

NAMA : RIFQI FATHUL AKHYAR
KELAS : 4G D3 TEKNIK KOMPUTER
NIM : 18041175

JURNAL 1

Aplikasi Computer Vision Untuk Penentuan Posisi Objek Simetris Pada Ruang Tiga Dimensi

Penulis : Najira Umar

1. PENGOLAHAN CITRA

Citra (Image) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi penerus dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian cahaya dan ditangkap oleh alat-alat optik seperti mata pada manusia, kamera, pemindai (scanner) dan sebagainya, sehingga bayangan objek yang disebut citra ini terekam (Rinaldi Munir, 2004).

Pengolahan citra adalah langkah yang digunakan untuk memperbaiki citra yang mengalami gangguan agar mudah diinterpretasi baik oleh manusia maupun oleh komputer yang bertujuan memperbaiki kualitas citra menjadi lebih baik (Rinaldi Munir, 2004). Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain, jadi masukannya adalah citra dan keluarannya juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan.

Pengolahan citra (*image processing*) adalah suatu ilmu komputasi yang memungkinkan manusia dapat mengambil informasi dari suatu citra, tidak lepas dari bidang *computer vision*.

Sesuai dengan perkembangannya terdapat dua tujuan utama, yakni :

- (1) Memperbaiki kualitas citra Hasilnya berupa informasi citra yang interprestasikan oleh manusia (*human perception*).
- (2) Mengekstraksi informasi ciri yang menonjol pada suatu citra

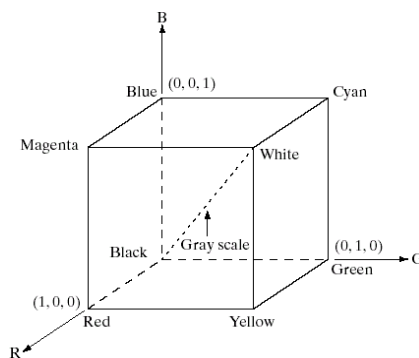
Hasilnya berupa informasi ciri dari citra secara numerik melalui besaran data yang dapat dibedakan secara jelas (Achmad Basuki, 2005).

Pengolahan citra (*image processing*) merupakan suatu proses filter gambar asli menjadi gambar lain sesuai kebutuhan. Misalnya, apabila mendapatkan gambar yang terlalu

gelap, dengan pengolahan citra dapat diproses agar mendapatkan gambar yang jelas seperti yang digambarkan dalam blok diagram (Riyanto Sigit, 2005).

Citra digital merupakan citra yang diambil berdasarkan sampling dan kuantisasi tertentu, terbentuk dari piksel-piksel yang besarnya tergantung pada sampling dan nilai derajat keabuan serta tergantung pada kuantisasi. Model citra digital dinyatakan dalam bentuk matrik, citra didefinisikan sebagai fungsi (x,y) dimana x menyatakan nomer baris dan y menyatakan kolom dan f menyatakan nilai dari derajat keabuan dari citra. Model matrik pada citra digital memungkinkan dilakukannya operasi matrik.

Citra merupakan dimensi spatial yang berisi informasi warna dan tidak bergantung pada waktu. Citra merupakan sekumpulan titik-titik dari gambar, yang disebut *pixel* (*picture element*). Titik-titik tersebut menggambarkan posisi koordinat dan mempunyai intensitas yang dapat dinyatakan dengan bilangan. Intensitas ini menunjukkan warna citra, melalui penjumlahan (*Red, Green dan Blue / RGB*).



Gambar 2. Skema Kubus Warna RGB Koordinat memberikan informasi warna pixel berdasarkan; *Brightness* (ketajaman) warna cahaya (hitam, abu-abu, putih) dari sumber, *Hue* (corak warna) yang ditimbulkan oleh warna (merah, kuning, hijau dll) dan merupakan panjang gelombang dominan dari sumber. Misalnya citra dengan 8 bit per pixel mempunyai 256 warna dan citra dengan 24 bit tiap pixel dinyatakan dengan ;

- bit 0 sampai dengan 7 untuk warna merah (red)
- bit 7 sampai dengan 15 untuk warna hijau (green)
- bit 16 sampai dengan 24 untuk warna biru (blue)

Kemungkinan kombinasi warna yang ada adalah $16.777.216$, dimana nilai 0 menyatakan warna **hitam** sedangkan nilai $16.777.216$ menyatakan warna **putih**.

Hubungan image processing dengan pembagian bidang dalam komputer yang melibatkan input dan output tertentu dapat di jelaskan pada tabel berikut ini :

Tabel .1 Hubungan Pengolahan Citra

		Output	
		Image	Deskripsi
Input	Image	Pengolahan Citra	- Penge nalan pola - Comp uter Vision
		Komputer Grafik	Pengolahan data lainnya

Dalam tabel diatas terlihat jelas bahwa pengolahan citra (image processing) merupakan suatu bidang pengetahuan dimana inputnya berupa citra dan hasilnya juga berupa citra dengan proses yang berupa perbaikan kualitas citra atau penyajian informasi citra. Agar hasil berupa data numerik atau teks yang menyatakan informasi yang ada dalam citra diperlukan pengetahuan yang dipelajari dalam pengenalan pola dan computer vision.

2. DIGITALISASI CITRA

Agar dapat diolah dengan komputer digital, maka suatu citra harus dipresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi malar(kontinu) menjadi nilai-nilai diskrit disebut digitalisasi, citra yang dihasilkan ini disebut dengan citra digital. Pada umumnya citra digital berbentuk persegi panjang yang dimensi ukurannya dinyatakan sebagai tinggi x lebar atau lebar x panjang. Citra digital yang tingginya N, lebarnya M, dan memiliki L derajat keabuan dapat dinyatakan sebagai fungsi :

$$0 \leq x \leq M$$

$$f(x,y) \quad 0 \leq y \leq N$$

$$0 \leq f \leq L$$

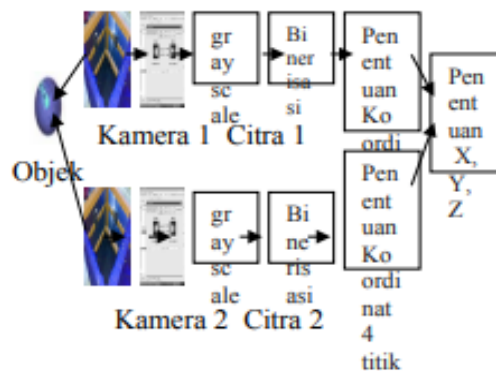
3. SEKENARIO UJI COBA

Dalam melaksanakan penelitian ini akan dilakukan perancangan alat dan sistem untuk melakukan pengambilan gambar yang akan dibuat dirancang dalam blok diagram seperti berikut:



