

**PERANCANGAN *OBJECT TRACKING ROBOT* BERBASIS *IMAGE PROCESSING* MENGGUNAKAN RASPBERRY PI**



**Disusun sebagai salah satu syarat menyelesaikan Program Studi Strata I pada Jurusan Teknik  
Elektro Fakultas Teknik**

**Oleh:**

**LAKSONO BUDI PRIANGGODO**

**D 400 120 042**

**PROGRAM STUDI TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS MUHAMMADIYAH SURAKARTA**

**2016**

**HALAMAN PERSETUJUAN**

**PERANCANGAN *OBJECT TRACKING ROBOT* BERBASIS *IMAGE PROCESSING* MENGGUNAKAN RASPBERRY PI**

**PUBLIKASI ILMIAH**

oleh:

**LAKSONO BUDI PRIANGGODO**  
**D 400 120 042**

Telah diperiksa dan disetujui untuk diuji oleh:

Dosen Pembimbing



**Dr. Ratnasari Nur Rohmah, ST. MT**

NIK. 780

**HALAMAN PENGESAHAN**

**PERANCANGAN *OBJECT TRACKING ROBOT* BERBASIS *IMAGE PROCESSING* MENGGUNAKAN RASPBERRY PI**

**OLEH**

**LAKSONO BUDI PRIANGGODO**

**D 400 120 042**

**Telah dipertahankan di depan Dewan Penguji  
Fakultas Teknik  
Universitas Muhammadiyah Surakarta  
Pada hari Sabtu, 11 Juni 2016  
dan dinyatakan telah memenuhi syarat**

**Dewan Penguji:**

**1. Dr. Ratnasari Nur Rohmah, ST. MT**


**(Ketua Dewan Penguji)**

**2. Ir. Pratomo Budi Santosa, MT**

**(Anggota I Dewan Penguji)**

**3. Ir. Abdul Basith, MT**

**(Anggota II Dewan Penguji)**



**Dekan,**



**Is. Sri Sunarjono, MT. Ph.D**

**NIK. 682**

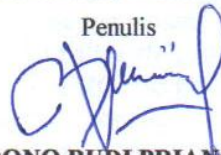
## **PERNYATAAN**

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

Surakarta, 11 Juni 2016

Penulis



**LAKDONO BUDI PRIANGGODO**

**D 400 120 042**

# PERANCANGAN *OBJECT TRACKING ROBOT* BERBASIS *IMAGE PROCESSING* MENGGUNAKAN RASPBERRY PI

## Abstrak

*Object tracking* merupakan teknik dalam bidang visi komputer yang digunakan untuk melakukan penjejakan pada suatu objek bergerak. *Object tracking* dilakukan dengan teknik pengolahan citra melalui rangkaian algoritma yang kompleks. Proses ini memberikan kemampuan pada komputer untuk mengetahui gerakan suatu objek tertentu. Penelitian ini merancang sebuah robot yang dapat mengikuti perpindahan gerak sebuah objek bola yang memiliki diameter 6,5 cm dengan warna tertentu. Proses pengolahan citra pada penelitian ini meliputi konversi ruang warna RGB ke HSV, *color filtering*, *edge detector*, dan *circle hough transform*. Proses pengolahan citra serta pengendalian gerak robot menggunakan *mini computer* Raspberry Pi 2 Model B dan kamera Raspberry Pi. Bahasa pemrograman menggunakan python yang telah tertanam library OpenCV. Library OpenCV merupakan library yang digunakan untuk dapat melakukan segala hal yang berkaitan dengan pengolahan gambar. Sistem mekanik menggunakan *pan tilt* sebagai penggerak kamera agar fleksibel dalam mengikuti perpindahan bola. Penggerak bodi robot menggunakan sepasang motor DC. Pegujian yang telah dilakukan ialah pengaruh resolusi gambar pada kamera terhadap sistem gerak, jarak pandang robot, pengenalan warna, pengenalan bentuk bola, kecepatan waktu tracking dan intensitas cahaya minimum. Hasil penelitian menunjukkan bahwa robot dapat melakukan *tracking* terhadap bola dengan menggunakan resolusi gambar terbaik pada ukuran 320x240 piksel, memiliki jarak pandang maksimum 113 cm, kecepatan maksimal mengikuti perpindahan objek 22,6 cm/detik dan mampu mengenali bola pada intensitas cahaya minimum 21,0 lux.

**Kata Kunci:** object tracking, image processing, circle hough, color filtering.

## Abstract

Object tracking is a technique in computer vision field that used to tracking on a moving object. Object tracking is done by image processing techniques through a combination of complex algorithm. This process gives the computer ability to know the movement of particular object. This study designs a robot that can follow the movement of a ball with diameter is 6.5 cm with particular color. Image processing in this study is conversion of RGB to HSV color space, color filtering, edge detector, and circle hough transform. Image processing and motion control of robot using Raspberry Pi 2 Model B mini computer and Raspberry Pi camera. Language programming that use is python with OpenCV library. OpenCV library is used to do all things related for image processing. The mechanical system using a pan tilt as camera driver to get flexibility in following the movement of ball. Robot's body move using a pair of DC motors. The test that have been done is influence the resolution of picture on the camera motion system, robot visibility, color recognition, ball's shape recognition, the speed when tracking, and minimum light intensity. The result show that robot can tracking a ball using the best image resolution in 320x240 pixel, it has a maximum visibility is 113 cm, maximum speed of object follow the movement is 22.6 cm/sec and it can recognize ball in minimum light intensity is 21.0 lux.

**Keywords:** object tracking, image processing, circle hough, color filtering.

## 1. PENDAHULUAN

Robot adalah mesin yang dapat melakukan berbagai pekerjaan yang menggantikan satu jenis atau lebih pekerjaan yang biasa dikerjakan oleh manusia atau suatu peralatan yang bekerja sendiri. Dalam melakukan pekerjaannya, robot mengikuti suatu pola gerakan yang ditentukan oleh pengguna robot. Sistem kontrol robot merupakan sistem yang digunakan oleh manusia untuk menentukan pola gerak robot. Terdapat tiga bagian

utama yang membangun suatu sistem kontrol, yaitu *input* (masukan), *processor* (pengolah), dan *output* (keluaran). *Input* adalah sekumpulan informasi yang ditangkap oleh robot. Informasi ini diperoleh dari sensor yang berperan sebagai indra bagi robot. Sensor merupakan indra bagi robot yang dapat mengenali berbagai parameter di sekitar lingkungan. Informasi yang didapat pada bagian *input* tidak dapat langsung digunakan melainkan harus diproses terlebih dahulu. Pengolahan sinyal input ini dilakukan oleh *processor* yang merupakan otak dari robot. Hasil pengolahan sinyal ini mengendalikan bagian output robot untuk melakukan kerja yang diinginkan oleh perancang robot. *Output* umumnya merupakan bagian mekanik pada robot yang melakukan gerakan tertentu. Bisa dibayangkan robot bekerja selayaknya manusia, semisal agar dapat mendengar menggunakan *microphone*, untuk berbicara menggunakan *loudspeaker*, untuk bergerak menggunakan *actuator*, untuk melihat menggunakan kamera, untuk merasakan menggunakan sensor, dan untuk berfikir menggunakan *processor*.

Supaya robot dapat melihat objek atau suasana sekitarnya dibutuhkan sebuah indra penglihatan yang dapat terintegrasi dengan sistem. Indra penglihatan untuk robot dapat menggunakan kamera. Kamera merupakan perangkat keras yang berfungsi menangkap gambar dan mengubahnya ke dalam bentuk citra digital yang dapat dibaca dan diproses oleh komputer. Citra digital ini merupakan sinyal masukan yang akan diproses pada komputer dapat menghasilkan informasi yang berguna untuk keperluan tertentu pada sistem. Pengolahan citra dengan komputer ini dikenal sebagai pengolahan citra digital.

