

Perancangan dan Pembuatan Sistem *Visual Inspection* Sebagai Seleksi Buah Tomat Berdasarkan Kematangan Berbasis *Web Camera*

Arif Zaenury Ichsan¹, Andrizal, MT², Dodon Yendri, M.Kom³

^{1,3} Jurusan Sistem Komputer FTI Universitas Andalas
Jln. Kampus Limau Manis Kota Padang 25163 INDONESIA

²Elektro Politeknik Negeri Padang Kampus UNAND Limau Manis Padang 25163
E-mail : ¹krypto304@ymail.com, ²andrizal@polinpdg.ac.id, ³dodon_y@yahoo.com

Abstrak— Teknologi saat ini telah mengalami perkembangan yang sangat pesat, diantaranya adalah pengembangan *image processing* dan sistem otomasi dalam menentukan kematangan suatu buah. Proses penentuan kematangan buah tomat menggunakan metode SSE dan Histogram yang diintegrasikan ke sensor gp2y dan rangkaian minimum beserta *webcam*. Proses integrasi sistem dilakukan dengan cara menghubungkan sensor, *conveyor*, motor, dan komponen lainnya dengan sistem minimum dan mengkoneksikan *webcam* ke PC. Proses penyeleksian buah Tomat dilakukan dengan cara membandingkan nilai RGB sampel dengan nilai database untuk menentukan matang atau tidak matangnya buah tomat. Hasil uji menunjukkan tingkat keberhasilan 80% s/d 100% berhasil mendeteksi dan melakukan seleksi terhadap matang dan tidak matangnya buah tomat.

Kata kunci : *image processing, sse, histogram, sensor gp2y, webcam*

I. PENDAHULUAN

Di zaman yang semakin canggih ini, teknologi berkembang dengan sangat pesat. Kebutuhan manusia juga semakin banyak yang bergantung dengan teknologi, baik dalam bidang komunikasi, pendidikan, bahkan dalam bidang perkebunan. Sampai saat ini teknologi yang membantu manusia di bidang perkebunan kurang berkembang.

Buah-buahan merupakan suatu komoditas yang menguntungkan karena keaneragaman varitas dan didukung oleh iklim yang sesuai, sehingga menghasilkan berbagai buah-buahan yang sangat bervariasi dan menarik.

Buah tomat merupakan salah satu produk hortikultura yang mempunyai prospek pemasaran. Penanganan pasca panen memegang peranan penting dalam penentuan mutu tomat, terutama kegiatan seleksi dan pemutuan. Selama ini kegiatan seleksi dan pemutuan buah tomat dilakukan secara manual, sehingga menghasilkan produk yang kurang seragam. Karena hasil seleksi manual yang

kurang memuaskan, maka diperlukan suatu metode untuk menseleksi dan mengelompokkan tomat dengan baik. Perkembangan metode pengolahan citra memungkinkannya dilakukan penentuan tingkat kematangan buah tomat tanpa merusaknya^[1]

Salah satu cara yang dapat menjawab permasalahan tersebut adalah proses otomasi. Sebelum sistem otomasi diterapkan, sistem inspeksi visual di industri banyak dilakukan secara manual oleh manusia. Sistem inspeksi oleh manusia mempunyai berbagai kelemahan yang disebabkan oleh banyak faktor, seperti kelelahan, ketiadaan motivasi, pengalaman, kemahiran dan lain-lain. Untuk mengatasi masalah ini, maka sistem inspeksi visual di industri telah digantikan secara efektif oleh sistem otomatis^[19]

II. TINJAUAN PUSTAKA

2.1 Citra, Pengolahan Citra, dan Pengenalan Pola

Definisi citra adalah suatu representasi, kemiripan, atau imitasi dari suatu objek atau benda. Secara umum citra dapat dibagi menjadi dua macam, yaitu :

1. Citra kontinyu, yaitu citra yang dihasilkan dari sistem optik yang menerima sinyal analog, misalnya gambar pada monitor televisi, foto sinar X, hasil CT Scan, mata manusia, kamera analog dll.
2. Citra diskrit, yaitu citra yang dihasilkan melalui proses digitalisasi terhadap citra kontinyu dan direpresentasikan sebagai sebuah matrik.^[2]

Pada bidang komputer, ada tiga bidang studi yang berkaitan dengan data citra, yaitu :^[2]

1. Grafika Komputer

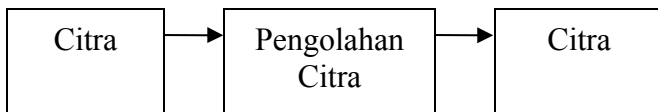
Yaitu proses untuk menciptakan suatu gambar berdasarkan deskripsi objek maupun latar belakang pada gambar tersebut dan merupakan teknik untuk membuat gambar objek sesuai dengan objek tersebut di alam nyata (realism). Primitif-primitif geometri tersebut memerlukan data deskriptif untuk melukis elemen-elemen gambar. Grafika komputer berperan dalam visualisasi dan *virtual reality*. Berikut gambar 2.2 alur grafik komputer.



