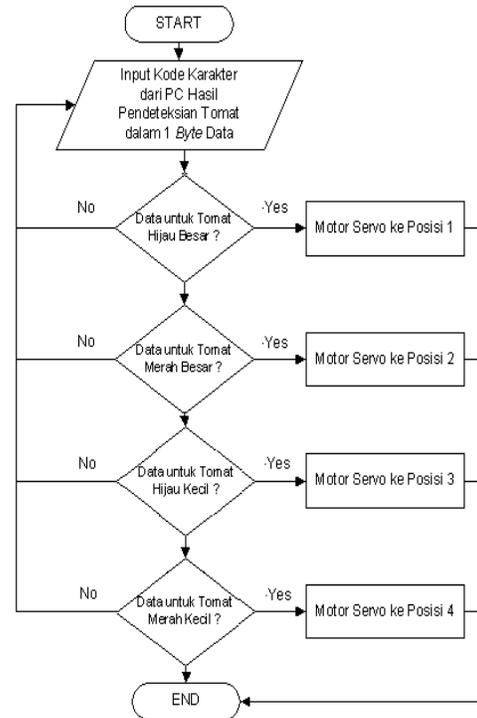
$$g(x,y) = \begin{cases} 1 & \text{untuk } f(x,y) \geq T \\ 0 & \text{untuk } f(x,y) < T \end{cases} \quad (2)$$

Untuk pengenalan ukuran besar dan kecil dengan metode *edge detection* dan yang hasilnya digunakan untuk dibandingkan dengan nilai yang ditentukan sebelumnya.



Gambar 5. Flowchart Perancangan Software

Mikrokontroler PIC16F84A akan memberi perintah kepada aktuator yaitu motor servo untuk menggerakkan selektor ke posisi masing-masing kelompok buah tomat. Mikrokontroler PIC16F84A mendapat perintah dari PC berupa kode karakter dalam 1 byte data agar dapat menggerakkan motor servo. Untuk mendapatkan sudut motor servo yang sesuai dengan konstruksi dari *belt conveyor* maka kode karakter tersebut diperoleh dengan *trial and error*. Gambar 6 menunjukkan *flowchart* komunikasi antara PC dengan mikrokontroler PIC16F84A.



Gambar 6. Flowchart Komunikasi Antara PC dengan PIC16F84A

3 HASIL DAN PEMBAHASAN

Pengujian dilakukan dengan tujuan untuk mengetahui apakah sistem yang telah dirancang dapat berfungsi atau bekerja dengan baik sesuai dengan yang diharapkan. Analisa didapatkan dari hasil pengujian program dan alat yang telah dibuat.

3.1 Pengujian Catu Daya

Sumber tegangan yang digunakan berasal dari transformator dan rangkaian regulator. Transformator mengeluarkan tegangan sebesar 18V yang digunakan sebagai *input* tegangan untuk regulator. Rangkaian regulator memiliki dua *output* tegangan yang berbeda sebesar 12V dan 5V yang digunakan untuk mencatu motor power window, motor servo, dan sistem minimum mikrokontroler PIC16F84A. Pengujian dilakukan dengan mengukur *output* pada rangkaian regulator menggunakan multimeter. Berikut Tabel 1 hasil uji coba terhadap catu daya.

Tabel 1: Hasil Pengukuran *Output* Catu Daya

Regulator	Hasil Seharusnya	Tanpa Beban	Berbekan
IC LM7812	12 V	11.3 V	10.9 V
IC LM7805	5 V	4.5 V	4.3 V

Menurut *datasheet* IC regulator 78xx hanya mampu mengeluarkan arus maksimal 1A tentunya arus sebesar ini tidak sanggup digunakan untuk menyuplai kebutuhan pada sistem ini, salah satunya motor *power window* yang membutuhkan arus yang cukup besar, untuk menanggulangi hal itu perlu dipasang transistor 2N3055 untuk melewati arus sehingga bisa memenuhi kebutuhan sistem.