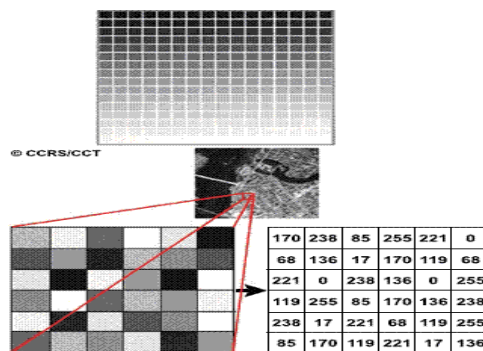
Gambar 3 Blok diagram perangkat keras

Aplikasi pengolahan citra yang dirancang bertujuan untuk menentukan posisi objek yang capture dengan menggunakan webcam dengan memanfaatkan komponen delphi dalam pengambilan gambar dan menampilkan gambar yang disusun dalam blok diagram sebagai berikut



Gambar 4 Disain Perangkat Lunak

- a. Pengambilan gambar dengan webcam menggunakan program aplikasi yang memanfaatkan komponen delphi berupa :
 1. TtsCap32 adalah komponen untuk menampilkan gambar berupa gambar bergerak
 2. TtsCap32PopupMenu adalah komponen untuk mengatur bagaimana pengambilan gambar
 3. TtsCap32Dialogs adalah komponen untuk mengatur format gambar yang akan dicapture.
- b. Mengubah ke citra gray scale dan citra biner Gambar yang telah capture berupa citra warna diolah dengan menggunakan program aplikasi kedalam citra keabuan (gray scale) dengan cara nilai piksel yang ada pada citra yang berupa citra warna dirataratakan kemudian dibagi tiga sesuai dengan jumlah layer pada citra warna yaitu layer r, layer b, dan layer g menjadi satu layer yaitu keabuan (gray scale). Unit terkecil dari data digital adalah bit, yaitu angka biner, 0 atau 1. Kumpulan dari data sejumlah 8 bit data adalah sebuah unit data yang disebut byte, dengan nilai dari 0 – 255 .Pixel (picture element) adalah sebuah titik yang merupakan elemen paling kecil pada citra. Angka numerik (1 byte) dari pixel disebut digital number (DN). Digital Number bisa ditampilkan dalam warna kelabu, berkisar antara putih dan hitam (gray scale), tergantung level energi yang terdeteksi. Pixel yang disusun dalam orde yang benar akan membentuk sebuah citra. Gambar di bawah ini menunjukkan derajat keabuan dan hubungan antara digital number dan derajat keabuan yang menyusun sebuah citra.



Gambar 5 Hubungan Digital Number dengan derajat keabuan

a. Algoritma Sistem

Algoritma Untuk Merancang Aplikasi penentuan posisi objek adalah sebagai berikut :

- 1). Membuat program aplikasi pengambilan gambar
- 2). Mengcapture Gambar dengan menggunakan webcam
- 3). Menampilkan gambar yang telah dicapture dalam bentuk citra diam

- 4). Mengubah Citra warna menjadi grayscale.
- 5). Mengubah Citra Gray Scale ke dalam citra biner
- 6). Menentukan koordinat empat titik. 7). Menentukan posisi X,Y,Z

Fungsi utama sistem ini adalah menentukan posisi objek dalam aplikasi pengolahan citra, prosedurnya adalah mengambil gambar (citra warna) dengan menggunakan webcam, mengubah ke citra keabuan dan citra biner, menentukan koordinat empat titik, menentukan posisi objek .

Proses awal untuk menentukan posisi objek dalam ruang tiga dimensi adalah mengcapture objek yang hasilnya berupa citra warna, kemudian diubah kedalam citra keabuan (gray scale), Prosedur sebagai berikut :

1. Meletakkan objek pada posisi yang diinginkan
2. Mengaktifkan program pengambilan gambar
3. Mengkoneksikan webcam satu dan dua dengan komputer
4. Mengkalibrasi kamera satu dan dua
5. Menampilkan gambar dilayar
6. Mengcapture gambar

Mengubah citra warna kedalam citra grayscale dengan cara menjumlah nilai dari tiga layer yaitu nilai r, nilai g dan nilai b kemudian dibagi tiga sehingga menghasilkan citra grayscale (keabuan) dengan rumus sebagai berikut :

$$s = \frac{r + g + b}{3}$$

Proses ini bertujuan mengelompokkan piksel-piksel objek kedalam wilayah yang mempresentasikan objek yang membedakan objek dengan latar belakang .

Citra yang telah diubah ke grayscale dilanjutkan dengan

binerisasi yang hanya bernilai 0 dan 1, Pada citra biner, batas antara objek dan latarbelakang terlihat jelas. Piksel objek berwarna putih sedang piksel latarbelakang berwarna hitam. Untuk menentukan nilai biner dari citra grayscale yang memiliki derajat keabuan 256 dibagi dua, maka nilai tengahnya adalah 128 sehingga untuk mengubah menjadi citra biner dapat dituliskan sebagai berikut :

Jika nilai keabuan < 128 maka nilainya sama dengan 0

Jika nilai keabuan >= 128 maka nilainya sama dengan 1

Proses mengubah citra warna ke dalam citra gray scale dan citra biner, maka proses

selanjutnya adalah menentukan koordinat empat titik berupa x_1-y_1 , x_2-y_2 , x_3-y_3 , x_4-y_4 , dengan prosedur sebagai berikut :

1. Hasil capture gambar yang berupa citra biner selanjutnya diolah untuk menentukan posisi koordinat titik x_1y_1 dengan cara melacak piksel yang bernilai 1 dimulai dari koordinat (0,0) yang letaknya pada sisi kiri atas dari citra biner, yang dilakukan berulang sampai ditemukan piksel yang bernilai 1 yang pertama, selanjutnya menjadi nilai x_1y_1 . Ketentuan pelacakannya adalah jika koordinat piksel bernilai 0 maka pencarian dilanjutkan sampai ditemukan koordinat piksel yang bernilai 1.
2. Setelah nilai piksel x_1y_1 ditemukan, maka dilakukan pelacakan sampai ditemukan piksel yang bernilai 1 yang merupakan koordinat piksel pada

kolom terdekat dari batas matriks citra, selanjutnya menjadi nilai x_2y_2 . Ketentuan pelacakannya adalah jika koordinat piksel bernilai 0 maka pencarian dilanjutkan sampai ditemukan koordinat piksel yang bernilai 1.