Pengolahan citra digital bisa dibagi dalam dua kategori, yaitu pengolahan citra dengan keluaran berupa citra, dan pengolahan citra yang keluarannya berupa suatu keputusan hasil analisis citra. Kategori pertama digunakan dalam proses perbaikan citra atau manipulasi citra untuk keperluan tertentu. Kategori kedua sering disebut sebagai analisis citra, yang banyak digunakan dalam beberapa bidang, diantaranya dalam bidang *artificial intelligence* untuk kecerdasan komputasi (*computational intelligence*), *automatic control robotics* untuk *computer vision*, *neurobiology* untuk *biological vision* dan *machine learning* untuk *cognitive vision*.

*Computer vision* adalah salah satu penerapan dari pengolahan citra dalam kategori analisis citra. Tahapan pertama pada pengolahan citra untuk keperluan *computer vision* adalah proses akuisisi citra. Tahapan selanjutnya adalah tahapan pengolahan citra, yang bisa meliputi ekstraksi ciri citra yang dilanjutkan dengan analisis citra berdasar ciri citra. Hasil analisis akan merupakan suatu keputusan yang dalam hal *computer vision* untuk robot adalah untuk menentukan gerakan robot. Salah satu kunci untuk melakukan analisis citra adalah kita memerlukan sebuah perangkat komputasi yang berkemampuan tinggi. Karena keperluan ini, maka tidak cukup jika hanya dilakukan dengan *microcontroller* atau *microprocessor*.

Penelitian ini merancang sebuah sistem robotika yang menangkap objek melalui sensor kamera, dapat melakukan proses pengenalan objek menurut bentuk dan warna, serta dapat mengikuti pergerakannya (*object tracking*) secara *real-time*. Peneliti akan melakukan *tracking* terhadap objek bola dengan warna tertentu. Untuk dapat melakukan segala hal tersebut diperlukan sebuah sistem *embedded* yang dapat berperan sebagai komputer untuk melakukan pengolahan citra pada prosesor robot, serta *microcontroller* sebagai pengendali untuk rangkaian elektronik *actuator* pada bagian keluaran robot.

Pada proses akuisisi citra, peletakan kamera pada sebuah sistem sangat berpengaruh pada hasil yang diperoleh. Terdapat 2 macam peletakan kamera pada sistem yaitu *eye in hand* dan *stand alone*. Kragic dan Christensen (2011) menyatakan bahwa peletakan unit kamera untuk sebuah sistem dapat secara *eye in hand* atau *stand alone*. *Eye in hand* merupakan metode peletakkan kamera pada *end reflector* (bagian ujung) dari robot dan *stand alone* merupakan metode peletakkan kamera terpisah dari bagian tubuh robot. Dengan menggunakan metode *eye in hand* sistem akan langsung melihat objek yang akan diikuti secara langsung sedangkan metode *stand alone* sistem akan dapat melihat posisi objek serta robot dari jarak jauh. Marchand (2007) menyatakan bahwa peletakkan kamera sangat memengaruhi hasil gambar yang diperoleh, pencahayaan juga dapat mempengaruhi gambar yang diperoleh dan akan diproses. Pada penelitian tersebut, terdapat berbagai cara untuk memperoleh gambar yang bagus seperti memaksimalkan intensitas cahaya dan meningkatkan kontras antara objek dengan latar, dan teknik pengambilan yang tepat.

Pada penelitian ini obyek yang akan dikenali oleh robot adalah obyek berbentuk bola. Obyek bola pada penelitian ini dikenali sebagai bentuk lingkaran. Ikwuagwu (2010) menyatakan bahwa sebuah fungsi pada Open CV yang disebut dengan *hough circles* biasa digunakan untuk mendeteksi seluruh bentuk lingkaran pada sebuah gambar 2D.

Selain pengenalan bentuk bola, pada penelitian ini robot dirancang untuk dapat mengenali warna. Ekstraksi ciri warna oleh robot dilakukan dengan *colour filtering*. Pada penelitian sebelumnya terdapat beberapa peneliti yang pernah melakukan percobaan mengenai perbandingan dari penggunaan *color filtering* pada ruang warna HSV dan RGB. Beberapa penelitian memperlihatkan hasil yang didapat pada *color filtering* pada ruang warna HSV lebih bagus dibandingkan dengan RGB. Sural, Qian dan Pramanik (2002) menganalisis sifat dari ruang warna HSV dengan penekanan pada variasi persepsi visual dalam nilai *hue*, *saturation* dan *value* dari suatu piksel *image*. Penelitian ini mengekstraksi fitur piksel dengan memilih *hue* atau *value* sebagai nilai dominan berdasarkan nilai saturasi piksel. Segmentasi ruang warna menggunakan metode ini memberikan identifikasi yang lebih baik terhadap objek dalam sebuah *image* dibandingkan dengan yang dihasilkan menggunakan ruang warna RGB. Sedangkan Yustinus (2012) menyatakan bahwa metode *color filtering* RGB untuk segmentasi warna masih belum dapat memahami objek secara akurat disebabkan oleh faktor dari intensitas cahaya yang belum diperhitungkan.

## **2. METODE**

### **2.1 Alat dan Bahan**

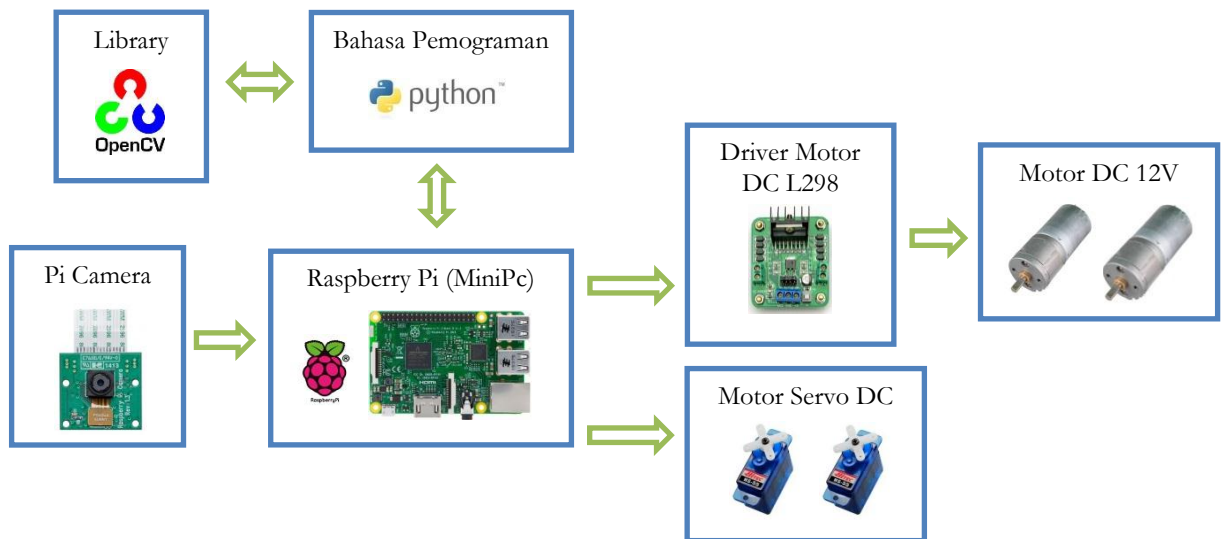
Peralatan dan komponen elektronika yang akan digunakan dalam perancangan ini meliputi :

- a. Raspberry Pi 2 Model B.
- b. Modul Kamera Raspberry Pi.
- c. Baterai Lippo 3Cell 2200mA.
- d. Konektor HDMI to VGA.
- e. Kabel LAN.
- f. Motor DC 12Volt.

- g. Motor Servo Micro.
- h. IC L298N, Resistor, Dioda, LED.
- i. Kabel, PCB, Solder, Timah, Atraktor.
- j. Obeng, AVO Meter, Tang Potong.
- k. Roda Tank Mainan.
- l. Komputer/laptop.

## 2.2 Perancangan

Dalam perancangan *object tracking robot* berbasis *image processing* menggunakan Raspberry Pi terdapat 2 tahap perancangan yaitu *hardware* dan *software*. Pada robot ini terpasang Pi Camera yang bekerja melakukan pengambilan gambar secara terus menerus dengan selang waktu tertentu (*streaming*), kemudian Raspberry Pi sebagai komputer akan memproses gambar dengan library Open CV menggunakan bahasa pemrograman python. Dari informasi yang didapat maka robot akan bergerak mengikuti pergerakan posisi bola dengan menggunakan motor DC 12 Volt. Digunakan motor servo agar kamera leluasa mengikuti perpindahan bola secara *real-time*. Raspberry pi menjadi sebuah unit komputer serta *microcontroller* yang dapat melakukan proses pengolahan citra sekaligus dapat mengendalikan rangkaian elektronik lain untuk melakukan pergerakan pada robot.

