

RANCANG BANGUN APLIKASI PENDETEKSI BENTUK DAN WARNA BENDA PADA MOBILE ROBOT BERBASIS WEBCAM

¹⁾Yustinus Pancasila Prayitno ²⁾Harianto ³⁾Madha Christian Wibowo

1) S1 / Jurusan Sistem Komputer, STIKOM Surabaya. Email: justinlyonhart@gmail.com

2) S1 / Jurusan Sistem Komputer, STIKOM Surabaya. Email: hari@stikom.edu

3) S1 / Jurusan Sistem Komputer, STIKOM Surabaya. Email: madha@stikom.edu

Abstract

Robotino is an omni-directional drive robot created by Festo Didactic and has integrated a variety of sensors and a webcam in it. By utilizing Computer Vision, Robotino can be used as a ball-detection robot based on shape and color of objects. Commonly, RGB color space are used while the process of color segmentation, but the RGB color space still unable to tolerate the intensity of light.

In this project has been build an application using the HSV color space for color filtering process which is able to tolerate the intensity of light in image processing and the methods for detect a circle is using Circular Hough Transform. Moreover, in this study the image data obtained directly via webcam on Robotino streamingly.

This application has been successfully detected the colored balls with the parameters specified filter color in HSV color space and the Circular Hough Transform method, as well as the processed image data obtained through wireless,streamingly from Robotino's webcam which is already integrated on the PC.

Keywords: *Robotino, HSV Color Filtering, Circular hough transform, Streaming*

1. PENDAHULUAN

Perkembangan teknologi yang begitu pesat membuat kebutuhan akan teknologi semakin bertambah. Berbagai macam cara dilakukan untuk mempermudah manusia dalam melakukan suatu pekerjaan, salah satunya yaitu dengan memanfaatkan kecerdasan buatan menggunakan *Computer Vision* kemudian diimplementasikan pada robot. Sebagai contoh *mobile robot, robotic soccer, robotic hokkey*, dan *automated surveillance* juga menggunakan pengolahan citra dengan memanfaatkan *Computer Vision* untuk mencapai hasil yang diinginkan.

Beberapa perusahaan besar di bidang otomasi industri seperti Festo Didactic telah menciptakan robot *omni-directional drive* yang dapat bergerak 360 derajat dan memiliki berbagai sensor yang dapat diprogram sesuai kebutuhan yang diberi nama Robotino. Robot ini terintegrasi dengan sebuah *webcam* dengan *interface* USB yang dapat mendeteksi garis yang biasa digunakan sebagai *line tracer*.

Pemanfaatan *webcam* ini dapat dikembangkan sehingga dapat digunakan sebagai sensor dalam mendeteksi sebuah benda melalui beberapa tahap pengolahan citra. *Webcam* dimanfaatkan sebagai indra penglihatan dalam robot yang dapat digunakan untuk menangkap gambar dan kemudian gambar tersebut akan diolah untuk menentukan objek yang dideteksi.

Penelitian sebelumnya yang sudah dilakukan oleh Gentang Syabba Nahla pada tahun 2011 telah memanfaatkan teknologi *Computer Vision* pada Robotino untuk mendeteksi bola. Namun penelitian tersebut yang menggunakan metode *color filtering* RGB untuk segmentasi warnanya masih belum

mampu mendeteksi bola secara akurat dan tingkat keberhasilannya hanya 66,67% dikarenakan faktor intensitas cahaya masih belum diperhitungkan.

RGB merupakan ruang warna yang langsung mengarah pada warna sesuai dengan ketiga parameter *red*, *green* dan *blue* tanpa memperhitungkan faktor hitam dan putih suatu warna. Namun pada ruang warna HSV, parameter *hue* berperan penting untuk menentukan warna, *saturation* untuk derajat keabuan suatu warna, serta *value* untuk intensitas kecerahan suatu warna.