3. Pelacakan dilanjutkan kepada koordinat piksel dari matriks citra biner sampai ditemukan piksel yang nilai 1, dari baris piksel yang terjauh, yang selanjutnya dijadikan nilai x_3y_3 . Ketentuan pelacakannya adalah jika koordinat piksel bernilai 0 maka pencarian dilanjutkan sampai ditemukan koordinat piksel yang bernilai 1.
4. Setelah nilai piksel x_3y_3 ditemukan, dilanjutkan pelacakan sampai ditemukan koordinat piksel yang bernilai 1 yang letaknya pada paling terakhir dari matriks citra, selanjutnya dijadikan sebagai nilai x_4y_4 . Ketentuan pelacakannya adalah jika koordinat piksel bernilai 0 maka pencarian dilanjutkan sampai ditemukan koordinat piksel yang bernilai 1.

Setelah koordinat empat titik didapatkan, maka dilanjutkan dengan menentukan posisi X, Y, Z yang diproses sebagai berikut :

1. Hasil capture

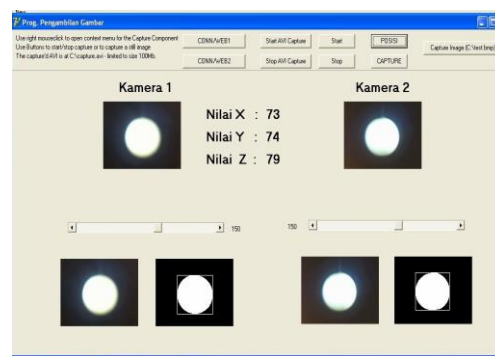
gambar yang ditampilkan pada kamera satu berupa citra biner, dan telah diolah kedalam koordinat empat titik, maka piksel yang titik koordinat x_1 yang pertama ditemukan pada saat pelacakan objek yang nilai piksel 1 selanjutnya dijadikan nilai x , karena merupakan nilai pertama diperoleh yang sejajar dengan sumbu x pada ruang tiga dimensi.

2. Citra biner

yang dihasilkan oleh kamera satu dan kamera dua, dijadikan nilai y dengan proses nilai y1 pada kamera 1 dan nilai y1 pada kamera 2 sama, maka nilai yang diambil untuk dijadikan nilai y dipilih salah satunya dengan cara, nilai piksel y4 dikurangi dengan nilai piksel y1 untuk memperoleh nilai y, karena nilai koordinat tersebut sejajar dengan sumbu y dalam ruang tiga dimensi.

3. Untuk nilai Z

diambil dari hasil capture gambar pada kamera 2 yaitu yang sejajar dengan sumbu Z yaitu nilai x1 dari koordinat x1y1 dari koordinat empat titik .



Gambar 6 Proses Penentuan Posisi Objek

4. KESIMPULAN

1. Tersusun suatu algoritma untuk mengolah obyek gambar digital dari citra warna yang telah dicapture, diolah dengan menggunakan citra gray scale dengan cara citra warna yang terdiri dari 3 layer matriks dengan menjumlahkan nilai RGB kemudian dibagi tiga, dan hasilnya berupa satu layer citra grayscale dengan rentang nilai keabuan 0 sampai dengan 255, dari citra grayscale diubah kedalam bentuk citra biner dimana objek bernilai 1 dan latarbelakang bernilai 0.
2. Telah berhasil merancang program aplikasi pengolahan citra untuk menentukan posisi objek.
3. Kekurangannya belum menghasilkan perbandingan posisi objek yang tepat berdasarkan tingkat resolusi citra, hal ini dipengaruhi oleh ketepatan menempatkan posisi objek pada saat pengcapturan gambar.

JURNAL 2

MENGHITUNG KECEPATAN MENGGUNAKAN COMPUTER VISION

Penulis : Danny Agus Wahyudi & Iman H. Kartowisastro

1. METODE

a. Grayscale

Grayscale adalah proses mengubah derajat gambar yang mulanya RGB menjadi keabu-abuan.

b. Thresholding

Thresholding adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk obyek dan *background* dari citra secara jelas.

c. Reverse Pixel

Mengubah nilai pixel dalam gambar, yang bernilai 1 menjadi 0, dan begitu juga sebaliknya.

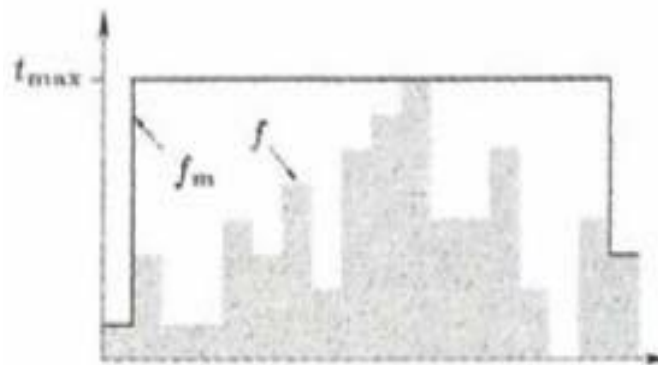
Dengan algoritma: *1-pixel awal = reverse pixel*

d. Removing Pixel

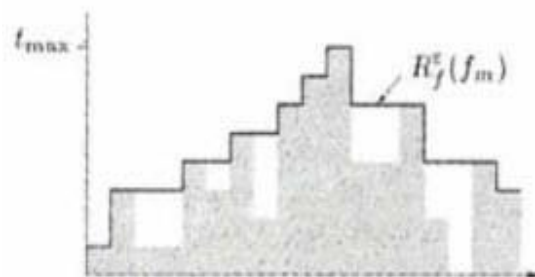
Removing pixel adalah teknik yang digunakan untuk menghilangkan pixel-pixel yang berhubungan dengan jumlah tertentu. Bila ingin menghilangkan pixel dengan jumlah 100, pixel yang berhubungan dengan jumlah 100 akan dihilangkan.

e. Image Filling Region

Teknik ini digunakan untuk mengisi sebuah daerah yang diinginkan dengan menyamakan jumlah pixel-nya. Proses pengisian daerah (*region*) dilakukan oleh sistem dengan cara mengisi daerah yang ditentukan dengan *pixel* yang sudah ditentukan juga, dengan syarat *pixel* yang diisi tidak melewati sisi luar (*edge*) dari objek tersebut, jadi batas pengisian *pixel* hanya sampai sisi luar objek (*edge*). Gambar 1 di bawah ini adalah bagaimana mengisi/mengubah nilai pixel untuk mengisi sebuah daerah, dengan cara menghilangkannya secara ‘paksa’ atau mengubah set nilai pixel (Gambar 2).