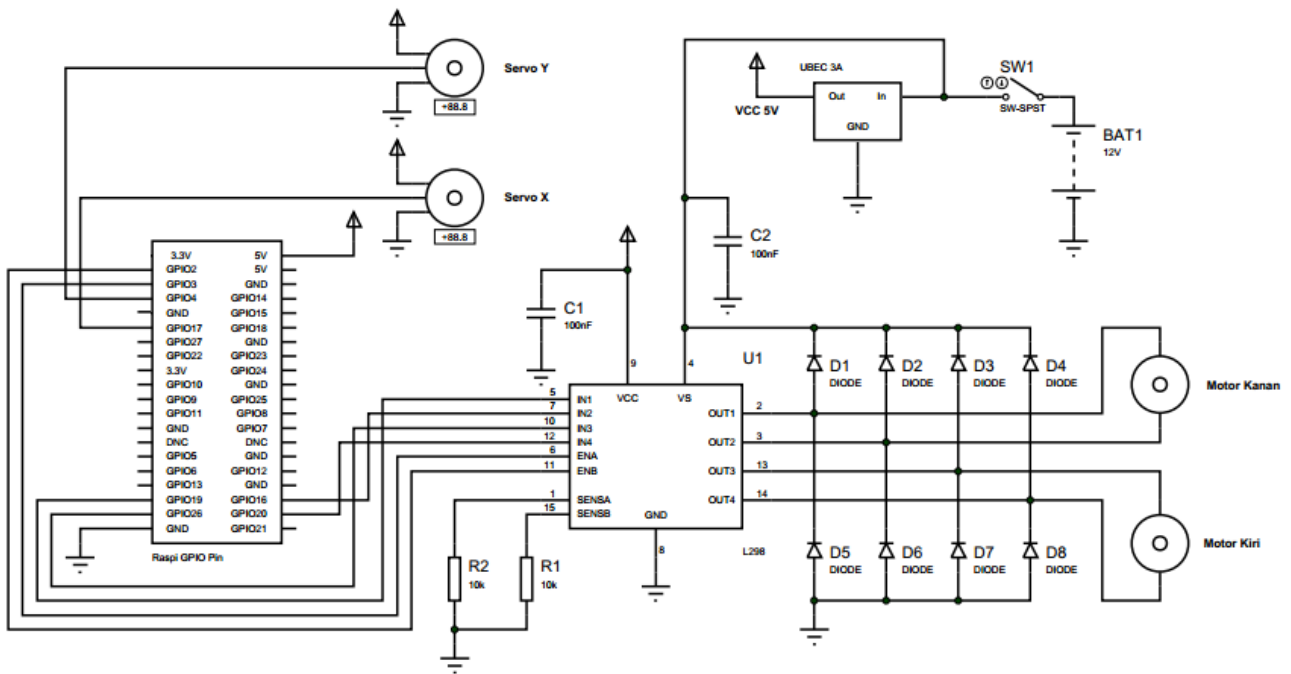


Gambar 1. Diagram blok sistem.

### 2.2.1 Perancangan Hardware

Perancangan *hardware* dilakukan dengan melakukan desain mekanik serta elektronik robot. Skema rangkaian dapat dilihat pada gambar 2. GPIO pin pada Raspberry Pi mengatur secara langsung pergerakan 2 motor servo dengan menyalurkan pulsa PWM (*Pulse Width Modulation*). Untuk melakukan pengendalian arah dan kecepatan motor DC melalui perantara IC L298N, karena Raspberry Pi bekerja pada tegangan 0-3.3V sedangkan motor bekerja pada tegangan 12V. Pada sistem ini digunakan sumber daya baterai 12 V DC untuk motor DC. Penggunaan UBEC 5V 3A ialah untuk memberikan catu kepada Raspberry Pi, motor servo dan IC L298N.

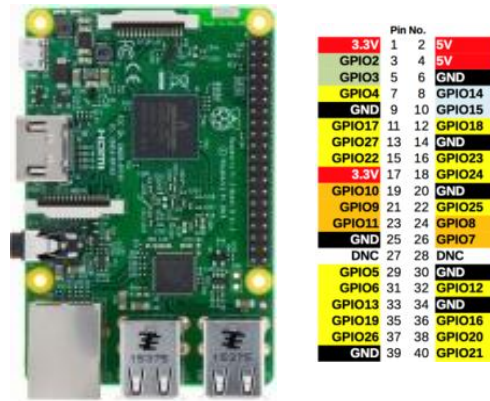




Gambar 2. Skema rangkaian.

Penggunaan *minicomputer* dipilih Raspberry Pi karena fitur yang disediakan sesuai dengan kebutuhan untuk pembuatan sistem robot ini, desain Raspberry Pi didasarkan seputar SoC (*System-on-a-Chip*) Broadcom BCM2835, yang telah menanamkan *prosesor* QuadCore ARM dengan kecepatan *clock* sebesar 900 MHz, VideoCore IV GPU, dan 1 Gigabyte RAM (Pi 2 model B). Penyimpanan data didisain tidak untuk menggunakan *hard disk* atau *solid-state drive*, melainkan mengandalkan kartu SD (*SD memory card*) untuk *booting* dan penyimpanan jangka panjang. Sistem operasi utama Raspberry Pi menggunakan Debian GNU/Linux dan bahasa pemrograman python. Komputer ini dilengkapi dengan 4 USB versi 2.0 Port yang dapat dihubungkan dengan perangkat yang menggunakan port USB apapun. Komputer dilengkapi juga dengan ethernet port untuk sambungan LAN (Local Area Network), HDMI port untuk disambungkan dengan perangkat layar LCD/LED dengan kualitas HD (*High Definition*), *connector* 3,5mm untuk disambungkan ke perangkat *speaker* atau *headset*, dan USB *micro* untuk *power supply*. Selain itu disediakan juga 2 Port khusus untuk kamera dan LCD Raspberry Pi. Komputer ini menyediakan 40 pin GPIO (*General Purpose Input Output*) agar Raspberry Pi dapat digunakan sebagai pengendali rangkaian elektronik.

Kamera yang digunakan untuk melakukan proses akuisisi gambar yaitu Pi Camera yang merupakan kamera khusus yang didesain untuk *minicomputer* Raspberry Pi. Dengan ukuran kecil, modul kamera Raspberry Pi dapat digunakan untuk mengambil gambar dengan kualitas *high definition* memiliki kualitas 5 MP mendukung resolusi video 1080p 30fps ,720p 60 fps dan VGA90, dapat bekerja pada semua model Raspberry Pi yang terhubung pada port CSI.



Gambar 3. Detil konfigurasi pin GPIO Raspberry Pi.



Gambar 4. Bentuk fisik modul kamera Raspberry Pi.

Sebagai penggerak untuk robot agar dapat berjalan digunakan sepasang motor DC 12 Volt. Keuntungan utama motor DC adalah dalam hal pengendalian kecepatan serta arah putaran motor DC tersebut sangat mudah. Motor ini dapat dikendalikan dengan mengatur tegangan sumber atau arus yang mengalir pada motor DC. Dengan meningkatkan tegangan sumber motor DC akan menambah kecepatan putaran sedangkan menambah arus yang mengalir pada motor DC akan menurunkan kecepatan.



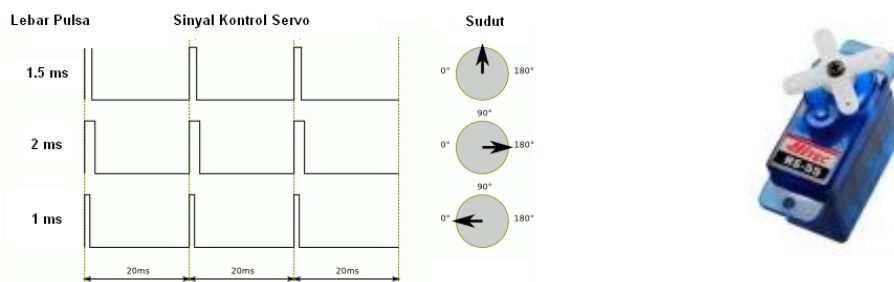
Gambar 5. Bentuk fisik motor DC.