Oleh sebab itu penulis mengangkat judul ini untuk membuat aplikasi yang lebih akurat dalam mendeteksi bola berwarna dengan menggunakan ruang warna HSV untuk segmentasi warna yang mampu menoleransi perubahan intensitas cahaya dalam mendeteksi warna benda dan juga metode *Circular Hough Transform* untuk mendeteksi benda dengan bentuk lingkaran.

2. LANDASAN TEORI

2.1 *Omni-Directional Robot*

Robotino adalah robot buatan Festo Didactic yang digunakan untuk edukasi dan penelitian serta kompetisi robot. Robotino memiliki fitur sistem gerak menggunakan *omni-directional drive*, *bumps sensors*, *infrared distance sensors*, dan *usb webcam*. Robotino didesain modular, sehingga dapat dengan mudah ditambahkan berbagai aksesoris pelengkap, seperti *sensor laser scanner*, *gyroscope*, dan *positioning system Northstar* dalam ruangan. (ROS, 2010).



Gambar 1. Robotino (ROS, 2010).

2.2 Citra Digital

Citra digital adalah gambar dua dimensi yang dapat ditampilkan pada layar monitor komputer sebagai himpunan berhingga (diskrit) nilai digital yang disebut pixel (*picture elements*). Pixel adalah elemen citra yang memiliki nilai yang menunjukkan intensitas warna (G.W. Awcock, 1996).

2.3 Pengolahan Citra

Pengolahan citra merupakan teknik manipulasi citra secara digital yang khususnya menggunakan komputer, menjadi citra lain yang sesuai untuk digunakan dalam aplikasi tertentu. Agar mudah diinterpretasi oleh manusia atau komputer, pengolahan citra harus dilakukan dengan berbagai macam metode untuk mencapai citra sesuai yang diinginkan.

Operasi pengolahan citra digital umumnya dilakukan dengan tujuan memperbaiki kualitas suatu gambar sehingga dapat dengan mudah diinterpretasikan oleh mata manusia dan untuk mengolah informasi yang ada pada suatu gambar untuk kebutuhan identifikasi objek secara otomatis. (Murinto, 2009)

A. Thresholding

Thresholding merupakan konversi citra berwarna ke citra biner yang dilakukan dengan cara mengelompokkan nilai derajat keabuan setiap *pixel* kedalam 2 kelas, hitam dan putih. Pada citra hitam putih terdapat 256 level, artinya mempunyai skala “0” sampai “255” atau [0,255], dalam hal ini nilai intensitas 0 menyatakan hitam, dan nilai intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih.

Contoh operasi titik berdasarkan intensitas adalah operasi pengambangan (*thresholding*). Pada operasi pengambangan, nilai intensitas *pixel* dipetakan ke salah satu dari dua nilai, α_1 atau α_2 , berdasarkan nilai ambang (*threshold*) T dapat ditunjukkan seperti pada persamaan berikut: (Munir, 2004)

$$f_{x,y}' = \alpha_1, \quad f_{x,y} < T \alpha_2, \quad f_{x,y} \geq T$$

B. Color Filtering

Color Filtering adalah suatu teknik pengolahan citra yang yang dipakai untuk memanipulasi suatu citra berdasarkan warna spesifik. Cara kerjanya adalah dengan membandingkan komponen warna setiap *pixel* citra dengan warna spesifik. Apabila warnanya sesuai dengan warna spesifik komponen warna *pixel* tersebut dibiarkan saja. Namun, bila warnanya tidak sesuai dengan warna spesifik maka komponen warna *pixel* tersebut diubah menjadi warna background, biasanya menjadi warna hitam.