Gambar 2.1 Alur Grafika Komputer^[2]

2. Pengolahan Citra (image processing)

Pengolahan citra pada dasarnya dilakukan dengan cara memodifikasi setiap titik dalam citra yang bertujuan untuk memperbaiki kualitas citra sehingga lebih mudah diinterpretasikan oleh manusia dan mesin (komputer). Teknik-teknik pengolahan citra biasanya mentransformasikan citra menjadi citra lain. Jadi input dan outputnya adalah sama-sama berupa citra. Berikut gambar 2.3 alur pengolahan citra

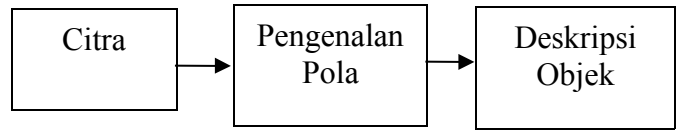


Gambar 2.2 Alur Pengolahan Citra^[2]

3. Pengenalan Pola

Yaitu mengelompokkan data numerik dan simbolik (termasuk citra) secara otomatis oleh mesin (komputer). Tujuan pengelompokkan adalah untuk mengenali suatu objek di dalam citra. Komputer menerima masukan berupa citra objek

yang akan diidentifikasi, memproses citra tersebut dan memberikan keluaran berupa deskripsi objek di dalam citra. Berikut gambar 2.4 alur pengenalan pola.



Gambar 2.3 Alur Pengenalan Pola^[2]

2.2 Jenis citra

Nilai suatu *pixel* memiliki nilai dalam rentang tertentu, dari nilai minimum sampai nilai maksimum. Jangkauan yang digunakan berbeda-beda tergantung dari jenis warnanya. Citra dengan penggambaran seperti ini digolongkan kedalam citra integer. Berikut adalah jenis-jenis citra berdasarkan nilai *pixel*-nya, sebagai berikut ^[2]:

a. Citra biner

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai *pixel* yaitu hitam dan putih. Citra biner juga disebut sebagai citra B&W (*black and white*) atau monokrom.

b. Citra grayscale

Citra *grayscale* member kemungkinan warna yang lebih banyak daripada citra biner, karena ada nilai-nilai lain diantara nilai minimum (biasanya = 0) dan nilai maksimumnya. Citra *grayscale* disebut juga citra keabuan karena pada umumnya warna yang dipakai adalah antara hitam sebagai warna minimal dan warna putih sebagai warna maksimal, sehingga warna antaranya adalah abu-abu.

Ada beberapa macam untuk mengkonversi system warna RGB menjadi *grayscale* yaitu:

1. Dengan merata-rata setiap komponen warna RGB

$$\text{Grayscale} = \frac{r + g + b}{3}$$

2. Dengan nilai maximal dari nilai RGB $\text{Grayscale} = \text{Max}[R, B, G]$
3. Dengan menggunakan YUV (sistem pada NTSC) yaitu dengan cara mengambil komponen Y (*iluminasi*). Komponen Y dapat diperoleh dari system warna RGB dengan konversi:

$$\text{Grayscale} = 0.299R + 0.587G + 0.114B$$

c. Citra warna (16bit)

Citra warna 16 bit (biasanya disebut sebagai citra high colour) dengan setiap *pixel* diwakili dengan 2 *byte memory* (16bit). Warna 16 bit memiliki warna 65536 warna. Dalam formasi bitnya, nilai merah dan biru mengambil tempat di 5 bit di kanan dan di kiri. Komponen hijau memiliki 5 bit ditambah 1 bit ekstra. Pemilihan komponen hijau dengan deret 6 bit dikarenakan penglihatan manusia lebih sensitif terhadap warna hijau.

2.3 Histogram Citra

Histogram citra adalah grafik yang menggambarkan penyebaran nilai-nilai intensitas *pixel* dari suatu citra atau bagian tertentu di dalam citra. Dari sebuah histogram dapat diketahui frekuensi kemunculan nisbi (*relative*) dari intensitas pada citra tersebut. Histogram juga dapat menunjukkan banyak hal tentang kecerahan (*brightness*) dan kontras (*contrast*) dari sebuah gambar. Karena itu, histogram adalah alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif.