Output yang seharusnya dari IC regulator 7812 adalah 12V dan IC regulator 7805 adalah 5V tetapi setelah dilakukan pengukuran terhadap output catu daya tanpa beban maka didapatkan hasil 11.3V dan 4.5V. Penurunan tegangan ini melewati batas toleransi minimum dari IC regulator 7812 yaitu 11.4V dan batas toleransi minimum 4.8V untuk IC regulator 7805. Pengukuran selanjutnya dilakukan dengan memasang semua beban yang akan digunakan pada alat ini. Tegangannya turun hingga 10.9V dan 4.3V. Kekurangan tegangan output regulator bisa terjadi karena pengaruh oleh beban-beban yang digunakan, toleransi dari IC regulator itu sendiri dan kebutuhan VBE (0,7V) pada transistor yang digunakan.

Tegangan catu daya pada kondisi berbeban, ternyata masih dalam jangkauan logika HIGH, sehingga semua tegangan, baik tanpa beban atau berbeban masih memenuhi syarat untuk dapat dianggap sebagai tegangan logika HIGH pada level TTL untuk mikrokontroler yaitu 3V s/d 5V. Dari pengujian dan pengambilan data di atas dapat disimpulkan bahwa catu daya yang terdiri dari transformator dan regulator dapat difungsikan untuk mencatu komponen yang membutuhkan asupan tegangan.

3.2 Pengujian Program Pendeteksian Warna dan Ukuran

Pengujian untuk melihat kemampuan program dalam membedakan warna dan ukuran dilakukan dengan kertas warna hijau dan merah, kertas yang digunakan adalah

kertas warna bulat dengan diameter yang ditentukan sebagai acuan untuk membaca nilai *pixel* pada masing-masing warna buah tomat dan pendeteksian ukuran buah tomat. Hasil pengukuran dengan pengulangan sebanyak tiga kali disajikan pada tabel berikut :

Tabel 2: Hasil Pembacaan Kertas Warna

Warna Kertas	Diameter Kertas	Gambar <i>Edge Detection</i>	Jumlah <i>pixel</i> yang Terbaca		
			I	II	III
Merah	3 cm		774571	757856	767700
	4 cm		757800	743728	763324
	5 cm		743324	757800	739793
	6 cm		739793	743324	742658
Hijau	3 cm		646546	657590	650108
	4 cm		647558	655211	653834
	5 cm		653834	657736	646652
	6 cm		637256	644231	658351

Dari tabel 2, dapat diketahui bahwa nilai jumlah *pixel* berubah nyata dengan berubahnya warna, warna merah menghasilkan jumlah *pixel* antara 560000 s/d 599000 dan hijau antara 730000 s/d 775000. Dari hasil pendeteksian tersebut dapat dijadikan acuan batas nilai pada program untuk pendeteksian buah tomat berdasarkan warnanya dengan kondisi pencahayaan yang sama. Batas nilai yang digunakan untuk pendeteksian warna pada kondisi ini adalah 650000 dimana jumlah *pixel* tomat berwarna merah lebih kecil dan jumlah *pixel* tomat berwarna hijau lebih besar dari nilai tersebut.

Berikut ini adalah hasil pembacaan ukuran pada program untuk menentukan ukuran sebenarnya dari objek dengan menentukan skala perbandingannya sehingga

ukuran sebenarnya dapat ditampilkan oleh program.

Tabel 3: Hasil Pembacaan Ukuran pada Program

Warna Kertas	Diameter Kertas (cm)	Hasil Pembacaan Program				Rata-rata (cm)
		Diameter				
		Diameter (pixel)		Diameter (cm)		
		a	b	a	b	
Merah	3	154	156	3,0	3,0	3,0
	4	194	196	4,0	4,0	4,0
	5	248	246	5,0	5,0	5,0
Hijau	6	284	289	6,0	6,1	6,05
	3	153	160	3,0	3,1	3,05
	4	196	199	4,0	4,0	4,0
Hijau	5	275	267	5,0	5,0	5,0
	6	285	295	6,0	6,1	6,05

Hasil pembacaan program mendekati ukuran objek yang sebenarnya yang dapat dikatakan berhasil apabila pada program hasil *edge detection* terlihat pada daerah kanan, kiri, atas dan bawah objek. Pada program dapat dikelompokkan ukuran objek berdasarkan besar dan kecil dengan mengambil nilai tengah yaitu 4,5 cm yang dijadikan batas nilai untuk menentukan kelompok kecil atau besar.