Warna yang digunakan dalam *Color Filtering* dapat direpresentasikan dalam berbagai ruang warna. Ada beberapa ruang warna yang dikenal, antara lain RGB (*Red, Green, Blue*), HSV (*Hue, Saturation, Value*), YCbCr, dsb. HSV merupakan ruang warna yang sangat cocok untuk mengidentifikasi warna-warna dasar, dimana warna dasar ini digunakan dalam penelitian sebagai warna identifikasi robot. Selain itu, HSV menoleransi terhadap perubahan intensitas cahaya. Inilah yang menjadi keunggulan HSV dibandingkan dengan ruang warna lainnya. (Dhiemas, 2011)

C. Hough Transform

Transformasi Hough adalah sebuah metode yang dapat digunakan untuk mengisolasi feature tertentu dalam sebuah citra. Metode Hough Transform biasanya digunakan untuk mendeteksi bentuk geometri yang dapat dispesifikasikan dalam bentuk parametrik seperti garis, lingkaran, elips, dan lain-lain. Prinsip kerja metode Transformasi Hough dalam mendeteksi garis adalah dengan mencari bentuk geometri yang paling sesuai dengan kumpulan titik pada citra. Untuk bentuk lingkaran, persamaan parametriknya adalah :

$$(x-a)^2 + (y-b)^2 = r$$

Dengan a dan b adalah koordinat pusat dari lingkaran dan r adalah radiusnya. Dalam kasus ini kompleksitas komputasi algoritma ini akan meningkat, karena jumlah parameter koordinat dan akumulator berdimensi 3. Secara umum komputasi dan ukuran deret akumulator meningkat secara polinomial dengan jumlah parameternya seperti rumus berikut:

$$x = a + r \cos(\theta)$$

$$y = b + r \sin(\theta)$$

Circular hough transform dapat ditentukan dengan parameter dari sebuah lingkaran, dimana nilai dari parameter yang digunakan sudah ditentukan terlebih dahulu. Sebuah lingkaran dengan radius r dan titik tengah (a,b) sesuai dengan rumus diatas. Ketika sudut pada θ bernilai 0-360derajat, parameter (x,y) akan langsung mengikuti perubahan. (Tharom,2003)

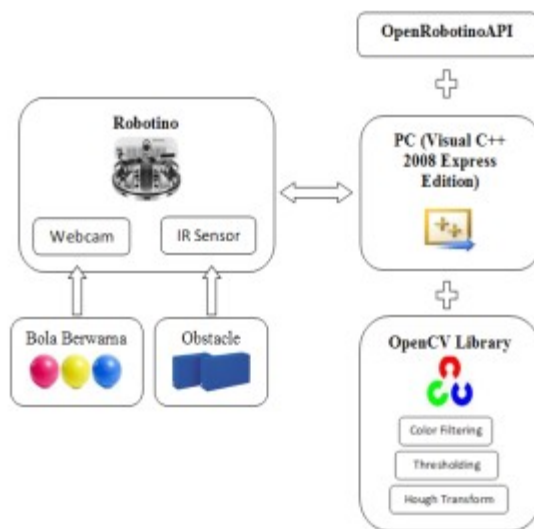
2.4 OpenCV

OpenCV (*Open Computer Vision*) adalah sebuah API (*Application Programming Interface*) library yang sudah sangat familiar pada pengolahan citra menggunakan *Computer Vision*. *Computer Vision* itu sendiri adalah salah satu cabang dari bidang ilmu pengolahan citra (*Image Processing*) yang memungkinkan komputer dapat melihat seperti manusia. Dengan *vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu objek. Beberapa pengimplementasiandari *Computer Vision* adalah *Face Recognition*, *Face Detection*, *Face/Object Tracking*, *Road Tracking*, dll.

OpenCV adalah *library Open Source* untuk *Computer Vision* untuk C/C++, OpenCV didesain untuk aplikasi *real-time*, memiliki fungsi-fungsi akuisisi yang baik untuk *image/video*.(Syafi'i,2011)