2.4 Metoda Summary Squared Error (SSE)

SSE (*Sum Square Error*) adalah salah satu metode statistik yang dipergunakan untuk mengukur selisih total dari nilai sebenarnya terhadap nilai yang tercapai. Istilah SSE disebut juga sebagai *Summed Square of Residuals*.^[23]

$$SSE = \sum_{i=1}^n (X_i - Y_i)^2$$

Dimana,

X = nilai aktual atau sebenarnya

Y = nilai yang tercapai

Nilai X dalam penelitian ini adalah nilai yang tersimpan dalam database sedangkan nilai Y adalah komponen data uji. Nilai SSE yang mendekati 0 menandakan bahwa model tersebut mempunyai komponen kesalahan acak terkecil dan nilai tersebut akan lebih berguna untuk peramalan terhadap suatu model yang diamati. Sebagai catatan bahwa sebelumnya SSE didefinisikan dalam metode kelayakan kuadrat minimum.

2.5 Analog to Digital Converter

Salah satu komponen penting dalam sistem akuisisi data adalah pengubah sinyal analog menjadi sinyal digital atau disebut juga ADC (*Analog to Digital Converter*). Pengubah ini mengubah sinyal-sinyal analog menjadi sinyal digital sehingga dapat diproses oleh komputer.

2.6 Pemodelan Sistem Inspeksi Visual

Sistem penglihatan manusia adalah suatu sistem penglihatan yang memiliki mekanisme sangat kompleks. Terlebih lagi jika mekanisme sistem penglihatan tersebut direalisasikan ke dalam sistem komputer.^[8]

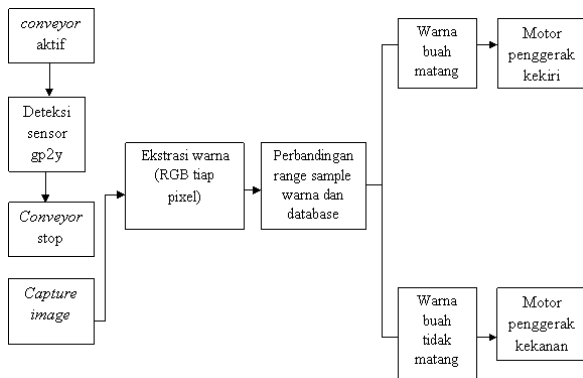
2.7 Komunikasi Port Paralel

Port paralel banyak digunakan dalam berbagai macam aplikasi antarmuka. Port ini membolehkan kita memiliki masukan hingga 8 bit atau keluaran hingga 12 bit pada saat yang bersamaan, dengan hanya membutuhkan rangkaian eksternal sederhana untuk melakukan suatu tugas tertentu. Port paralel ini terdiri dari 4 jalur kontrol, 5 jalur status dan 8 jalur data. Biasanya dapat Anda

jumpai sebagai port pencetak (*printer*), dalam bentuk konektor DB-25 betina (*female*). Port paralel yang baru, distandarisasi dengan IEEE.1284 yang dikeluarkan pada tahun 1984. Standar ini mendefinisikan 5 macam mode operasi sebagai berikut:^[22]

III. METODE PENELITIAN DAN PERANCANGAN

Perancangan sistem ini menggunakan pendekatan model proses sekuensial linier. Untuk lebih jelasnya sistem ini dapat dilihat pada gambar 3.1.

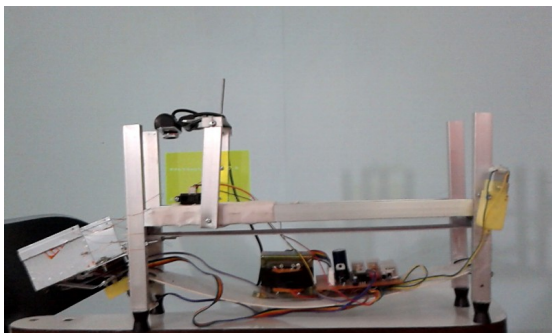


Gambar 3.1 Diagram Uji Selektor Kematangan Buah Tomat

Untuk perancangannya sendiri, juga terdiri atas 3 bagian, yaitu :

3.1 Perancangan Mekanik

Perancangan sistem dimulai dari perancangan mekanik, berdasarkan blok diagram pada Gambar 3.1 dapat digambarkan rancangan alat yang akan dibuat. Adapun rancangan mekanik tersebut dapat dilihat pada gambar 3.2