Gambar 1. Cara mengisi/mengubah nilai pixel.



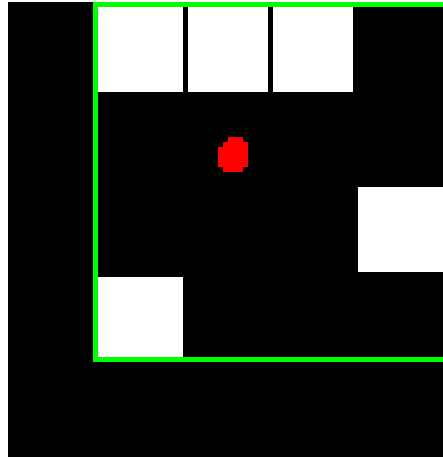
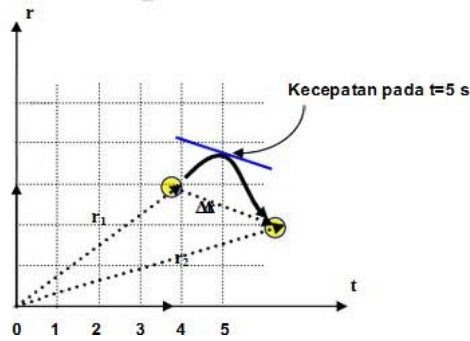
Gambar 2. Cara menghilangkan atau mengubah set nilai pixel.

f. Mencari Centroid

Untuk mendapatkan kecepatan sebuah benda yang bergerak, yang dibutuhkan adalah informasi posisi benda, melalui perubahan posisi benda baru dapat diberikan informasi kecepataannya. Oleh sebab itu, diperlu dicari informasi dari benda tersebut menggunakan algoritma untuk mencari titik tengah dari benda tersebut. Gambar 3 di bawah ini memperlihatkan bagaimana titik tengah adalah pixel merah, titik tengah di cari berdasarkan daerah yang sudah ditentukan. Pada gambar tersebut, daerah yang ditentukan adalah bounding box yang mempunyai tepi berwarna hijau.

g. Speed dan Velocity

Speed adalah perubahan jarak terhadap perubahan waktu, sementara *velocity* adalah perubahan posisi terhadap perubahan waktu. *Speed* menggunakan *distance*, sementara *velocity* menggunakan *displacement*. speed satuannya skalar jadi tidak ada informasi tentang arah di dalamnya, sementara velocity satuannya vektor sehingga mempunyai arah.



Gambar 3. Titik tengah dan *bounding box*.

Karena pada studi ini gambar yang ditangkap dalam bentuk 2D, akan didapatkan informasi dari perubahan posisi tersebut. Untuk menghitung velocity-nya, kita akan

$$\Delta \mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$$

$$\mathbf{v}_r = \frac{\Delta \mathbf{r}}{\Delta t} = \frac{\mathbf{r}_2 - \mathbf{r}_1}{t_2 - t_1}$$

menggunakan rumus berikut:

Di mana r berisi informasi x dan y , dan apabila dinyatakan dalam vektor satuan, maka:

$$\mathbf{v}_r = \frac{\Delta x \mathbf{i} + \Delta y \mathbf{j}}{\Delta t} = \frac{\Delta x}{\Delta t} \mathbf{i} + \frac{\Delta y}{\Delta t} \mathbf{j}$$

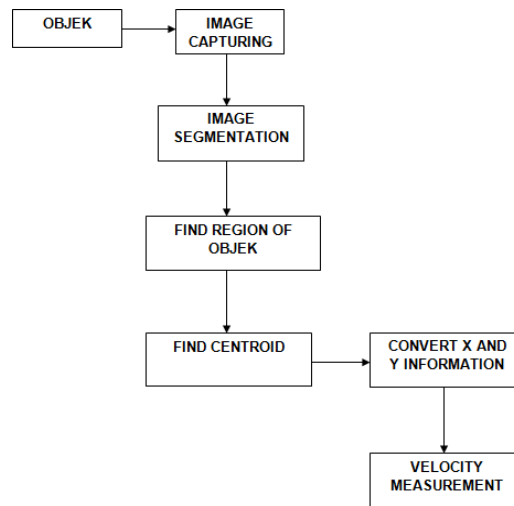
$$\mathbf{v}_r = v_{rx} \mathbf{i} + v_{ry} \mathbf{j}$$

dan besar kecepatan rata-rata dan arah kecepatan rata-rata dapat dihitung dengan rumus

$$|\mathbf{v}_r| = v_r = \sqrt{v_{rx}^2 + v_{ry}^2} \quad \theta = \tan^{-1} \left(\frac{v_{ry}}{v_{rx}} \right)$$

h. Blok Diagram Sistem

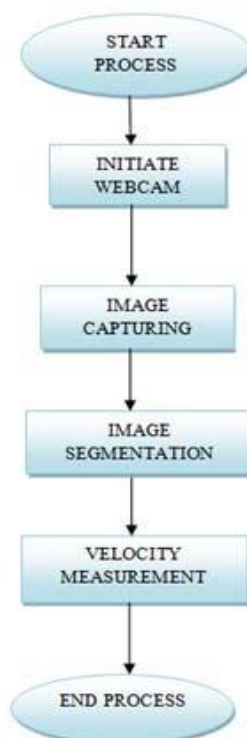
Gambar 4 di bawah ini menunjukkan alur dari sistem secara keseluruhan. Objek yang di tangkap oleh kamera, akan diproses dan sistem akan mengeluarkan informasi dari kecepatan benda tersebut.



Gambar 4. Blok diagram sistem.

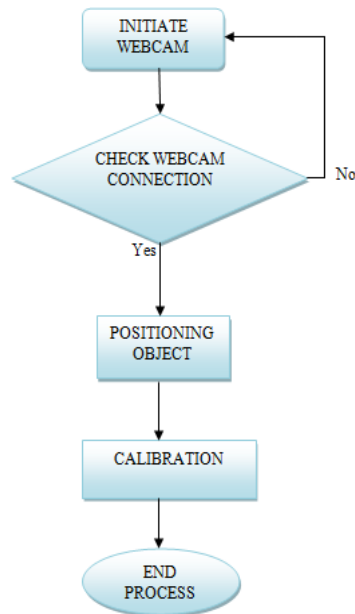
i. Diagram Alir Sistem

Gambar 5 berikut menunjukkan alir sistem secara keseluruhan.



