

**PEMANFAATAN KAMERA SEBAGAI SENSOR
ROBOT DUA KAKI**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Mesin**



Disusun Oleh :

Nama : Amrullah Mahardhika

No. Mahasiswa : 07525003

**JURUSAN TEKNIK MESIN
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2011

LEMBAR PENGESAHAN DOSEN PEMBIMBING

**PEMANFAATAN KAMERA SEBAGAI SENSOR
ROBOT DUA KAKI**

TUGAS AKHIR

Disusun Oleh :

Nama : Amrullah Mahardhika

No. Mahasiswa : 07525003



Yogyakarta, 21 November 2011

Pembimbing I,

Pembimbing II,

Agung Nugroho Adi, ST., MT.

Tri Setya Putra, ST.

LEMBAR PENGESAHAN DOSEN PENGUJI

**PEMANFAATAN KAMERA SEBAGAI SENSOR
ROBOT DUA KAKI**

TUGAS AKHIR

Disusun Oleh :

Nama : Amrullah Mahardhika

No. Mahasiswa : 07525003

Tim Penguji

Agung Nugroho Adi, ST., MT

Ketua

Tanggal : _____

Purtojo, ST., M.Sc

Anggota I

Tanggal : _____

Mohammad Faizun, ST., M.Eng

Anggota II

Tanggal : _____

Mengetahui

Ketua Jurusan Teknik Mesin

Agung Nugroho Adi, ST., MT.

LEMBAR PENGESAHAN DOSEN PEMBIMBING

**PEMANFAATAN KAMERA SEBAGAI SENSOR
ROBOT DUA KAKI**

TUGAS AKHIR

Disusun Oleh :

Nama : Amrullah Mahardhika

No. Mahasiswa : 07525003

Yogyakarta, 21 November 2011

Pembimbing I,



Agung Nugroho Adi, ST., MT.

Pembimbing II,



Tri Setya Putra, ST.

LEMBAR PENGESAHAN DOSEN PENGUJI

PEMANFAATAN KAMERA SEBAGAI SENSOR
ROBOT DUA KAKI

TUGAS AKHIR

Disusun Oleh :

Nama : Amrullah Mahardhika

No. Mahasiswa : 07525003

Tim Penguji

Agung Nugroho Adi, ST., MT

Ketua

Tanggal : 15-12-2011

Purtojo, ST., M.Sc

Anggota I

Tanggal : 15-12-2011

Mohammad Faizun, ST., M.Eng

Anggota II

Tanggal : 15-12-2011

Mengetahui

Ketua Jurusan Teknik Mesin



Agung Nugroho Adi, ST., MT.

DAFTAR GAMBAR

Gambar 2-1 Sistem Robot.....	7
Gambar 2-2 Robot Dua Kaki atau Robot <i>Humanoid</i>	8
Gambar 2-3 Motor Servo.....	9
Gambar 2-4 Hubungan Lebar Pulsa dengan Putaran Arah Motor Servo.....	10
Gambar 2-5 Susunan Pin Motor Servo.....	10
Gambar 2-6 Sistem Kontroller Berbasis Mikrokontroller ATmega16.....	12
Gambar 2-7 Tampilan Code Vision AVR.....	15
Gambar 2-8 Sensor Camera CMUcam3.....	16
Gambar 3-1 <i>Flowchat</i> Kaitan Keseluruhan Penelitian.....	22
Gambar 3-2 <i>Flowchart</i> Penelitian.....	23
Gambar 3-3 Urutan Perintah Pengklasifikasian Warna dan Penjejakan Obyek.....	27
Gambar 3-4 Contoh Perintah GT.....	30
Gambar 3-5 Nilai Hasil Perintah TC Terhadap Obyek Berwarna <i>Orange</i> pada Hiperteminal.....	30
Gambar 3-6 Tampak <i>Real</i> Obyek Bola Berwarna <i>Orange</i> pada CMCcam2GUI.....	31
Gambar 3-7 Tampilan dari CMUcam2GUI Hasil <i>Image Processing</i> Sensor Kamera CMUcam3.....	31
Gambar 3-8 <i>Flowchart</i> Memperoleh Hingga Mengolah Data.....	32
Gambar 3-9 Konfigurasi Pin pada ATmega16.....	35
Gambar 3-10 <i>Serial Port</i> Sensor Kamera CMUcam3.....	35
Gambar 3-11 Konfigurasi USART.....	36
Gambar 3-12 Program Interupsi Menerima Data.....	36
Gambar 3-13 Program Perintah GT dan TC.....	38
Gambar 3-14 Menbandingkan Nilai RGB dan Nilai Posisi dari Sistem Kontroller (LCD) dan Komputer (<i>Serial Watcher</i>).....	39
Gambar 3-15 <i>Flowchart</i> Penyelesaian Tugas Robot.....	40
Gambar 3-16 Scan Warna.....	43