3. METODE

3.1 Blok Diagram Sistem



Gambar 2. Blok Diagram Sistem

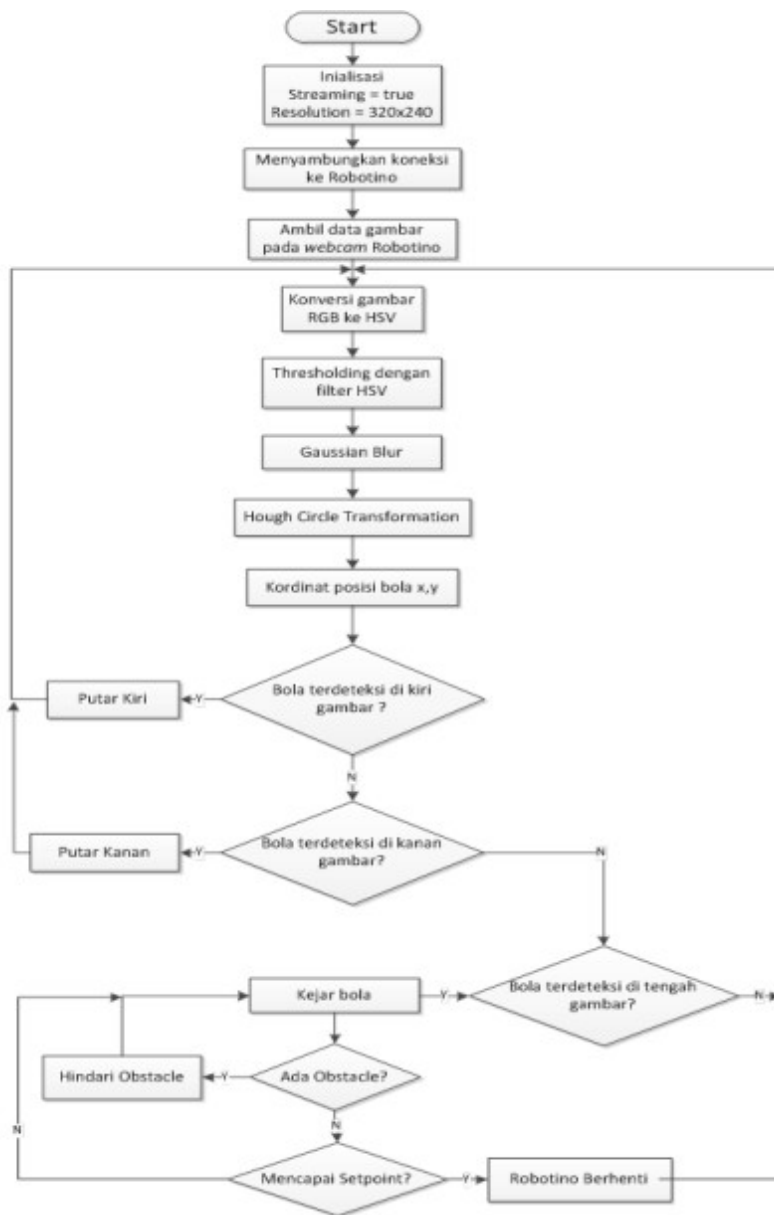
Pada blok diagram tersebut dapat dijelaskan bahwa Robotino akan mendeteksi benda melalui *webcam*, kemudian data gambar pada *webcam* akan dikirimkan melalui media *wireless* secara *streaming* menggunakan *console application* Visual C++ 2008 yang sudah terkoneksi dengan PC menggunakan *library* OpenRobotinoAPI. Data gambar tersebut akan diolah dengan memanfaatkan *library* OpenCV, yang dalam proses pengolahan citra melalui beberapa tahapan yaitu pertama melakukan proses *color filtering* HSV untuk mendeteksi warna benda, kemudian proses *thresholding* untuk menciptakan citra yang dideteksi menjadi citra biner dan *Circular Hough Transform* digunakan untuk mendeteksi benda dengan bentuk lingkaran. Setelah proses pengolahan citra dan bola telah terdeteksi, kemudian Robotino akan bergerak maju mendekati bola tersebut. Dan memeriksa data dari *infrared distance sensor* apakah terdapat halangan, jika iya maka Robotino akan menghindari halangan tersebut.