3.3 Pengujian Pengontrolan Motor Servo

Tujuan dari pengujian ini adalah untuk mengetahui perbedaan besar sudut motor servo untuk dapat menggerakkan selektor pemisah ke posisi kelompok-kelompok buah tomat yang sudah dikenali program berdasarkan pengiriman karakter yang diterima oleh mikrokontroler PIC16F84A. Program pengujian pengiriman karakter ke mikrokontroler ini mengirimkan 2 *byte* data, 1 *byte* untuk informasi nomer servo dan 1 *byte* untuk informasi pengaturan posisi servo. Nomer servo yang digunakan adalah 0 karena pada mikrokontroler port yang digunakan untuk pengontrolan servo adalah port B.0 dan pengaturan posisi motor servo disesuaikan dengan konstruksi *conveyor*.

Pembuatan program untuk mikrokontroler PIC16F84A untuk pengaturan servo ini menggunakan *software* PICBasic. Instruksi PULSEOUT pada PICBasic yang digunakan pada program pengontrolan servo ini mempunyai perioda 10 μ s saat osilator yang digunakan adalah 4Mhz. Dari pengujian juga didapatkan kode

karakter yang mengatur putaran servo ke sudut 0° (minimal) dan 180° maksimal dari motor servo yaitu Chr\$(46) dan Chr\$(255). Dari data tersebut dapat diketahui lebar pulsa untuk gerak per derajat, lebar pulsa yang diberikan untuk menggerakkan motor servo ke posisi kelompok keluaran buah tomat, dan besar sudut masing-masingnya.

Tabel 4: Hasil Pengujian Pengiriman Kode Karakter untuk Pengaturan Posisi Servo

Posisi Selektor	Kode Karakter
0° (sudut minimum)	Chr\$(0) & Chr\$(46)
180° (sudut maksimum)	Chr\$(0) & Chr\$(255)
1	Chr\$(0) & Chr\$(152)
2	Chr\$(0) & Chr\$(134)
3	Chr\$(0) & Chr\$(117)
4	Chr\$(0) & Chr\$(101)

Dari uji coba yang dilakukan, pergerakan motor servo sesuai dengan *input* yang diberikan dari PC. Dapat disimpulkan bahwa motor servo memiliki akurasi yang baik dan dapat digunakan untuk pengujian selanjutnya.

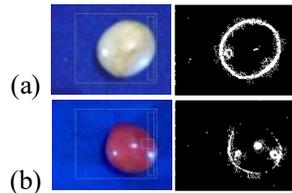
3.4 Pengujian Sistem Secara Keseluruhan

Pengujian terakhir adalah pengujian sistem secara keseluruhan dari awal hingga akhir, dimana pengujian dilakukan mulai dari pembacaan ukuran dan warna buah tomat sampai pada proses penyortiran buah tomat oleh selektor yang digerakkan motor servo ke kelompok keluaran masing-masing buah tomat. Pengujian ini menggunakan tujuh buah tomat berbagai warna dan ukuran dengan tiga kali perulangan untuk menguji konsistensi pembacaan program.

Pada pengujian ini didapatkan perbedaan jumlah *pixel* pada warna yang sama antara kulit buah tomat dan kertas warna yang diuji sebelumnya. Hal ini disebabkan adanya perbedaan pada kejenuhan warna dan dari segi kecemerlangan maupun bentuk permukaannya yang tidak seragam, dimana kertas relatif seragam.