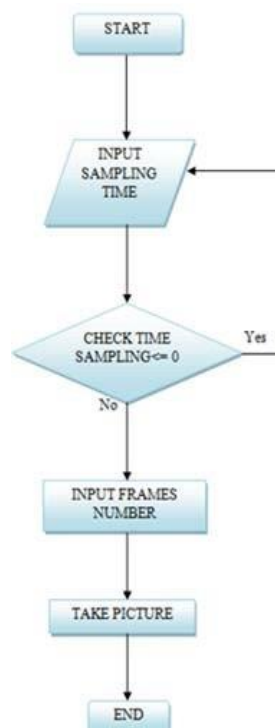
Gambar 5. Diagram alir sistem.

Gambar 6 menunjukkan diagram alir dari inisiasi *webcam* yang di pakai.



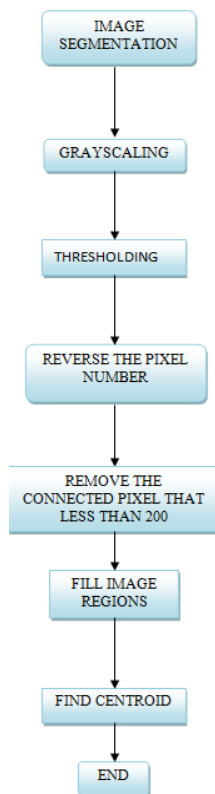
Gambar 6. Diagram alir *initiate webcam*.

Gambar 7 menunjukkan pengambilan gambar yang dilakukan oleh *webcam*. Dengan *interval* waktu yang dimasukkan, sistem akan mengambil gambar selama t second.

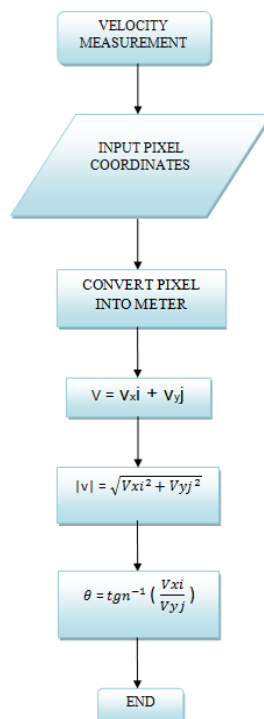


Gambar 7. Diagram alir pengambilan gambar.

Gambar 8 menunjukkan proses segmentasi yang dilakukan oleh sistem, bagaimana sistem memproses image sehingga menghasilkan titik tengah yang nantinya akan di analisis dan menghasilkan informasi kecepatan benda yang di amati.



Gambar 8. Diagram alir segmentasi citra.

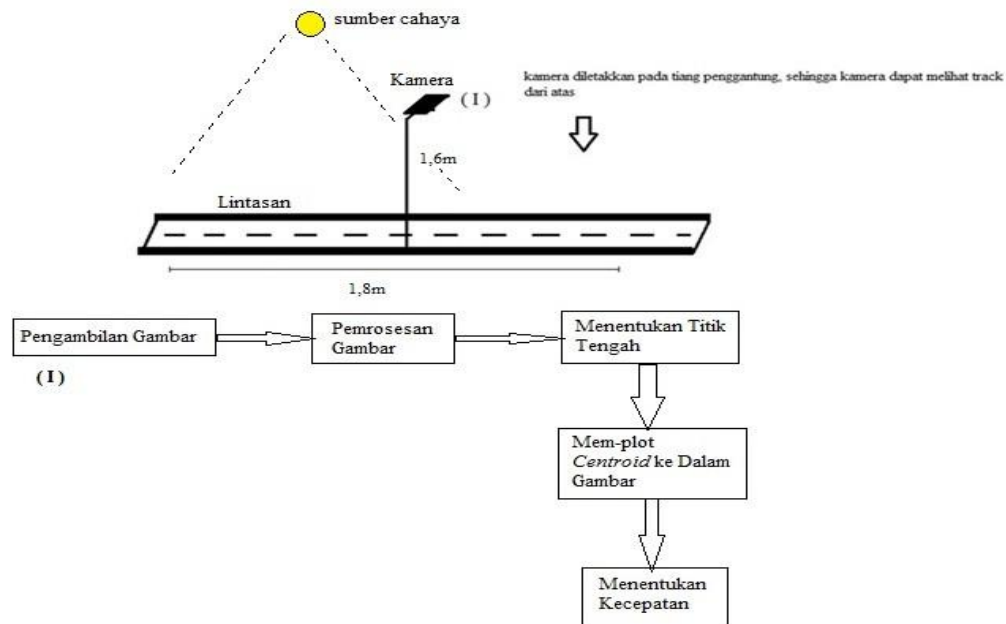


Gambar 9 menunjukkan proses perhitungan kecepatan terhadap informasi yang telah diterima, yaitu titik tengah, dan hasil dari perhitungan akan di tampilkan dan dapat di lihat oleh *user*.

Gambar 9. Diagram alir menghitung kecepatan.

2. Rancang Bangun

Berikut adalah rancang bangun dari sistem secara keseluruhan (Gambar 10).



Gambar 10. Rancang bangun dari sistem secara keseluruhan.

3. Spesifikasi Sistem

Spesifikasi sistem yang digunakan yaitu: webcam Logitech c270, *software* Matlab, *regionproperties* untuk mencari *centroid*, sistem operasi Windows 7 pada PC, Tamiya dengan dimensi (14cm x 8cm), track lintasan dengan panjang 1,8m, *background* putih, lampu kamar sebagai sumber pencahayaan dengan spesifikasi lampu: 23watt (2 buah) dan 18watt (1 buah), penyangga dengan tinggi 2,20m, dan kamar dengan dimensi panjang 3,5m dan lebar 2,5m.

4. Hasil Dan Pembahasan

Implementasi yang dilakukan ditujukan untuk menguji sistem yang telah dirancang, sistem dikatakan stabil apabila sistem dapat mempertahankan output yang stabil dan kinerja sistem yang stabil. Pada perancangan ini sistem diuji dengan pencahayaan yang masuk ke dalam sistem. Berikut adalah hasil percobaan dari sistem (Tabel 1 – 4).