3.2 Perancangan Perangkat Lunak

Dalam perancangan perangkat lunak, *compiler* yang digunakan adalah Microsoft Visual C++ 2008. Untuk *library* yang digunakan pada pengolahan citra yaitu *library* OpenCV v2.1 dan *library* OpenRobotinoAPI digunakan untuk mengintegrasikan Robotino dengan PC, sehingga seluruh modul-modul dan sensor didalamnya dapat diakses.

1. Algoritma Program

Untuk algoritma program yang digunakan akan dijelaskan melalui *flowchart* berikut:



Gambar 3. Flowchart Aplikasi

2. Kendali Robot

Untuk kendali Robotino digunakan OpenRobotinoAPI yaitu *library* aplikasi *programming* yang dibuat khusus untuk Robotino. *Library* ini memungkinkan akses penuh terhadap sensor dan *actors* pada Robotino. Tahap penelitian untuk kendali robot dimulai dari menyambungkan koneksi dari PC ke Robotino melalui media *wireless*, kemudian mengatur pergerakan dari Robotino, mengakses *infrared distance sensor* pada Robotino, dan penerimaan data gambar secara *streaming*.

3. Pengolahan Citra

Pada tahap pengolahan citra dilakukan penelitian tentang cara konversi dari ruang warna RGB ke ruang warna HSV, kemudian hasil dari konversi citra digunakan untuk proses *color filtering* gambar dan diubah menjadi citra biner melalui proses *thresholding*, setelah mendapatkan citra biner hasil

color filtering maka dilakukan proses *smoothing*, yang digunakan untuk menghilangkan *noise* pada citra tersebut.

Setelah proses *color filtering* selesai, maka yang terakhir proses pendeteksian lingkaran pada bola yang terdeteksi pada citra menggunakan metode *Circular hough transform*.

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Color Filtering

Untuk proses *color filtering* pada OpenCV menggunakan ruang warna HSV digunakan baris perintah berikut:

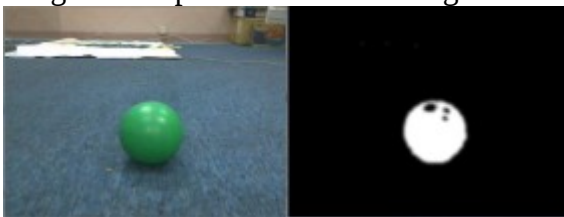
```
cvInRangeS(hsv,cvScalar(hsv_min,sat_min, val_min),cvScalar(hsv_max, sat_max, val_max, 0),thresholded);
```

Pada baris perintah tersebut terdapat batas filter maksimum dan minimum, dimana pada batas filter minimum perlu dilakukan pengaturan untuk tiap parameter (hue, saturation, value), dan pada batas filter maksimum hanya parameter hue yang perlu dilakukan pengaturan karena pada parameter saturation dan value bernilai maksimal atau 255. Kemudian akan dilakukan pengujian terhadap parameter filter untuk tiap-tiap warna. Untuk data pengujian dapat dilihat pada tabel 1

Tabel 1. Parameter *Color Filtering*

Warna Bola	HSV Min	HSV Max	Saturat ion Min	Val ue Min
Merah	0	15	175	50
Kuning	15	30	100	95
Hijau	45	75	75	50
Biru	100	120	75	50
Pink	169	179	100	50

Berikut adalah hasil bola warna hijau yang terdeteksi setelah melakukan proses *color filtering* dengan filter parameter sesuai dengan tabel 1.



Gambar 4. Hasil Proses *Color Filtering*

4.2 Pengujian Intensitas Cahaya



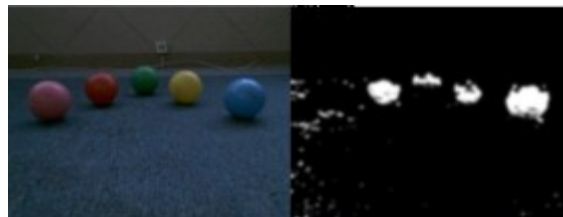
Gambar 5. Ruang Pengujian Intensitas Cahaya