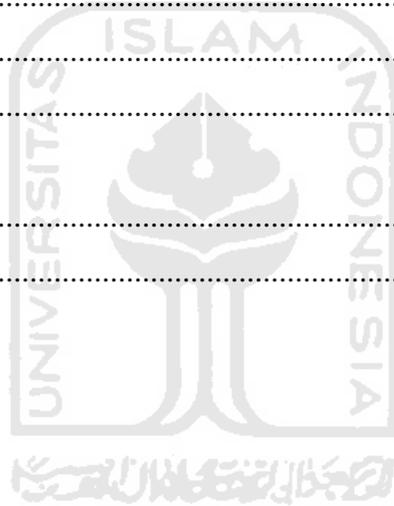
Gambar 3-17 Contoh Program Logika	45
Gambar 3-18 Jangkauan Penglihatan Sensor Kamera CMUcam	45
Gambar 4-1 Obyek Percobaan Pengklasifikasian Warna	48
Gambar 4-2 Tampak Warna Merah Hijau Biru dan Oranye dari Obyek Berwarna pada CMUcam3 <i>frame grabber</i>	48
Gambar 4-3 Ilustrasi Hasil dari Perintah TW	50
Gambar 4-4 Obyek Percobaan Penjejakan Obyek Berwarna	52
Gambar 4-5 Tampak <i>Real</i> Obyek Berwarna Merah Hijau Biru dan Oranye Pada CMUcam3 <i>frame grabber</i>	53
Gambar 4-6 Robot Beroda Mencari dan Mendapatkan Target.....	57
Gambar 4-7 Robot Dua Kaki atau <i>Humanoid</i> Mencari dan Mendapatkan Target.....	60



DAFTAR ISI

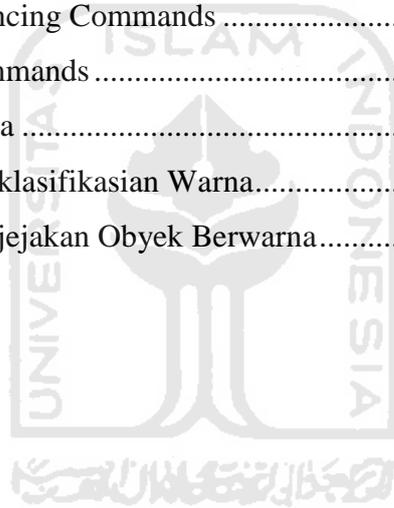
HALAMAN JUDUL	i
LEMBAR PENGESAHAN DOSEN PEMBIMBING	ii
LEMBAR PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERSEMBAHAN	iv
HALAMAN MOTTO	v
KATA PENGANTAR	vi
ABSTRAK.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Sistematika Penulisan	3
BAB II KAJIAN PUSTAKA.....	5
2.1 Definisi Robot.....	5
2.2 Mekanik Robot.....	7
2.3 Aktuator	8
2.4 Sistem Gerak atau <i>Locomotion system</i>	10
2.5 Sistem Kontroller	11
2.6 Sensor.....	15
BAB III METODOLOGI PENELITIAN	21
3.1 Pendahuluan	21
3.2 <i>Flowchat</i> Penelitian	23