Supaya Raspberry Pi dapat melakukan *controlling* terhadap motor DC digunakan *driver* motor. Kelebihan dari *driver* motor ini adalah cukup presisi dalam mengendalikan motor DC dan mudah dikendalikan. Untuk mengendalikan driver L298N ini dibutuhkan 6 pin. Pada prinsipnya rangkaian driver motor L298N ini hanya mengatur tegangan dan arus sehingga kecepatan dan arah motor dapat disesuaikan dengan keinginan.



Gambar 6. Bentuk fisik IC L298N

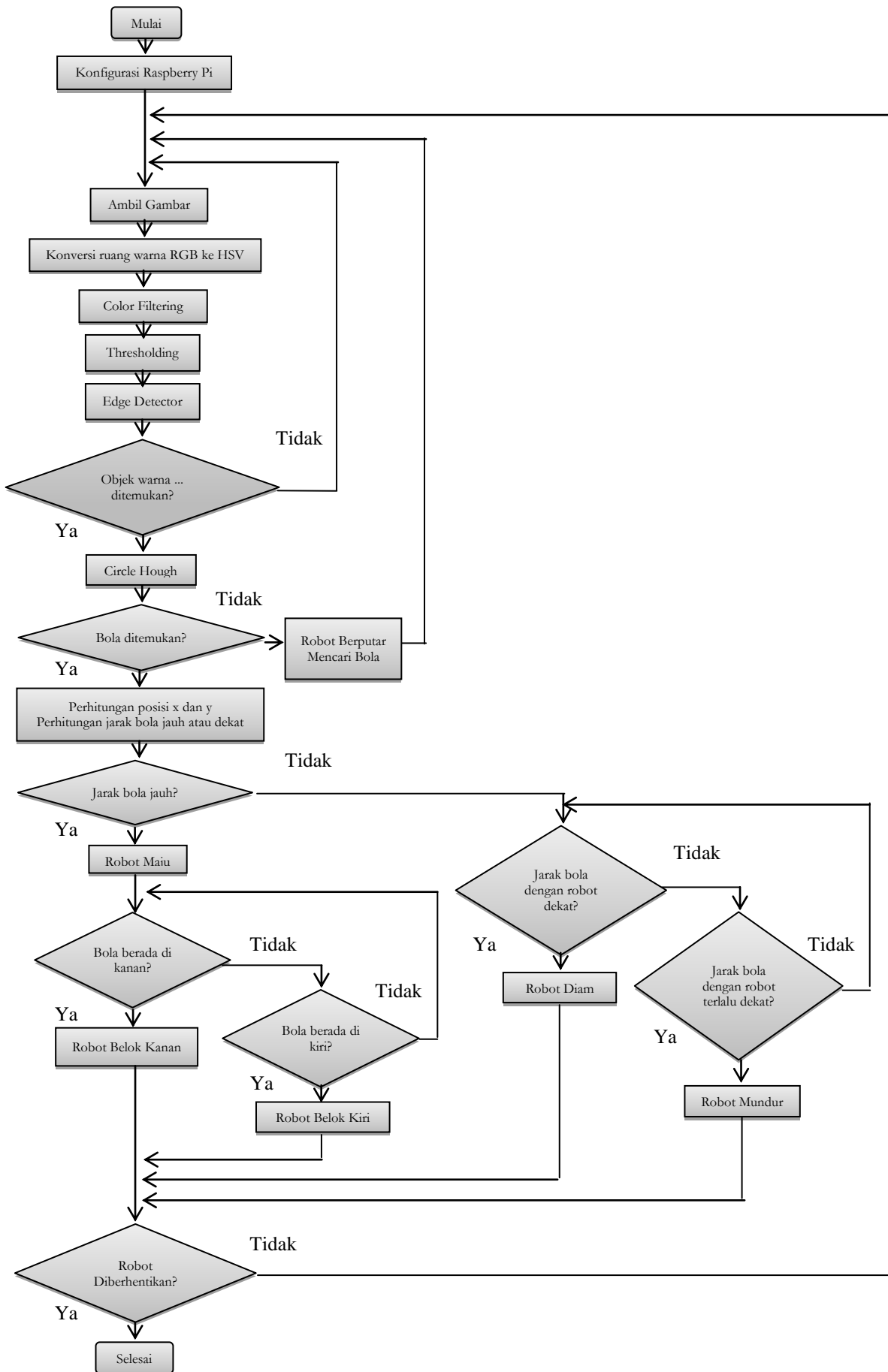
Agar kamera dapat melakukan *tracking* terhadap objek dibuat sistem mekanik yang dapat membuat kamera leluasa bergerak dengan sumbu sejajar x dan y, untuk melakukan hal tersebut *actuator* yang cocok ialah motor servo. Untuk mengendalikan putaran motor servo dilakukan dengan mengirimkan pulsa kontrol dengan frekuensi 50 hz dengan periode 20 ms dan *duty cycle* yang berbeda.



Gambar 7. Sinyal pulsa motor servo.

### 2.2.2 Perancangan Software

Pada perancangan software dilakukan dengan pembuatan program pada Python Idle yang merupakan compiler bawaan pada Raspberry Pi. Dalam menjalankan suatu program hardware Raspberry Pi perlu dilakukan pengaturan terlebih dahulu, hal-hal yang perlu diatur pada pembuatan robot ini ialah pin GPIO mana saja yang digunakan kemudian resolusi gambar dari kamera, pada sistem ini digunakan resolusi gambar 320x240 piksel. Awal mula proses yang dilakukan oleh sistem ialah pengambilan gambar melalui Pi Camera. Setelah itu proses pengolahan citra, mula – mula gambar yang diambil melalui pi camera ruang warna-nya yang mula-mula RGB perlu di konversi ke ruang warna HSV. Proses kedua pada pengolahan citra dalam sistem ini adalah proses *color filtering*. Proses ini dilakukan untuk melakukan pengambilan piksel pada suatu *image* yang memiliki nilai intensitas warna tertentu. Setelah didapatkan objek dengan warna tertentu saja, langkah selanjutnya ialah proses *thresholding* yang bertujuan untuk mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk obyek dan *background* dari citra secara jelas. Agar memudahkan proses pelacakan bentuk lingkaran pada sebuah citra digital digunakan proses *edge detector*. Proses ini bertujuan untuk melakukan perubahan format gambar hanya menampilkan bagian tepi saja. Selanjutnya dapat dilakukan *circle hough* untuk mendeteksi objek yang berbentuk lingkaran. Pola gerakan robot pada saat posisi objek jauh maka robot akan bergerak maju mendekati objek, pada saat dekat akan diam, dan pada saat terlalu dekat akan mundur. Ketika maju robot akan cenderung kekanan jika objek berada di kanan dan akan cenderung ke kiri jika objek berada di kiri. Algoritma pemrograman yang dibuat untuk sistem robot dapat dilihat pada gambar 8.



Gambar 8. Flowchart program.