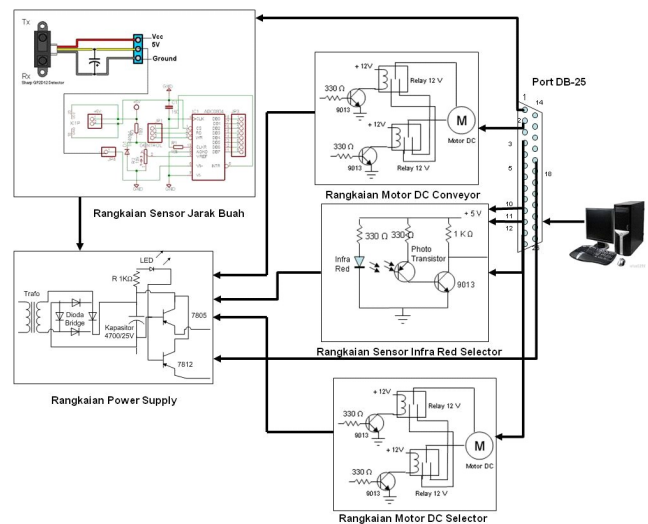


Gambar 3.2 Rancangan Alat Secara Keseluruhan

Pada Gambar 3.2 terlihat bagaimana rancangan mekanik yang telah terpasang sedemikian rupa, *belt conveyort* yang dipakai terbuat dari kain sistetis yang telah dipasang motor DC sebagai penggerak, letak selektor berada sesudah webcam, sedangkan posisi webcam dengan belt $\pm 8\text{cm}$ dan sensor gp2y diposisikan vertical di atas webcam. Pada rangkaian catudaya dan ADC tepat berada di bawah *conveyor belt*. Koneksi dari rangkaian ke PC menggunakan port paralel db-25.

3.2 Perancangan Hardware

Blok diagram sistem secara keseluruhan dapat dilihat pada gambar berikut :



Gambar 3.3 Diagram sistem Secara Keseluruhan

Pada perancangan hardware terdapat beberapa rangkaian yang akan dipakai diantaranya, rangkaian motor dc untuk konveyor dan selektor, rangkaian infra red selektor, rangkaian untuk sensor gp2y dan rangkain untuk power supply.

3.3 Perancangan Software

3.3.1 Inisialisasi Sistem

Pada tahap ini dilakukan penginisialisasian sistem agar software pemrograman dengan borland delphi dapat berinteraksi dengan mekanik, baik itu rangkaian maupun web camernya. Inisialisasi

sistem dilakukan dengan cara men-setting port seperti dapat kita lihat pada penggalan program dibawah ini :

Deklarasi IO.DLL

```
procedure
PortOut(Port: Word;Data: Byte);stdcall;external'io.dll';
```

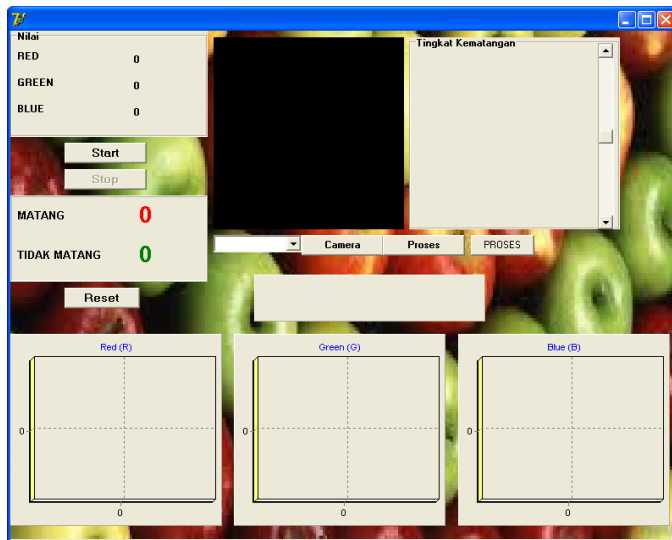
```
function
PortIn(Port: Word):byte;stdcall;external'io.dll';
```

Program Nilai Port Input DB-25

```
procedure TForm1.TStatusKontrolTimer(Sender: TObject);
var status,kontrol:Integer;
begin
status:=portin($379);
kontrol:=portin($37A);
LStatus.Caption :=inttostr(status);
LKontrol.Caption :=inttostr(kontrol);
End ;
```

3.2.2 User Interface Sistem

Berikut merupakan bentuk *User Interface* program delphi yang akan digunakan dalam proses pencitraan untuk mendapatkan jumlah *RGB* pada suatu objek yang nantinya akan digunakan dalam penentuan kematangan buah tomat.



Gambar 3.4 User Interface Sistem

Pada *user interface* terdapat beberapa menu, diantaranya adalah tombol *Camera* yang digunakan untuk mengaktifkan program. Tombol *Proses* untuk melihat nilai *RGB*. Tombol *Start* digunakan untuk menggerakkan *belt conveyor*. Tombol *Stop* untuk menghentikan *belt conveyor*. Tombol reset digunakan untuk mereset counter.