Tabel 1. Hasil Percobaan 1

No.	Lokasi	Pencahayaannya	Sistem – Velocity Measurement		Sistem Manual – (Stopwatch) (m/s)	Kondisi Baterai	Error rate (%)
			Magnitude (m/s)	Sudut Deviasi (°)			
1	Indoor	23 watt	1,09m/s	0,5	1,1m/s	10 – 15 kali pakai	0,9%
2	Indoor	23 watt	1,17m/s	0,8	1,2m/s	10 – 15 kali pakai	2,5%
3	Indoor	23 watt	0,75m/s	0,7	0,74m/s	10 – 20 kali pakai	1,35%
4	Indoor	23 watt	0,66m/s	0,6	0,7m/s	10 – 20 kali pakai	5,7%
5	Indoor	23 watt	0,72m/s	0,6	0,71	10 – 20 kali pakai	1%
6	Indoor	23 watt	0,73m/s	0,6	0,69	10 – 20 kali pakai	4%
7	Indoor	23 watt	0,7m/s	0,6	0,71	10 – 20 kali pakai	1%
8	Indoor	23 watt	0,73m/s	0,5	0,72	10 – 20 kali pakai	1%
9	Indoor	23 watt	0,7m/s	0,5	0,7	10 – 20 kali pakai	0%
10	Indoor	23 watt	0,72m/s	0,6	0,73	10 – 20 kali pakai	1%
Rata – rata			0,797m/s	0,6	0,8m/s		1,935%

Tabel 2. Hasil Percobaan 2

No.	Lokasi	Pencahayaannya	Sistem – Velocity Measurement		Sistem Manual – (Stopwatch) (m/s)	Kondisi Baterai	Error rate (%)
			Magnitude (m/s)	Sudut Deviasi (°)			
1	Indoor	18 watt	2,25m/s	0,3	2,23m/s	1 – 10 kali pakai	0,8%
2	Indoor	18 watt	2,28m/s	0,2	2,3m/s	1 – 10 kali pakai	2%
3	Indoor	18 watt	2,3m/s	0,3	2,3m/s	1 – 10 kali pakai	0%
4	Indoor	18 watt	2,39m/s	0,3	2,36m/s	1 – 10 kali pakai	1,27%
5	Indoor	18 watt	2,33m/s	0,5	2,35m/s	1 – 10 kali pakai	0,8%
6	Indoor	18 watt	2,3m/s	0,4	2,3m/s	1 – 10 kali pakai	0%

7	Indoor	18 watt	2,4m/s	0,1	2,36m/s	1 – 10 kali pakai	1,69%
8	Indoor	18 watt	2,36m/s	0,2	2,3m/s	1 – 10 kali pakai	2,6%
9	Indoor	18 watt	2,29m/s	0,9	2,23m/s	1 – 10 kali pakai	2,6%
10	Indoor	18 watt	2,26m/s	0,5	2,2m/s	1 – 10 kali pakai	2,7%
Rata – rata			2,313m/s	0,37	2,293m/s		1,446%

No.	Lokasi	Pencahayaannya	Sistem – Velocity Measurement		Sistem Manual – (Stopwatch) (m/s)	Kondisi baterai	Error rate (%)
			Magnitude (m/s)	Sudut Deviasi (°)			
1	Indoor	23 watt +18 watt	1,55m/s	0,3	1,56 m/s	10 - 15 kali pakai	0,6%
2	Indoor	23 watt +18 watt	1,5 m/s	0,2	1,49 m/s	10 - 15 kali pakai	0,6%
3	Indoor	23 watt +18 watt	1,6m/s	0,3	1,58 m/s	10 - 15 kali pakai	1,2%
4	Indoor	23 watt +18 watt	1,3m/s	0,3	1,3 m/s	10 - 15 kali pakai	0%
5	Indoor	23 watt +18 watt	1,4m/s	0,4	1,4m/s	10 - 15 kali pakai	0%
6	Indoor	23 watt +18 watt	1,4m/s	0,5	1,42m/s	1 – 10 kali pakai	1,4%
7	Indoor	23 watt +18 watt	2,4m/s	0,6	2,37m/s	1 – 10 kali pakai	1,26%
8	Indoor	23 watt +18 watt	2,36m/s	0,7	2,35m/s	1 – 10 kali pakai	0,42%
9	Indoor	23 watt +18 watt	2,3m/s	0,7	2,3m/s	1 – 10 kali pakai	0%
10	Indoor	23 watt +18 watt	2,4m/s	0,7	2,34m/s	1 – 10 kali pakai	2,5%
Rata – rata			1,821m/s	0,47	1,81m/s		0,858%

Tabel 3. Hasil Percobaan 3

Tabel 4. Hasil Percobaan 4

No.	Lokasi	Pencahayaannya	Sistem – Velocity Measurement		Sistem Manual – (Stopwatch) (m/s)	Kondisi Baterai	Error rate (%)
			Magnitude (m/s)	Sudut Deviasi (°)			
1	Indoor	23 watt + 23 watt + 18 watt	1,63m/s	0,3	1,62 m/s	10 - 15 kali pakai	0,6%
2	Indoor	23 watt + 23 watt + 18 watt	1,6m/s	0,2	1,6 m/s	10 - 15 kali pakai	0%
3	Indoor	23 watt + 23 watt + 18 watt	1,6m/s	0,3	1,59 m/s	10 - 15 kali pakai	0,6%
4	Indoor	23 watt + 23 watt + 18 watt	1,62m/s	0,3	1,64 m/s	10 - 15 kali pakai	1,2%

5	Indoor	23 watt + 23 watt + 18 watt	1,45m/s	0,4	1,48m/s	10 - 15 kali pakai	2%
6	Indoor	23 watt + 23 watt + 18 watt	1,1m/s	0,6	1,15m/s	15 - 20 kali pakai	4,34%
7	Indoor	23 watt + 23 watt + 18 watt	1m/s	0,5	1m/s	15 - 20 kali pakai	0%
8	Indoor	23 watt + 23 watt + 18 watt	1,2m/s	0,6	1,18m/s	15 - 20 kali pakai	1,69%
9	Indoor	23 watt + 23 watt + 18 watt	1,14m/s	0,4	1,2m/s	15 - 20 kali pakai	5%
10	Indoor	23 watt + 23 watt + 18 watt	1,2m/s	0,3	1,2m/s	15 - 20 kali pakai	0%
Rata – rata			1,354m/s	0.39	1.366m/s		1,543%

Pada percobaan di ruang yang tidak terdapat cahaya, sistem tidak dapat mengambil gambar apapun. Pada keadaan ini sistem dianggap tidak dapat bekerja secara maksimal.

Terdapat beberapa poin yang didapatkan pada implementasi ini, yaitu:

- (1) perhitungan dari sistem dibandingkan dengan perhitungan sistem lain agar dapat membuktikan kestabilan sistem;
- (2) pencahayaan tidak merata/homogen mengakibatkan adanya *noise*;
- (3) pada ruangan yang gelap sistem tidak dapat bekerja karena tidak ada cahaya.