Pada sistem ruang warna HSV, *hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning digunakan untuk menentukan kemerahan (*redness*), kehijauan (*greeness*), dsb. *Saturation* menyatakan kemurnian atau kekuatan warna. *Value* menyatakan kecerahan dari warna, nilainya berkisar antara 0-100 % apabila nilainya 0 maka warnanya akan menjadi hitam, semakin besar nilai maka semakin cerah dan muncul variasi-variasi baru dari warna tersebut. Untuk merubah nilai RGB menjadi nilai HSV dapat menggunakan teori trAVIS, dapat dilihat pada persamaan 1 hingga 9.

$$R' = \frac{R}{255} \quad (1)$$

$$G' = \frac{G}{255} \quad (2)$$

$$B' = \frac{B}{255} \quad (3)$$

$$Cmax = \max(R'.B'.C') \quad (4)$$

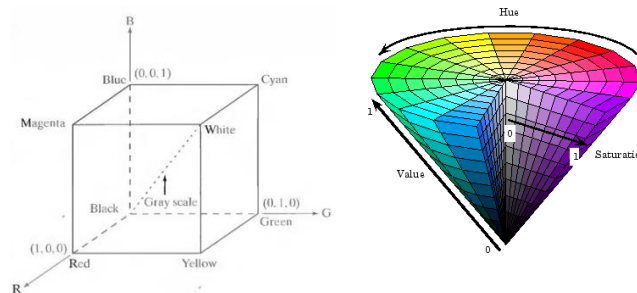
$$Cmin = \min(R'.B'.C') \quad (5)$$

$$\Delta = Cmax - Cmin \quad (6)$$

$$H = \begin{cases} 0, & \Delta = 0 \\ 60x \left( \frac{G'-B'}{\Delta} \text{mod} 6 \right), & Cmax = R' \\ 60x \left( \frac{B'-R'}{\Delta} + 2 \right), & Cmax = G' \\ 60x \left( \frac{R'-G'}{\Delta} + 4 \right), & Cmax = B' \end{cases} \quad (7)$$

$$S = \begin{cases} 0, & Cmax \\ \frac{\Delta}{Cmax}, & Cmax \end{cases} \quad (8)$$

$$V = Cmax \quad (9)$$



Gambar 9. Model ruang warna RGB / CYMK dan HSV.

Cara yang dilakukan pada *color filtering* ialah dengan menentukan batas atas serta batas bawah nilai dari sebuah piksel pada gambar, ketika dalam suatu image terdapat piksel yang memiliki nilai intensitas warna diantara batas atas dan bawah yang dikehendaki maka piksel tersebut akan diloloskan.

Pada proses *thresholding*, piksel-piksel obyek dan *background* dikelompokkan menjadi dua mode yang dominan. Cara untuk mengekstraks obyek dari *background* adalah dengan memilih nilai *threshold* T yang memisahkan dua mode tersebut. Nilai T didapat dari proses *color filtering* menggunakan ruang warna

HSV. Kemudian untuk sembarang titik  $(x,y)$  yang memenuhi  $f(x,y) > T$  disebut titik obyek, selain itu disebut titik *background*. Kesuksesan metode ini bergantung pada seberapa bagus teknik partisi *histogram*. Citra hasil *thresholding* dapat didefinisikan sebagaimana persamaan 10.

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases} \quad (10)$$

Keuntungan utama dari *circle hough* adalah dapat mendeteksi sebuah tepian dengan celah pada batas fitur dan secara relatif tidak dipengaruhi oleh derau atau noise. Dalam mendeteksi lingkaran dikerjakan pada ruang dimensi yang kompleks, yaitu dalam parameter ruang 3D  $(X_0, Y_0, R)$ . Di mana  $X_0$  dan  $Y_0$  merupakan koordinat pusat lingkaran dan  $r$  adalah jari-jari lingkaran seperti persamaan 11. Hasil dari *circle hough* adalah berupa koordinat  $(x,y)$  titik pusat lingkaran dan diameternya, parameter ini dijadikan titik acuan untuk melakukan perhitungan pada posisi dan jarak bola dengan robot.

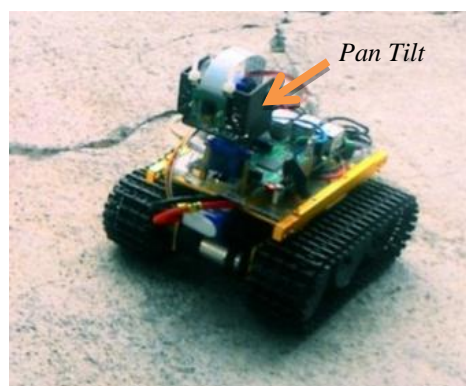
$$(x - x_0)^2 + (y - y_0)^2 = r^2 \quad (11)$$

Segala keperluan proses pengolahan citra dapat tersedia dengan penggunaan library Open CV. OpenCV (*Open Computer Vision*) merupakan sebuah API (*Application Programming Interface*) library yang sudah sangat familiar pada pengolahan citra *computer vision*. OpenCV adalah library open source untuk *computer vision* yang didesain untuk aplikasi *real-time*, memiliki fungsi-fungsi akuisisi yang baik untuk *image/video*.

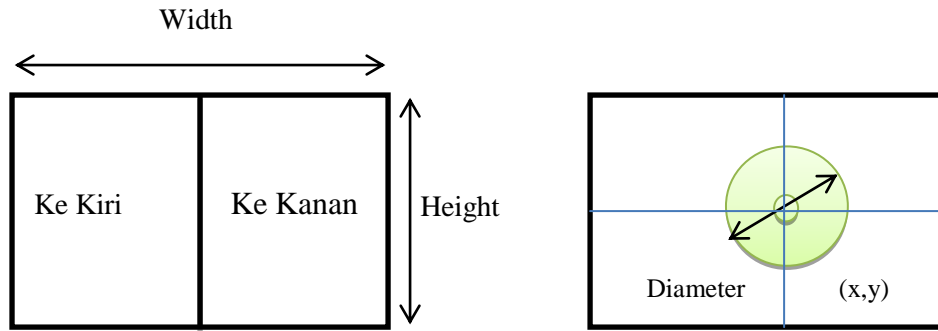
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Sistem Pergerakan Robot

Pada hasil percobaan robot bergerak mengikuti perubahan posisi bola. Ketika posisi bola berada di kanan dari *center* resolusi gambar maka robot akan bergerak cenderung kekanan dan ketika posisi bola berada di kiri dari *center* resolusi gambar maka robot akan bergerak ke kiri. Jarak robot dengan bola didapat dari diameter bola. Posisi robot akan selalu menjaga jaraknya dengan posisi bola. Lebih jelasnya dapat dilihat pada gambar 11. Pada percobaan ini menggunakan bola dengan diameter 6.5 cm.



Gambar 10. Bentuk fisik robot.



Gambar 11. Koordinat serta diameter gambar bola untuk sistem gerak robot.

### 3.2 Pengaruh Resolusi Gambar Dengan Sistem Kerja Robot

Dengan mengatur resolusi dari gambar yang ditangkap melalui kamera mempengaruhi kualitas gambar yang ditangkap, jarak pandang robot terhadap objek serta kecepatan dalam melakukan pergerakan untuk mengikuti perpindahan posisi objek karena memerlukan banyak waktu untuk proses pengolahan citra. Jarak robot dengan objek bola dihitung dari posisi kamera dengan pusat bola, sedangkan waktu *tracking* objek diambil saat robot bergerak dari posisi awal hingga posisi akhir hingga robot berhenti bergerak karena telah mengenali bola.



Gambar 12. Hasil gambar yang ditangkap pada beberapa resolusi.



Gambar 13. Pengukuran jarak pandang maksimum.

Tabel 1. Efek perubahan resolusi gambar kamera.

Resolusi Gambar	Jarak Maksimal	Waktu Tracking Objek
240x160	102 cm	4 detik
320x240	113 cm	5 detik
480x320	135 cm	6 detik
640x480	172 cm	error
720x560	180 cm	error
800x600	190 cm	error

Pada resolusi gambar 640x480 hingga 800x600 terjadi error yang membuat *streaming* gambar pada kamera terjadi lag serta pergerakan motor servo dalam mencari objek memiliki *delay* yang lama sehingga menyebabkan robot terlalu lama dalam mengenali objek bola.

### 3.3 Pemilihan Objek Berdasarkan Warna

Gambar dengan ruang warna RGB dan HSV sangat berbeda, gambar dengan ruang warna RGB didapat dari kamera, perubahan ruang warna menjadi HSV dilakukan menggunakan library OpenCV yang berguna mempermudah proses *color filtering*. Para peneliti sebelumnya menyatakan bahwa menggunakan ruang warna HSV untuk melakukan proses *color filtering* lebih bagus. Perbedaan hasil *color filtering* RGB dan HSV dapat dilihat pada gambar 14 hingga 17.