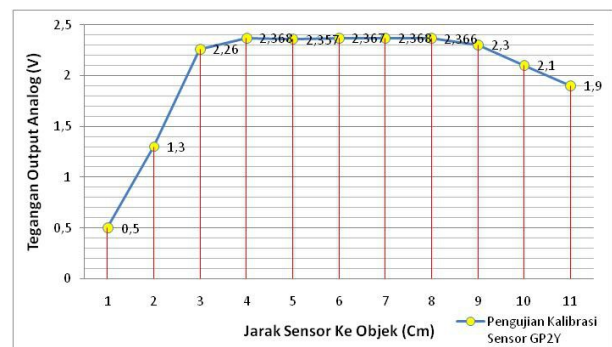
IV. HASIL DAN ANALISA

4.1 Pengujian Rangkaian Sistem

Pengujian rangkaian sistem dilakukan untuk mengetahui bahwa alat yang dibuat dapat berjalan dengan baik dan tidak menimbulkan masalah pada saat pengambilan data.

4.1.1 Pengujian Rangkaian Sensor GP2Y

Sensor jarak GP2Y memiliki tegangan input yang di berikan harus konstan dengan nilai tegangan masukan 5V. Pengujian kalibrasi sensor dilakukan dengan cara membandingkan nilai jarak pendeteksian sensor dengan nilai voltage nya yang diukur menggunakan voltmeter. Dari sini akan didapatkan posisi optimal sensor untuk dapat bekerja secara maksimal. Adapun kalibrasi sensor dapat kita lihat pada tabel I



Gambar 4.1 Grafik Pengujian Kalibrasi Sensor GP2Y

TABEL I
HASIL KALIBRASI SENSOR

Jarak Pengukuran (Cm)	Tegangan Output (V)
1 Cm	0,5 V
2 Cm	1,3 V
3 Cm	2,26 V
4 Cm	2,368 V
5 Cm	2,357 V
6 Cm	2,367 V
7 Cm	2,368 V
8 Cm	2,366 V
9 Cm	2,3 V
10 Cm	2,1 V
11 Cm	1,9 V

Dari tabel 4.1 dapat disimpulkan bahwa sensor pada penelitian ini efektif bekerja pada rentang jarak $\pm 3-9$ Cm. Dimana, lebih jauh dari 9 cm terjadi penurunan tegangan sensor yang menimbulkan kegagalan pembacaan pendeteksian oleh sensor. Sehingga, jika objek terlalu dekat dengan sensor maka akan terjadi penurunan efektifitas pembacaan oleh sensor. Oleh karena itu agar mendapatkan hasil yang optimal, disarankan untuk menempatkan posisi objek pada jarak yang tepat. Disamping itu, didapatkan juga tegangan rata – rata yang dikeluarkan oleh sensor yaitu sebesar 2,0169 V.

4.1.2 Pengujian ADC

ADC yang digunakan adalah ADC 0804 8-bit. Namun, pada proses pengamatan ADC digunakan ADC inboard 10-bit pada Arduino untuk mempermudah dalam melihat perbedaan antara jarak sensor ke objek dengan output digital yang dihasilkan oleh ADC menggunakan input analog yang diberikan oleh sensor.

Kita dapat mengetahui nilai tegangan analog yang keluar dengan menggunakan persamaan dibawah :

$$V_{out} = \frac{\text{kode digital}}{\text{resolusi}} \times V_{reff}$$

Keterangan:

Kode Digital = Hasil pembacaan ADC (bil.decimal)
 Vout = Tegangan keluaran sensor (V)
 Vreff = Tegangan referensi (V) = 5 V

Resolusi = bit ADC (2^n) = 10-bit (1024) atau 8-bit (256)

Jika keluaran sensor 116 maka :

$$V_{out} = \frac{116}{1024} \times 5$$

$$V_{out} = 0,566 \text{ V}$$

Sehingga didapatkan perbandingan antara Vout ADC yang diukur dengan multimeter dan Vout ADC dengan perhitungan ialah sama, selengkapnya dapat kita lihat pada Tabel II berikut

TABEL II
HASIL PENGUJIAN ADC

Jarak Objek Dengan Sensor (Cm)	Nilai Desimal	Tegangan Output (V)
1	116 – 153	0,56 – 0,74 V
2	229 – 251	1,11 – 1,22 V
3	378 – 430	1,84 – 2,09 V
4	403 – 477	1,96 – 2,32 V
5	457 – 473	2,23 – 2,30 V
6	429 – 473	2,09 – 2,30 V
7	450 – 481	2,19 – 2,34 V
8	251 – 510	1,22 – 2,48 V
9	411 - 442	2,00 – 2,15 V
10	378 – 420	1,84 – 2,04 V
11	339 – 381	1,65 – 1,85 V
12	224 – 337	1,09 – 1,64 V