5. KESIMPULAN

Pencahayaan yang tidak merata menyebabkan *noise*, dan tingkat pencahayaan yang tinggi ataupun rendah juga menyebabkan *noise*. *Noise* yang muncul akibat pencahayaan dapat dihilangkan dengan algoritma *remove pixel*. Akan tetapi, saat *noise* memiliki jumlah yang lebih besar dari obyek, sistem tidak dapat mengatasinya lagi, contohnya pada saat kamar gelap. Gambar tidak lagi dapat diproses karena tidak ada obyek apapun yang tertangkap.

**PENGEMBANGAN COMPUTER VISION SYSTEM SEDERHANA UNTUK
MENENTUKAN KUALITAS TOMAT**

Penulis :

Rudiati Evi Masithoh , Budi Rahardjo , Lilik Sutiarso , Agus Hardjoko

A. ISI

Kualitas merupakan karakter suatu produk yang menunjukkan derajat keunggulan suatu produk atau kemampuannya dalam memenuhi kebutuhan tertentu (Abbott, 1999). Warna merupakan parameter utama dari konsumen untuk menentukan kualitas (Francis, 1995). Instrumen pengukur warna semakin berkembang, sehingga pengamatan yang sebelumnya hanya dapat dilakukan oleh indra manusia sekarang dapat dilakukan dengan instrumen. Instrumen pengukur warna konvensional, misalnya colorimeters dan spektrofotometer, menyediakan informasi warna dalam format XYZ, RGB dan $L^*a^*b^*$ (Mendoza dkk., 2006). Beberapa metode subyektif untuk mengukur warna telah dikembangkan antara lain color chart dan rating scale (Grierson dan Kader, 1986). CVS merupakan sistem berbasis pada pengembangan algoritma dan komputer untuk menstimulasi visualisasi manusia agar dapat secara otomatis mengekstrak informasi berharga dari suatu obyek (Panigrahi dan Gunasekaran, 2001).

Mata manusia adalah sebuah sistem canggih yang melakukan respon atas rangsangan visual. Secara fungsional, computer vision dan penglihatan manusia adalah sama, dengan tujuan menafsirkan data spasial yaitu data yang diindeks oleh lebih dari satu dimensi. Meskipun demikian, computer vision tidak dapat diharapkan untuk mereplikasi persis seperti mata manusia. Hal ini disebabkan pengetahuan tentang bagaimana sistem mata dan otak bekerja belum sepenuhnya dipahami, sehingga tidak dapat merancang sebuah sistem untuk mereplikasi secara tepat fungsi mata manusia. Yang dapat dilakukan adalah teknik computer vision yang dapat mereplikasi dan, dalam beberapa kasus bahkan memperbaiki sistem penglihatan manusia (Nixon dan Aguado, 2008). CVS terdiri dari komputer yang merupakan analog dari otak manusia, sensor atau kamera digital (analog dari mata manusia), dan sistem penerangan berupa sumber cahaya untuk memfasilitasi akuisisi citra. Secara umum, terdapat 3 (tiga) operasi utama dalam CVS yaitu akuisisi citra (image acquisition), pengolahan citra (image processing), dan penerjemahan citra (image understanding) (Panigrahi dan Gunasekaran, 2001). Secara lebih rinci, tahapan-tahapan dalam analisis citra adalah: 1) pembentukan citra, dimana citra suatu obyek diambil dan disimpan dalam komputer, 2) pre-proses citra, dimana kualitas citra digital diperbaiki untuk meningkatkan detail citra, 3) segmentasi citra, dimana citra obyek diidentifikasi dan dipisahkan dari latar belakangnya, 4) pengukuran citra, dimana beberapa fitur-fitur yang signifikan dikuantitaskan, dan 5) interpretasi citra, dimana fitur-fitur yang telah diekstrak kemudian diinterpretasikan (Mery dan Pedreschi, 2005). CVS membantu kerja manusia dalam mengambil keputusan secara cepat, akurat, dan konsisten. Namun

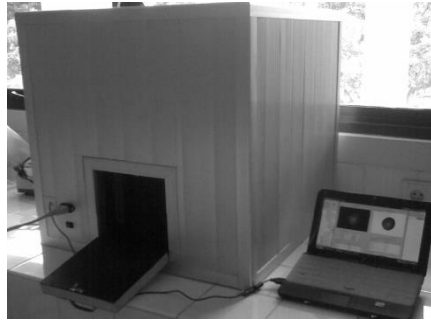
demikian, kinerja CVS memerlukan kajian dan pengembangan yang berkelanjutan karena keterbatasannya untuk melakukan analisis kuantitatif secara mendalam (Abdullah, dkk., 2004). Hal ini disebabkan karena kamera yang digunakan pada CVS biasanya hanya beroperasi pada panjang gelombang 10^{-4} – 10^{-7} m yang hanya dapat menghasilkan informasi parameter eksternal dari obyek dan bukan parameter internal. Tetapi, beberapa penelitian telah dilakukan untuk mengetahui hubungan antara kenampakan eksternal yang diperoleh dari CVS dengan parameter internalnya, seperti kandungan kadar air (Masithoh dan Kusuma, 2008), klorofil (Kawashima dan Nakatani, 1998), serta kadar gula dan pH pada jeruk (Kondo dkk., 2000). Dengan mengembangkan metode CVS ini maka pengukuran dapat dilakukan secara berulang untuk memperoleh hasil yang lebih akurat pada sampel yang sama karena menggunakan metode non-destruktif, serta dapat digunakan di lapangan untuk mengukur kualitas secara cepat. Penelitian ini bertujuan untuk mengukur RGB warna dan memprediksi kualitas buah secara non-destruktif berdasarkan warna untuk mendapatkan informasi hayati secara real-time dengan menggunakan tomat sebagai obyek validasi.

METODE PENELITIAN

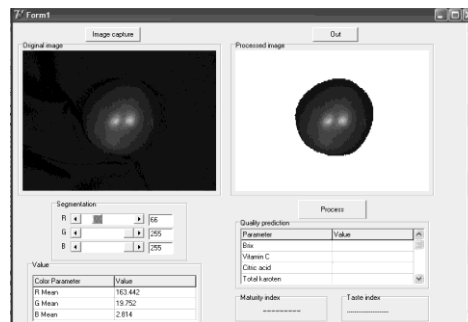
Penelitian ini mencakup dua tahap, yaitu pembuatan instrumen penangkap citra dan pembuatan perangkat lunak untuk menentukan kualitas tomat berdasarkan parameter citra dan jaringan syaraf tiruan. CVS yang dibangun terdiri dari (1) kotak untuk meletakkan obyek dan menangkap citra, (2) webcam untuk mendapatkan gambar dalam format digital, (3) sumber cahaya yang berasal dari lampu Light Emitting Diode (LED) sepanjang 5 meter dan 4 (empat) buah lampu TL masing-masing berdaya 8W, (4) komputer untuk menyimpan gambar dan mengolah data, berikut monitor untuk menampilkan citra, dan (5) analisis citra yang dilengkapi dengan jaringan syaraf tiruan (JST) untuk menentukan kualitas tomat.