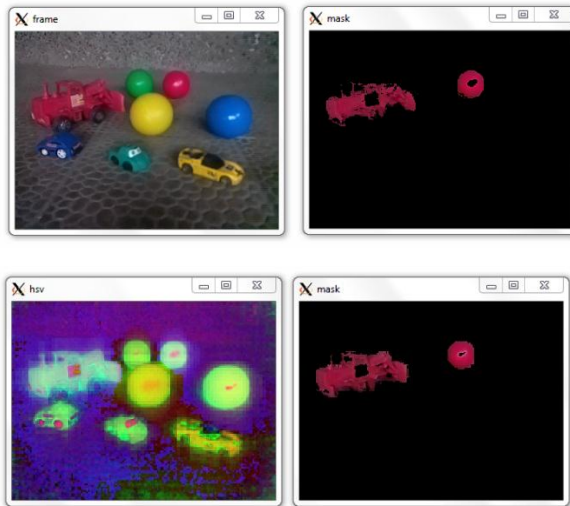
Tabel 2. RGB *color range filter*.

Warna	Nilai R ( <i>red</i> )		Nilai G ( <i>green</i> )		Nilai B ( <i>blue</i> )	
	Min	Max	Min	Max	Min	Max
Merah	90	255	0	40	20	100
Hijau	10	80	50	150	10	80
Biru	0	30	30	170	100	230
Kuning	170	240	130	220	0	60

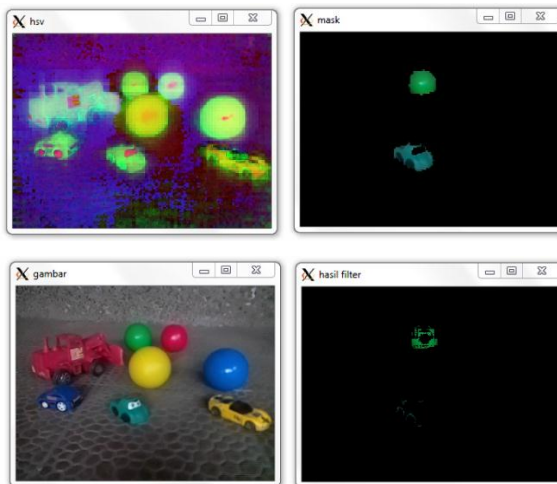
Tabel 3. HSV *color range filter*.

Warna	Nilai H ( <i>hue</i> )		Nilai S ( <i>saturation</i> )		Nilai V ( <i>value</i> )	
	Min	Max	Min	Max	Min	Max
Merah	150	179	150	250	50	255
Hijau	60	100	133	255	30	255
Biru	102	139	140	255	30	255
Kuning	0	35	90	255	70	255

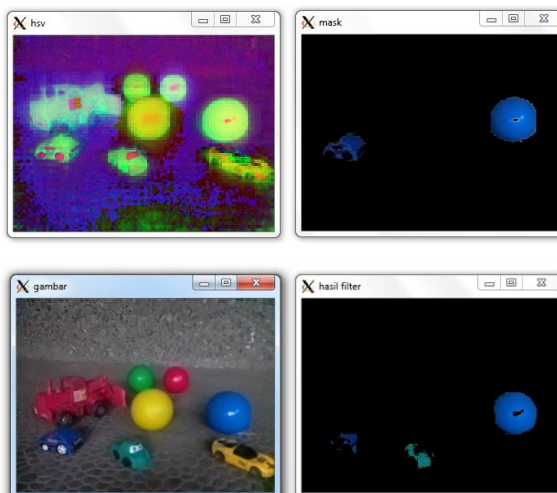




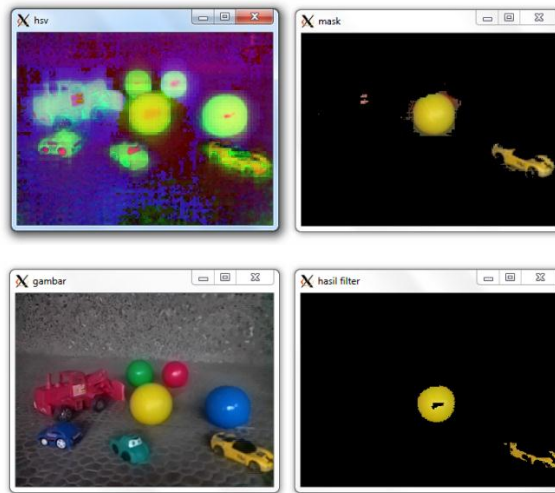
Gambar 14. Hasil *color filtering* gambar warna merah dengan ruang warna RGB dan HSV.



Gambar 15. Hasil *color filtering* gambar warna hijau dengan ruang warna RGB dan HSV.

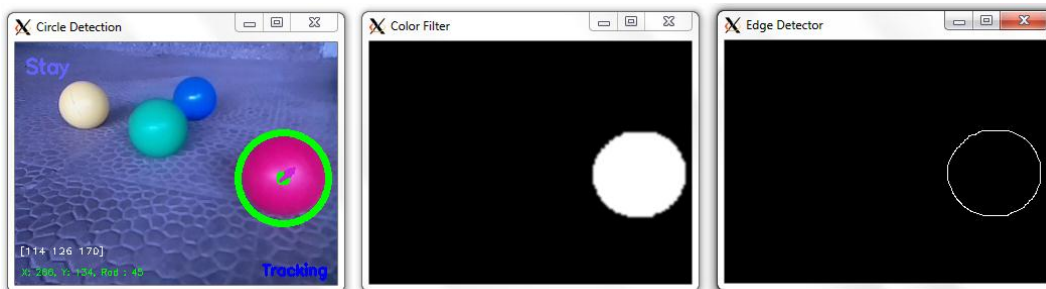


Gambar 16. Hasil *color filtering* gambar warna biru dengan ruang warna RGB dan HSV.

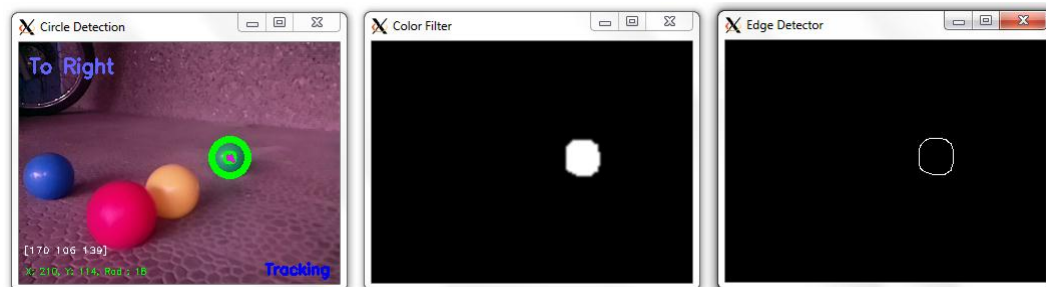


Gambar 17. Hasil *color filtering* gambar warna kuning dengan ruang warna RGB dan HSV.

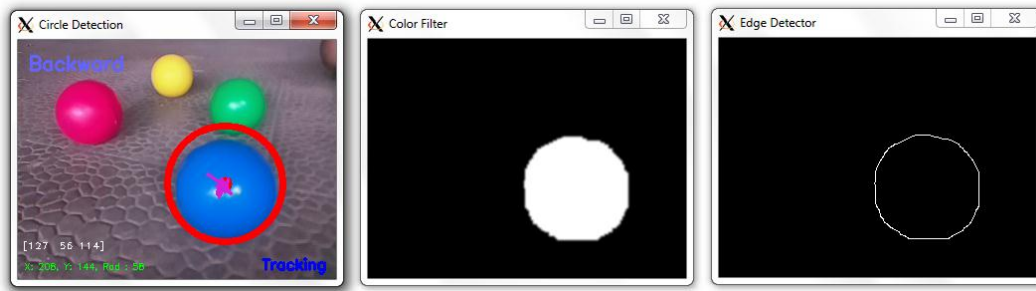
Agar dapat mengenali warna tertentu di berbagai kondisi pencahayaan menggunakan proses *trial and error*, sehingga didapatkan nilai *range* warna seperti pada tabel 2 dan 3. Kekurangan dari proses *color filtering* dengan ruang warna RGB yaitu proses *trial and error* sulit diperoleh nilai yang sesuai, pada saat percobaan di kondisi pencahayaan yang baik hasil yang diperoleh cukup baik namun di kondisi intensitas cahaya yang berbeda, maka hasilnya pun akan kurang sesuai yang di inginkan. Dengan pengaturan nilai range HSV maka didapatkan hasil seperti pada gambar 18 hingga 21. Saat robot dihadapkan beberapa bola yang memiliki warna berbeda robot dapat memilih bola dengan warna sesuai perintah yang dibuat.