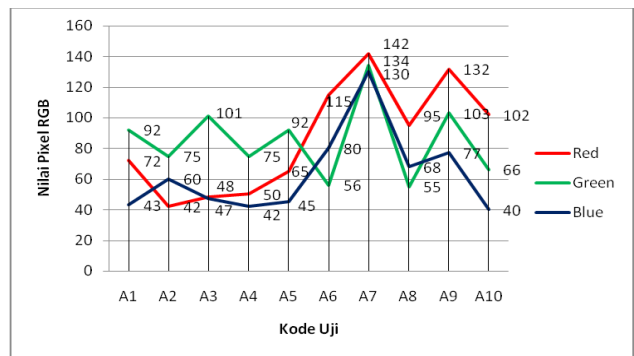
4.2 Analisa Hasil

4.2.1 Data Sampel

Proses pengambilan data sampel tomat matang dan tomat tidak matang adalah untuk menjadikan data sampel tomat tersebut sebagai pembandingan untuk melakukan pengujian buah tomat matang dan tidak matang sebagai data uji. Pengambilan data sampel ini dilakukan pada pencahayaan yang sama. Sampel tomat yang dijadikan sebagai pembandingan didapatkan dari 5 kali capture untuk setiap tingkat kematangan tomat matang dan tidak matang yang dilakukan secara acak. Berikut adalah data sample tingkat kematangan buah tomat yang tersimpan dalam database. Berikut adalah tabel III sampel kematangan buah tomat.

TABEL III
SAMPel KEMATANGAN BUAH TOMAT

Kode	Red	Green	Blue	Kematangan
S1	130	53	43	Matang
S2	84	87	68	Tidak Matang
S3	200	125	100	Matang
S4	100	200	90	Tidak Matang
S5	95	80	70	Matang



Gambar 4.3 Grafik Uji

4.2.2 Proses Pengambilan Data Uji

4.2.2 Data Identifikasi Kematangan Buah Tomat

Berikut pengujian software dilakukan pada pencahayaan yang sama untuk mengidentifikasi matang atau tidak matangnya buah tomat. Salah satu hasil interface dari program identifikasi kematangan buah tomat dapat dilihat pada gambar 4.2



Gambar 4.2 Hasil Identifikasi Tomat Tidak Matang Kode 'A1'

Berikut adalah gambar grafik uji kematangan tomat pada gambar 4.3

Berikut ini juga merupakan hasil data uji kematangan buah tomat yang terdapat pada tabel IV

TABEL IV
DATA UJI KEMATANGAN BUAH TOMAT

Kode	Red	Green	Blue	Kematangan	Hasil
A1	72	92	43	Tidak Matang	Berhasil
A2	42	75	60	Tidak Matang	Berhasil
A3	101	48	47	Matang	Gagal
A4	50	75	42	Tidak Matang	Berhasil
A5	65	92	45	Tidak Matang	Berhasil
A6	115	56	80	Matang	Berhasil
A7	142	134	130	Matang	Berhasil
A8	95	55	68	Matang	Berhasil
A9	132	103	77	Matang	Berhasil
A10	102	66	40	Matang	Berhasil

Pada 10 kali pengujian data diatas, 5 kali pengujian untuk tomat tidak matang dan 5 kali pengujian untuk tomat matang pada pencahayaan yang sama/normal dengan database. Pada tomat matang, hasil yang didapat yaitu sistem mampu mendeteksi kematangan tomat yang matang 100% sedangkan tomat tidak matang yang berhasil teridentifikasi oleh sistem hanya 4 tomat dari 5 tomat yang diidentifikasi, yaitu sebesar 80%.

Persentase keberhasilan pada tomat matang dan tidak matang adalah:

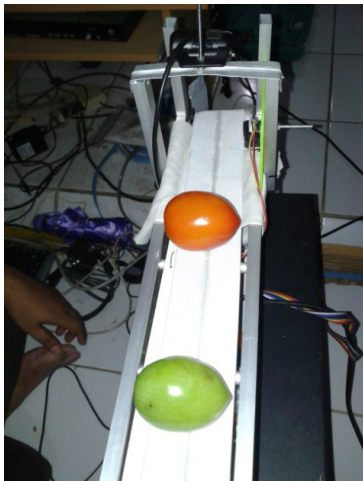
$$\text{Hasil pengujian/total pengujian} \times 100\% = \dots\dots\dots(\text{presentase keberhasilan})$$