Hasil dan Penelitian

CVS yang dikembangkan tampak seperti Gambar 1. CVS terdiri dari kotak aluminium tertutup yang di dalamnya terdapat webcam, lampu LED dan 4 buah lampu TL sebagai sumber cahaya, serta komputer sebagai pengolah citra dan penampil data hasil ekstraksi citra. Citra yang diambil kemudian diolah dengan program pengolah citra yang dibuat. Gambar 2 merupakan tampilan antar muka atau graphical user interface (GUI) dari perangkat lunak pengolah citra yang dibuat. Bagian-bagian dari GUI ini adalah citra awal dan citra hasil segmentasi, nilai rata-rata warna Red Green Blue (RGB) dari citra yang telah disegmentasi, dan prediksi nilai kualitas tomat yang meliputi Brix, vitamin C, asam sitrat, dan gula total. Bagian-bagian dari GUI ini adalah citra awal dan citra hasil segmentasi, nilai rata-rata warna Red Green Blue (RGB) dari citra yang telah disegmentasi, dan prediksi nilai kualitas tomat yang meliputi Brix, vitamin C, asam sitrat, dan gula total.



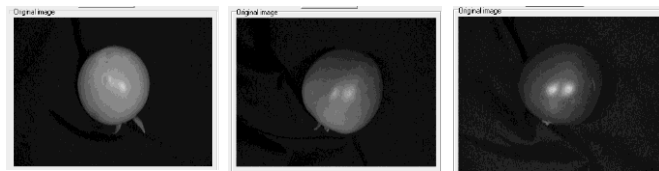
Gambar 1. *Computer vision system yang dikembangkan*



Gambar 2. Tampilan antar muka untuk prediksi kualitas tomat

Warna dan Kualitas Tomat

Gambar 5 menunjukkan nilai warna RGB tomat pada berbagai kelas kematangan. Tampak bahwa pada kelas kematangan yang berbeda maka nilai yang dihasilkan juga berbeda. Dari semua citra tomat yang dianalisis, bila ditampilkan dalam bentuk grafik maka dapat dilihat pada Gambar 6. Tampak bahwa nilai R akan semakin besar jika kelas kematangan tomat semakin meningkat. Jika kelas kematangan meningkat yang ditandai dengan semakin meningkatnya warna merah pada buah, maka nilai R juga meningkat. Nilai R merupakan representasi nilai pigmen tomat berupa karoten dan likopen yang mengalami sintesis selama pemasakan, sehingga warnanya akan meningkat (Grierson dan Kader, 1986). Nilai G tomat akan semakin menurun dengan meningkatnya kelas kematangan (Gambar 6.). Hal ini terjadi karena nilai G mencerminkan klorofil warna hijau pada tomat yang mengalami degradasi selama pemasakan berlangsung. Sedangkan untuk parameter warna B, maka semakin besar kelas kematangan, maka nilai B akan menurun, dengan penurunan yang besar pada kelas kematangan Green sampai Light red, kemudian akan mendekati konstan sampai dengan akhir pemasakan (kelas Red). Hal ini sejalan dengan yang ditemukan oleh Gautier dkk. (2008) yaitu nilai Blue akan semakin menurun mengarah ke warna Yellow. Akan tetapi, fenomena yang berbeda ditemukan oleh Lana dkk. (2006) yang meneliti bahwa nilai B pada tomat iris akan naik.

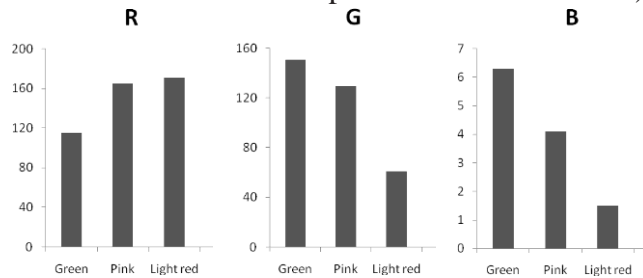


Color Parameter	Value
R Mean	105.512
G Mean	133.733
B Mean	3.395

Color Parameter	Value
R Mean	173.007
G Mean	62.026
B Mean	1.472

Color Parameter	Value
R Mean	196.463
G Mean	29.842
B Mean	4.175

Gambar 5. Nilai RGB tomat pada kelas kematangan a) Green, b) Light red, dan c) Red.



Gambar 6. Nilai R, G, dan B tomat pada berbagai kelas kematangan

Penentuan Kualitas Tomat Menggunakan CVS

Untuk menentukan akurasi program dalam menentukan parameter kualitas, maka perlu dilakukan uji akurasi, dengan membandingkan antara nilai yang diperoleh dari program yang dibuat dengan nilai parameter kualitas dari pengukuran secara destruktif (kimia) di laboratorium. Gambar 8 merupakan grafik hubungan antara nilai Brix, vitamin C, asam sitrat, dan gula total prediksi dan aktual. Agar diperoleh nilai prediksi yang sama atau mendekati nilai aktual maka dibuatkan nilai kalibrasi. Untuk nilai Brix, maka nilai aktualnya diperoleh dari persamaan $y = 12,16x - 26,46$ dengan x adalah nilai Brix prediksi. Sedangkan kadar vitamin C, asam sitrat, dan gula total aktual secara berturut-turut diperoleh dari $y = 1,09x - 3,13$, $y = 7,35x - 19,44$, dan $y = 1,58x - 0,18$, dengan x adalah nilai vitamin C prediksi, asam sitrat prediksi, dan gula total prediksi.

B. KESIMPULAN

Sebuah *Computer Vision System* (CVS) yang terdiri dari kotak obyek, webcam, komputer, sistem penerangan, dan perangkat lunak analisis citra dengan jaringan syaraf tiruan dikembangkan untuk menentukan kualitas tomat, yaitu Brix, asam sitrat, vitamin C, dan gula total. Arsitektur jaringan dibentuk dengan 3 lapisan terdiri dari 1 lapisan input dengan 3 sel syaraf masukan, 1 lapisan tersembunyi dengan 14 sel syaraf berfungsi aktivasi logsig dan 5 lapisan output dengan fungsi aktivasi purelin menggunakan algoritma pelatihan *backpropagation*.