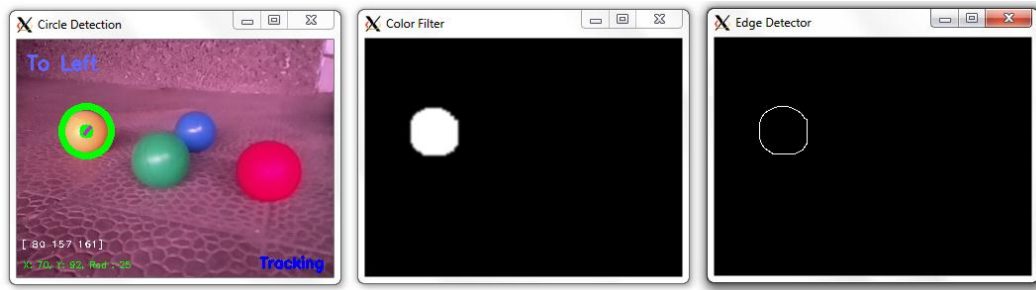
Gambar 18. Hasil pengolahan citra bola merah.



Gambar 19. Hasil pengolahan citra bola hijau.



.Gambar 20. Hasil pengolahan citra bola biru.



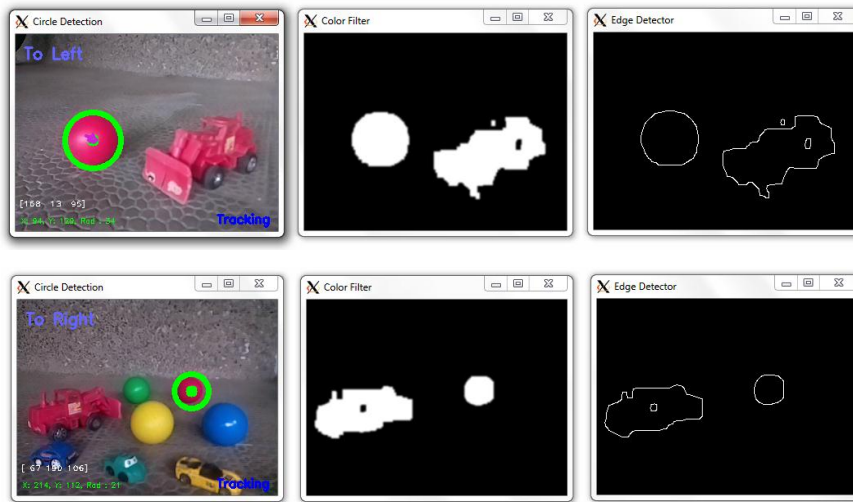
Gambar 21. Hasil pengolahan citra bola kuning.

### 3.4 Pemilihan Objek Berdasarkan Bentuk

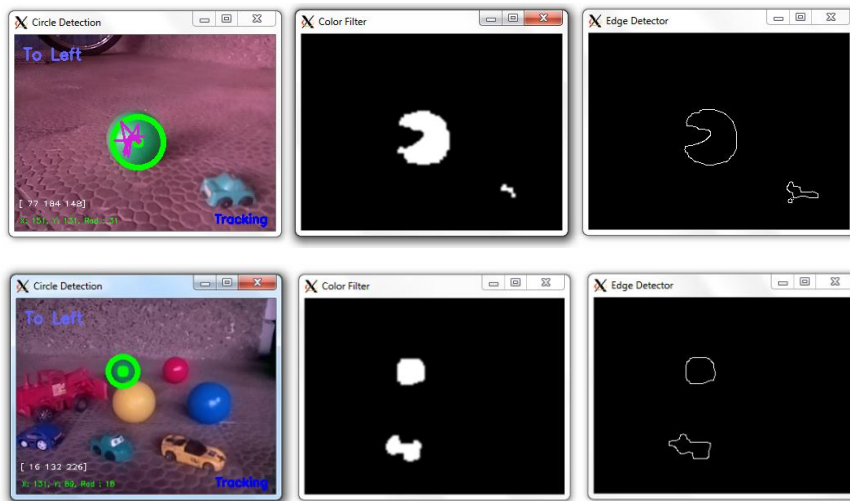
Saat dihadapkan dengan objek yang memiliki warna sama namun berbeda bentuk, robot dapat memilih objek mana yang memiliki bentuk lingkaran melalui proses *hough circle transform*, dapat dilihat pada tabel 4. Robot dapat mengenali bentuk lingkaran walaupun hasil proses *edge detector* menghasilkan bentuk lingkaran yang tidak sempurna, dapat dilihat pada gambar 22 hingga 25.

Tabel 4. Objek yang diikuti.

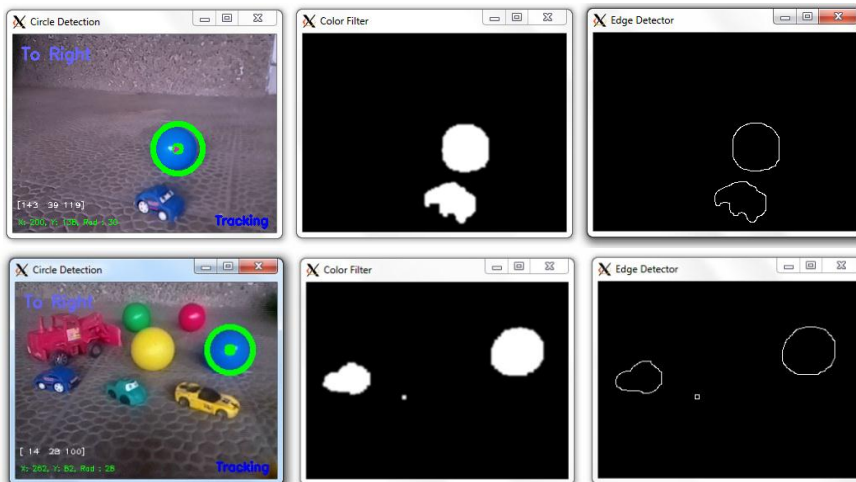
Pengaturan warna bola yang diikuti	Bola				Mobil-Mobilan			
	Merah	Hijau	Biru	Kuning	Merah	Hijau	Biru	Kuning
Merah	<b>diikuti</b>	-	-	-	-	-	-	-
Hijau	-	<b>diikuti</b>	-	-	-	-	-	-
Biru	-	-	<b>Diikuti</b>	-	-	-	-	-
Kuning	-	-	-	<b>diikuti</b>	-	-	-	-



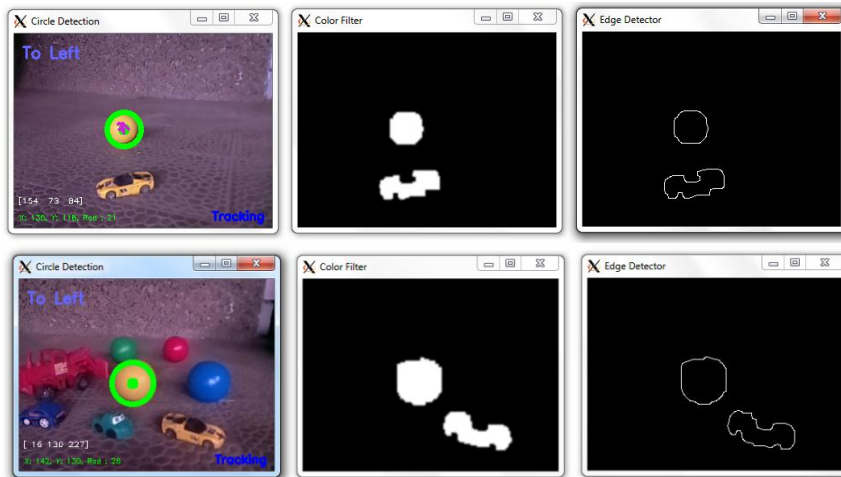
Gambar 22. Hasil pengolahan citra membandingkan bentuk objek berwarna merah.