$$5/5 \times 100\% = 100\% \text{ (tomat matang)}$$

$$4/5 \times 100\% = 80\% \text{ (tomat tidak matang)}$$

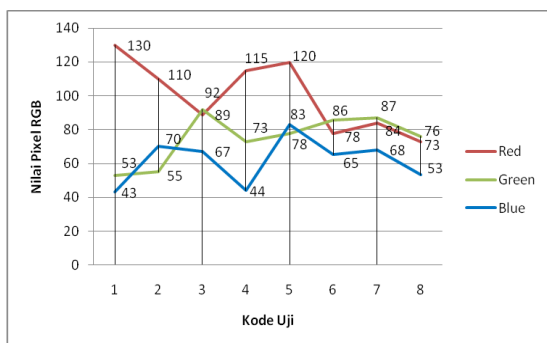
4.2.3 Data Uji Tomat Secara Acak

Pada proses pengambilan data uji secara acak, digunakan 8 sampel buah tomat yang disusun secara acak di atas konveyor dengan keadaan pencahayaan yang cukup atau normal pada pukul 20.30. Berikut pada gambar 4.4 adalah gambar proses pengujian identifikasi kematangan buah tomat untuk kode uji C3.



Gambar 4.4 Proses Uji tomat kode 'C3'

Dari proses pengujian tomat secara acak didapatkan grafik nilai pixel kematangan buah tomat yang terdapat pada gambar 4.5



Gambar 4.5 Grafik Uji Tomat Secara Acak

Dari 8 kali percobaan yang dilakukan dengan menggunakan buah tomat matang dan tidak matang, didapatkan persentase keberhasilan sebesar 100%. Dimana, dari total 8 kali percobaan didapatkan 4 buah tomat yang matang dan 4 buah tomat yang tidak matang yang mana dilakukan secara acak dapat terlihat pada grafik.

$$\text{Hasil pengujian/total pengujian} \times 100\% = \dots\dots\dots(\text{presentase kematangan})$$

$$8/8 \times 100\% = 100\% \text{ (tomat matang dan tidak matang).}$$

Sehingga menghasilkan hasil data uji buah tomat yang dilakukan secara acak yang terdapat pada tabel V

TABEL V
DATA Uji BUAH TOMAT SECARA ACAK

Kode	Red	Green	Blue	Kematangan	Hasil
C1	130	53	43	Matang	Berhasil
	110	55	70	Matang	Berhasil
C2	89	92	67	Tidak Matang	Berhasil
	115	73	44	Matang	Berhasil
C3	120	78	83	Matang	Berhasil
	78	86	65	Tidak Matang	Berhasil
C4	84	87	68	Tidak Matang	Berhasil
	73	76	53	Tidak Matang	Berhasil

V. KESIMPULAN

Dari hasil pengujian yang dilakukan dan analisa tentang pendeteksian kematangan dan ukuran buah tomat, maka dapat ditarik kesimpulan sebagai berikut:

1. Sistem yang dirancang sudah dapat membedakan range RGB nilai tomat matang dan tidak matang untuk tingkat kematangan dan terlihat pada grafik. Range nilai klasifikasi tingkat kematangan ini juga menggunakan histogram.

Tingkat kematangan tomat :

- a. Tomat matang = nilai RGB merah > nilai RGB hijau

- b. Tomat muda = nilai RGB hijau > nilai RGB merah
2. Pendeteksian untuk tingkat kematangan tomat sangat berpengaruh terhadap cahaya dan jarak konstan webcam pada saat peng-capture-an, jika luminasi pencahayaan yang kurang maka gambar yang dicapture akan kurang bagus (banyak noise).
 3. Persentasi keberhasilan pendeteksian kematangan tomat pada sistem yang dibuat ini, hasil yang didapat yaitu sistem mampu mendeteksi kematangan tomat yang matang sebesar 100% dari 10 kali pengujian sampel, sedangkan tomat tidak matang berhasil teridentifikasi sebesar 80% pada 10 kali pengujian sampel pada cahaya normal. Pada pengujian buah tomat secara acak yaitu sistem dapat mendeteksi kematangan tomat matang dan tidak matang sebesar 100 % dari 8 kali pengujian sampel.

UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada Bapak Andrizal, MT dan Bapak Dodon Yendri, M.Kom yang telah membimbing dari awal sampai akhir dalam proses penulisan tugas akhir ini.