Pada hasil pengujian sistem secara keseluruhan didapatkan hasil dimana metode *edge detection* tidak bekerja dengan baik pada beberapa buah tomat sehingga program pada pengujian ini tidak dapat menyesuaikan perhitungan ukuran dengan ukuran sebenarnya dari buah tomat tersebut. Berikut

contoh gambar hasil *edge detection* pada buah tomat:



Gambar 7. Contoh Hasil *Edge Detection* pada Buah Tomat (a) Sesuai (b) Tidak Sesuai

Hasil pembacaan program akan mendekati ukuran objek yang sebenarnya apabila pada program hasil *edge detection* terlihat pada daerah kanan, kiri, atas dan bawah. Persentase tingkat keberhasilan metode *edge detection* pada pengujian ini adalah 71%.

Program untuk mengelompokkan buah tomat dalam kategori ukuran besar dan kecil dan kategori warna merah dan hijau pada pengujian ini dapat dikatakan berhasil karena hanya terdapat terdapat satu kesalahan dari pengujian dimana pada pendeteksian buah tomat hijau terdeteksi sebagai tomat merah karena permukaan buah tomat yang ditangkap kamera untuk mendeteksi warna memiliki bercak hitam yang otomatis mengubah nilai jumlah *pixel* menjadi lebih rendah dari nilai batas antara warna merah dan warna hijau. Posisi selektor pemisah akan selalu mengikuti pembacaan program sehingga juga terdapat satu kesalahan posisi karena kesalahan pembacaan warna tersebut.

Pada pengujian di atas juga menguji tomat setengah matang yang masuk dalam kategori warna hijau karena jumlah *pixel*nya hampir sama dengan tomat yang berwarna hijau. Persentase tingkat keberhasilan pengelompokan ukuran adalah 100% dan 95% untuk pembacaan warna.

4. KESIMPULAN

Berdasarkan hasil perancangan dan pengujian alat sortasi otomatis untuk buah tomat ini dapat disimpulkan bahwa:

1. Pengujian sistem secara keseluruhan telah dilakukan menggunakan tujuh buah tomat yang memiliki kriteria berbeda sebanyak tiga kali pengulangan dengan hasil keberhasilan metode *edge detection* 71%, pengelompokan ukuran adalah 100% dan 95% untuk pembacaan warna.
2. Kesalahan pada pengujian dapat terjadi karena beberapa faktor, yaitu: pencahayaan yang kurang baik, setting nilai *threshold* yang tidak tepat, dan tidak meratanya warna buah tomat.
3. Pengukuran hanya dilakukan sebatas luas benda dari satu sisi yang ditangkap oleh kamera untuk menentukan ukuran besar atau kecilnya buah tomat.

DAFTAR PUSTAKA

- [1] Agustina L, *Penerapan Region of Interest (ROI) pada Metode Kompresi JPEG2000*, ITB, Bandung.
- [2] Ahmad Usman, *Pengolahan Citra Digital & Teknik Pemrogramannya*, Graha Ilmu, Yogyakarta, 2005.
- [3] Indrotanoto L, *Pemisah Buah Tomat Berdasarkan Ukuran dan Warna Menggunakan Webcam Sebagai Sensor*, Universitas Kristen Petra, Surabaya, 2008.
- [4] Kanara Nahda, *Pengemasan dan Penyimpanan Buah Tomat*, UGM-Press, Yogyakarta, 2006.
- [5] Nagantara Jemie, *Sistem Kerja PID pada Rangkaian Penggerak Servo untuk Motor DC*, BINUS, Jakarta, 2010.
- [6] Putra Darma, *Pengolahan Citra Digital*, ANDI, Yogyakarta, 2010.
- [7] Retna P dan Catur Edi W, *Interfacing Port Paralel dan Port Serial Komputer dengan Visual Basic 6.0*, ANDI: Yogyakarta, 2004.
- [8] Sutejo Agus, *Rekayasa Alat Sortasi Buah Jeruk (Citrus Nobilis) Berdasarkan Warna*, IPB, Bogor, 1999.