Gambar 23. Hasil pengolahan citra membandingkan bentuk objek berwarna hijau.



Gambar 24. Hasil pengolahan citra membandingkan bentuk objek berwarna biru.



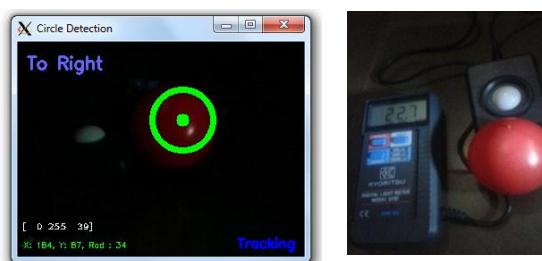
Gambar 25. Hasil pengolahan citra membandingkan bentuk objek berwarna kuning.

### 3.5 Pengaruh Intensitas Cahaya Terhadap Sistem Kerja Robot

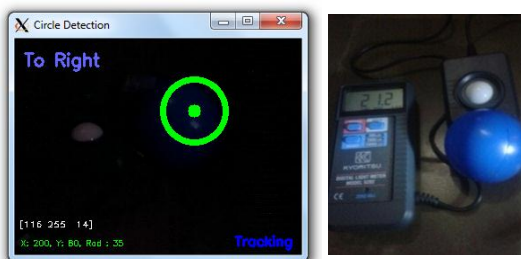
Saat penentuan nilai range warna dari proses *color filter* pada gambar dengan ruang warna HSV, nilai dari V (*value*) mempengaruhi nilai maksimum dan minimum tingkat kecerahan warna. Dapat diamati pada gambar 26 hingga 29 bahwa walaupun dalam keadaan gelap robot masih dapat mengenali warna dengan baik.

Tabel 5. Intensitas cahaya minimum robot dapat mengenali bola.

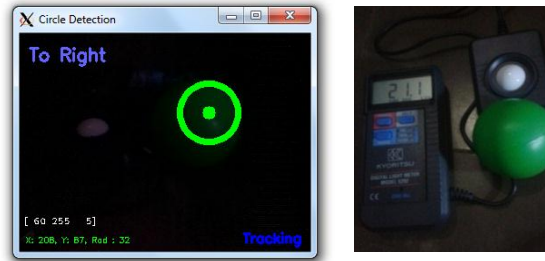
Warna	Intensitas Cahaya Minimum
Merah	22.7 lux
Hijau	21.2 lux
Biru	21.1 lux
Kuning	23.6 lux



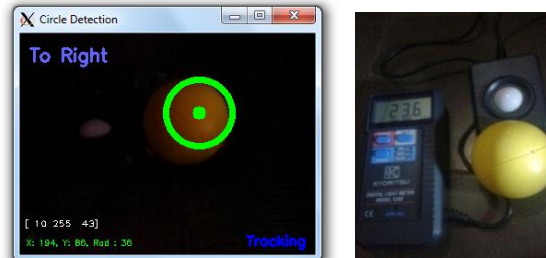
Gambar 26. Intensitas cahaya paling gelap dalam mengenali bola warna merah.



Gambar 27. Intensitas cahaya paling gelap dalam mengenali bola warna biru.



Gambar 28. Intensitas cahaya paling gelap dalam mengenali bola warna hijau.



Gambar 29. Intensitas cahaya paling gelap dalam mengenali bola warna kuning.

#### 4. PENUTUP

Pada hasil penelitian dapat disimpulkan bahwa penggunaan resolusi kamera mempengaruhi jarak objek yang dapat dikenali. Semakin besar resolusi kamera maka akan semakin jauh jarak objek dapat dikenali oleh robot. Meskipun demikian, besarnya resolusi ini akan menyebabkan kerja *processor* dalam proses pengolahan citra yang semakin berat. Penggunaan format gambar dengan ruang warna HSV untuk proses *color filtering* menghasilkan performa yang baik dalam mengenali objek dengan warna yang diinginkan. Pada kondisi pencahayaan yang bagus yaitu tidak terlalu terang dan tidak terlalu gelap didapatkan proses *color filtering* yang baik. Bentuk objek dapat diamati dengan baik melalui proses *edge detector*. Dengan menggunakan proses *circle hough*, kamera dapat mengenali objek yang berbentuk lingkaran dengan baik. Robot dapat melakukan *tracking* terhadap bola dengan menggunakan resolusi gambar terbaik pada ukuran 320x240 piksel, memiliki jarak pandang maksimum 113 cm, kecepatan maksimal mengikuti perpindahan objek 22,6 cm/detik dan mampu mengenali bola pada intensitas cahaya minimum 21,0 lux.

Saran untuk penelitian selanjutnya baiknya digunakan penggerak motor *omnidirectional* agar robot dapat bergerak bebas ke berbagai arah. Perlu ditambahkan sensor pendeteksi halangan agar tidak terjadi robot menabrak dinding atau benda lain. Penggunaan motor servo baiknya menggunakan motor servo jenis digital agar pergerakannya lebih halus serta derajatnya lebih teliti dan dapat bergerak 0° hingga 360°. Proses pengolahan citra diperlukan algoritma yang lebih kompleks lagi untuk mengurangi error dalam pengenalan objek.

## PERSANTUNAN

Penulis mengucapkan terima kasih kepada pihak yang telah membantu dalam pengerjaan tugas akhir ini, terutama kepada :

1. ALLAH SWT dan Nabi Muhammad SAW yang telah memberikan banyak kenikmatan dan kemuliaannya.
2. Orang Tua yang selalu mendo'akan, memberikan dukunganserta semangat dalam pengerjaan Tugas Akhir.
3. Bapak Ir. Sri Sunarjono, M.T, Ph.D, selaku Dekan Fakultas Teknik Universitas Muhammadiyah Surakarta.
4. Bapak Umar, S.T, M.T, sebagai Ketua Jurusan Teknik Elektro Universitas Muhammadiyah Surakarta.
5. Ibu Dr. Ratnasari Nur Rohmah, S.T, M.T, sebagai pembimbing Tugas Akhir ini yang selalu memberikan pengarahan kepada penulis.
6. Para Dosen Jurusan Teknik Elektro Universitas Muhammadiyah Surakarta.
7. Rekan-rekan seangkatan Saleh Syahmi, Dharu Rendro A.P, Juna Chairul E, Rheksi Hermawan, Dedy Wiratmoko, Fajar Andar C, Sahid Sholihin dan seluruh teman-teman Teknik Elektro lainnya.
8. Semua pihak yang telah membantu terselesaikannya tugas akhir ini yang tidak dapat saya sebutkan satu persatu.

## DAFTAR PUSTAKA

- Sural, S., Qian, G., Pramanik, S. (2002). Segmentation and histogram generation using the HSV color space for image retrieval. *Proceedings of IEEE International Conference on Image Processing*. 589-592.
- Marchand, E. (2007). Control Camera and Light Source Positions using Image Gradient Information. *IEEE Int. Conf. on Robotics and Automation*. 417 – 422.
- Evan, Y. (2010). *Thresholding citra*, kuliahinformatika.wordpress.com
- Sigit, H., & Agung, T. (2010). *Image processing dasar*, kliktedy.wordpress.com
- Kragic, D., Christensen, H.I. (2011). Survey on Visual Servoing for Manipulation: Centre for Autonomous Systems. *Numerical Analysis and Computer Science*.
- Ikwuagu, E. (2011). Design Of An Image Processing Algorithm For Ball Detection. *Computing Research Association*.
- Brigida, A. (2012). *Transformasi hough*, informatika.web.id
- Yustinus, P. (2012). Rancang Bangun Aplikasi Pendeteksi Bentuk Dan Warna Benda Pada Mobile Robot Berbasis Webcam. *Academia*.
- Agus, K. (2015). *Convert RGB To HSV for color tracking in Raspberry and openCV*, ilmu-otomasi.blogspot.com.