REFERENSI

- [1] Tim Bina Karya Tani 2009. *Budidaya Tanaman*. Yrama Widya, Bandung
- [2] Munir, Rinaldi. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritma*. Bandung : Penerbit Informatika
- [3] Sutoyo. 2009. *Citra Digital*. Graha Ilmu, Yogyakarta
- [4] Pantastico EB. 1986. *Fisiologi Pasca Panen Penanganan dan Pemanfaatan Buah buahan dan Sayur-sayuran Tropika dan Sub Tropika*. Kamaryani penerjemah; Yogyakarta: Gajahmada University Press.
- [5] Silanam K, Narongsak S, Sthit S. 2002. *Mango Sorter Machine*. Bangkok, Thailand: Institute for Scientific and Technological Research and Services, King Mongkut's University of Technology Thonburi.
- [6] Prathama YG. 2002. *Desain Sistem Mekanik dan Uji Teknis Mesin Sortasi Mangga Berdasarkan Pengolahan Citra Menggunakan Pengendali Mikrokomputer* [skripsi]. Bogor: Jurusan Teknik Pertanian, Fakultas Teknologi Pertanian, Institut Pertanian Bogor.
- [7] Rice Lake Weighing System. 2001. *Product Information for Sortation/Classifier System*. Canada: RLWS.
- [8] Meng, C.K., Away, Y., Elias, N.F. and Prabuwo, A.S., 2004, *The Real Time Visual Inspection System for Bottling Machine*, Proceeding of the 2nd Conference on Computer Graphics and Multimedia (CoGRAMM'04), pp.
- [9] Anonymous Tanpa tahun. Landasan Teori http://repository.upi.edu/operator/upload/s_kom060762chapter3.pdf (Diakses pada 10 april 2012)
- [10] Anonymous Tanpa tahun. Landasan Teori <http://rapidlibrary.com/jbptunikompp-gdl-untungwalu-19876-4-babii.doc> (Diakses pada 23 maret 2013)
- [11] Anonymous Tanpa tahun. Landasan Teori <http://library.binus.ac.id/eColls/eThesis/Bab2/2007-1-00229-SK-Bab%202.pdf> (Diakses pada 25 maret 2013)
- [12] Triwiyatno, Aris. *Buku Ajar Sistem Kontrol Analog* <http://aristriwiyatno.blog.undip.ac.id> (Diakses pada 17 maret 2013)
- [13] Fadlisyah. 2004. *Computer Vision Dan Pengolahan Citra*. Andi, Yogyakarta
- [14] Thiang Leonardus Indrotanoto. 2008. *Otomatisasi Pemisah Buah Tomat Berdasarkan Ukuran dan Warna Menggunakan Webcam Sebagai Sensor*. *Jurnal Seminar Nasional Ilmu Komputer dan Aplikasinya (SNIKA)* <http://scribd.com>. (Diakses tanggal 24 Februari 2013)
- [15] Pressman, R.S. (2001). *Software Engineering : A Practioners Approach*. Edisi Kelima. McGraw Hill, United States Of America.
- [16] Gonzales, wood 1993. *Citra Digital dan Implementasinya*. Erlangga. Jakarta
- [17] Hariyanto, Didik. 2009. *Teknik Antarmuka ADC*. Staff Site Universitas Negeri Yogyakarta. <http://directlinksearch.com/dir/teknik-antarmuka> (Diakses pada 1 april 2013)
- [18] Yudha, O, D. Purwanto M, dan A. Sardjono. *Aplilkasi Komputer Vision Untuk Identifikasi Kematangan Jeruk Nipis*. *Jurnal Jurusan Teknik Elektro FTI – ITS*. <http://digilib.its.ac.id/ITS-Undergraduate-17648-Paper-525217.pdf>, (Diakses tanggal 18 Februari 2013)
- [19] Kopardekar, P , Mital, A. and Anand, S., 1993, *Manual, Hybrid and Automated Literature and Current Research, Integrated Manufacturing System*, vol. 4, pp. 18-29.
- [20] Adluna, Y. Prasetyawan dan P.D Karningsih. *Perancangan Automated Multi-View Visual Inspectin And Grading System Berbasis Digital Image Processing*. *Jurnal Institut Teknologi Sepuluh Nopember (ITS)*. <http://digilib.its.ac.id/public/ITS-Undergraduate-18388-Paperpdf.pdf>, (Diakses tanggal 25 Februari 2013)
- [21] Parulian P, Lasman. 2006. *Sistem Pengaturan Dengan Komputer . Graha Ilmu, Yogyakarta*.
- [22] Hermanto, Lucky. Tanpa Tahun. *Konsep Komunikasi Paralel*. <http://luckyhermanto.dosen.narotama.ac.id/files/2011/10/konsep-komunikasi-paralel.pdf> (Diakses tanggal 28 Oktober 2013)
- [23] Nalwan, Agustinus, 2004, *Pengolahan Gambar Secara Digital*, PT Elex Media Komputindo, Jakarta.
- [24] <http://saptaji.com/2011/07/20/interfacing-parallel-port-dengan-delphi-bagian-1/>