Seperti yang terlihat pada gambar 5, terdapat 3 lampu pada ruang pengujian intensitas cahaya, dimana bola yang dideteksi tepat diletakan dibawah lampu 1. Kemudian akan dilakukannya proses pengujian secara berurutan sesuai tab

Tabel 2. Tabel Pengujian Intensitas Cahaya

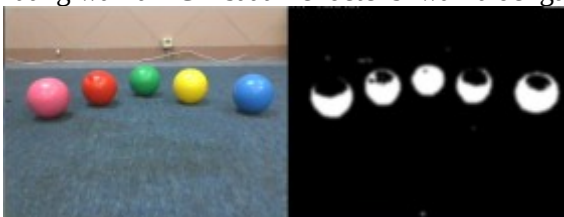
Pengujian	Lampu 1	Lampu 2	Lampu 3
1	Mati	Mati	Mati
2	Mati	Mati	Nyala
3	Mati	Nyala	Mati
4	Mati	Nyala	Nyala
5	Nyala	Mati	Mati
6	Nyala	Mati	Nyala
7	Nyala	Nyala	Mati
8	Nyala	Nyala	Nyala

Dengan melakukan pengujian sesuai prosedur diatas, akan didapatkan hasil proses *color filtering* sesuai dengan intensitas cahaya masing-masing. Berikut adalah hasil pengujian saat kondisi ketiga lampu mati.



Gambar 6. Lampu 1,2,3 Mati

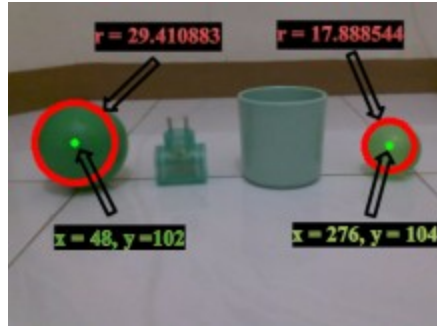
Dari gambar tersebut, pada kondisi sangat gelap gambar tidak mampu difilter sesuai warna. Dan ketika intensitas cahaya yang terlalu terang, permukaan bola akan memantulkan cahaya sehingga akan terdeteksi warna putih. Namun dapat disimpulkan bahwa ruang warna HSV lebih unggul dibandingkan ruang warna RGB saat mendeteksi warna dengan intensitas cahaya yang berubah-ubah.



Gambar 7. Lampu 1,2,3 Nyala

4.3 Pengujian Pendeteksian Lingkaran

Pendeteksian lingkaran dilakukan dengan menggunakan metode *Circular hough transform*. Untuk setiap lingkaran pada gambar yang dideteksi akan ditandai dengan gambar lingkaran sesuai radius bola dan titik tengah bola sesuai koordinat yang didapatkan dari *Circular hough transform*. Berikut gambar hasil pendeteksian lingkaran.



Gambar 8. Hasil Pendeteksian Lingkaran

Untuk mendeteksi lingkaran pada citra harus menggunakan citra dengan 1 *channel*, untuk itu digunakan citra *greyscale* atau *thresholding* dalam mendeteksi lingkaran.

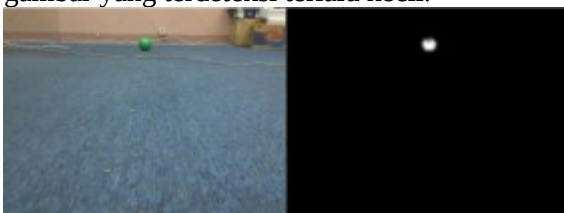
4.4 Pengujian Jarak Deteksi Lingkaran

Pengujian jarak dalam pendeteksian lingkaran dilakukan dengan cara menempatkan bola pada jarak yang ditentukan. Kemudian pada *console* akan didapatkan nilai parameter *x*, *y*, dan radius yaitu titik tengah dan radius lingkaran yang terdeteksi. Dengan melihat hasil pendeteksian lingkaran pada setiap *frame* gambar tersebut, maka dapat disimpulkan pada jarak tertentu apakah aplikasi dapat mendeteksi bola. Berikut adalah hasil pengujian pada tabel 3.

Tabel 3. Tabel Pengujian Intensitas Cahaya

Pengujian	Jarak(cm)	x	y	Radius
1	20	162	182	44.6878
2	30	166	140	37.3363
3	50	165	102	21.0950
4	100	168	68	13.0000
5	150	164	48	9.4339
6	200	162	40	7,2801
7	210	-	-	-

Parameter jarak pada tabel diatas merupakan jarak antara Robotino dengan bola. Berikut adalah gambar yang ditampilkan pada *window* saat Robotino pada jarak 2 meter. Pada jarak ± 2 meter data gambar masih dapat diproses untuk mendeteksi lingkaran. Namun ketika bola diletakkan pada jarak lebih dari 2,1 meter, data gambar tidak dapat di proses untuk mendeteksi lingkaran, dikarenakan gambar yang terdeteksi terlalu kecil.



Gambar 9. Deteksi Lingkaran pada Jarak 2 meter

4.5 Pengujian Sistem Keseluruhan

Pengujian terakhir adalah pengujian sistem secara keseluruhan dari awal hingga akhir, dimana pengujian dilakukan dengan menjalankan aplikasi secara keseluruhan. Kemudian meletakkan bola disekitar Robotino dan kemudian melihat Robotino apakah mampu mendeteksi bola. Setelah dalam posisi mengejar bola, Robotino akan diberikan halangan, ketika sensor jarak inframerah bagian depan mendeteksi terdapat halangan maka Robotino akan bergeser untuk menghindari halangan tersebut.

5. KESIMPULAN

1. Aplikasi sudah dapat mengintegrasikan Robotino dengan PC melalui jaringan *wireless*.
2. Aplikasi dapat mendeteksi warna benda dengan menggunakan ruang warna HSV yang selanjutnya digunakan untuk proses *color filtering*. Tetapi aplikasi tidak mampu mendeteksi bola pada saat kondisi terlalu gelap atau terang terang. Namun perubahan intensitas cahaya tidak mempengaruhi bola yang terdeteksi.
3. Aplikasi dapat mendeteksi benda berbentuk lingkaran menggunakan metode *Circular Hough Transform*. Tetapi ketika bola dengan diameter 10cm yang dideteksi terletak lebih dari 2,1 meter, maka gambar bola yang terlalu kecil tidak dapat dideteksi sebagai lingkaran.

6. DAFTAR PUSTAKA

Awcock, G.W. 1996. *Applied Image Processing*.Singapore:McGraw-Hill

Dhiemas, R.Y.S. 2011. *Pencarian Posisi Robot: Studi Kasus Pencarian Sumber Kebocoran Gas*. (Online). (<http://jiki.cs.ui.ac.id/index.php/jiki/article/download/39/44>). Diakses Tanggal 27 Oktober 2011

Munir, Rinaldi. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung : Informatika Bandung

Murinto, M. 2009. *Analisis Perbandingan Metode Intensity Filtering Dengan Metode Frequency Filtering Sebagai Reduksi Noise Citra Digital*. (Online). (<http://journal.uii.ac.id/index.php/Snati/article/view/1695/1477>). Diakses 15 September 2011.

Nahla, Gentang Syabba. 2011. *Tracking Bola Menggunakan Robotino*.(Online). (<http://digilib.its.ac.id/ITS-Undergraduate-3100010041280/14629>). Diakses 9 September 2011

ROS. 2010. *Robots Using ROS Robotino*.(Online). (<http://www.ros.org/news/2010/06/robots-using-ros-Robotino.html>). Diakses 29 September 2011

Syafi'i, Slamet Imam.2011. *Open Computer Vision(OpenCV)*. (Online). (<http://slametux.blogdetik.com/2011/10/19/open-computer-vision-opencv>). Diakses 30 September 2011

Tharom,Tabratas.2003. *Pengolan Citra pada Mobil Robot*. Bandung: ITB