3.3	Spesifikasi Alat dan Bahan	25
3.4	Komunikasi CMUcam3	26
3.4.1	Komunikasi CMUcam3 dengan Komputer.....	29
3.4.2	Komunikasi CMUcam3 dengan Sistem Kontroller	32
3.5	Penggabungan Sensor Kamera CMUcam3 dengan Robot	40
BAB IV HASIL DAN PEMBAHASAN		47
4.1	Pengklasifikasian Warna.....	47
4.2	Penjejakan Obyek Berwarna.....	52
4.3	Kemampuan Robot Mencari dan Mendapatkan Target	56
BAB V PENUTUP		65
5.1	Kesimpulan	65
5.2	Saran	65
DAFTAR PUSTAKA		67
LAMPIRAN.....		69



DAFTAR TABEL

Tabel 2-1 Buffer Commands	17
Tabel 2-2 Color Statistics Commands	17
Tabel 2-3 Windowing Commands	18
Tabel 2-4 Servo Commands.....	18
Tabel 2-5 Camera Module Commandsi.....	18
Tabel 2-6 Auxiliary I/O Commands	18
Tabel 2-7 Histogram Commands	18
Tabel 2-8 System Level Commands	19
Tabel 2-9 Color Tracking Commands	19
Tabel 2-10 Frame Differencing Commands	19
Tabel 2-11 Data Rate Commands	19
Tabel 2-12 Tipe Paket Data	20
Tabel 4-1 Percobaan Pengklasifikasian Warna.....	50
Tabel 4-2 Percobaan Pengjejukan Obyek Berwarna.....	55



HALAMAN PERSEMBAHAN

Laporan tugas akhir ini, saya persembakan kepada ayah, ibu, kakak, seluruh teman-teman dan dosen dari Teknik Mesin UII. Terimakasih telah memberikan doa, bimbingan, bantuan, semangat, kasih sayang, ketulusan dan pengorbanan. Sehingga saya dapat menyelesaikan studi ini dengan baik.

Amin ya robal alamin



HALAMAN MOTO

يَتَأْتِيهَا الَّذِينَ ءَامَنُوا اسْتَعِينُوا بِالصَّبْرِ وَالصَّلَاةِ إِنَّ اللَّهَ مَعَ الصَّابِرِينَ



Hai orang-orang yang beriman, Jadikanlah sabar dan shalatmu Sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar (Al-Baqarah: 153)

وَيَرْزُقْهُ مِنْ حَيْثُ لَا يَحْتَسِبُ وَمَنْ يَتَوَكَّلْ عَلَى اللَّهِ فَهُوَ
حَسْبُهُ إِنَّ اللَّهَ بَلِغُ أَمْرِهِ قَدْ جَعَلَ اللَّهُ لِكُلِّ شَيْءٍ قَدْرًا ﴿٣﴾

“...Barang siapa bersungguh-sungguh mendekati Allah (bertaqwa) niscaya akan diberi jalan keluar bagi setiap urusannya, dan akan diberi rizqi dari tempat yang tak pernah disangka-sangka, dan barang siapa yang bertawakal hanya kepada Allah niscaya akan dicukupi segala kebutuhannya... “.

(Al Qur'an surat Ath Thalaq : 3)

KATA PENGANTAR



Assalamu'alaikum Wr, Wb.

Segala puja dan puji syukur bagi Allah SWT yang telah melimpahkan taufik dan hidayahnya, sehingga penulisan laporan tugas akhir ini dapat terselesaikan. Sholawat serta salam kita curahkan kepada Nabi Muhammad saw, sebagai pembawa rahmat di muka bumi.

Laporan tugas akhir dengan judul “Pemanfaatan Kamera Sebagai Sensor Robot Dua Kaki Dengan Sepuluh Derajat Kebebasan ” ini dibuat sebagai salah satu syarat untuk meraih gelar Sarjana (S1) di Jurusan Teknik Mesin, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Dalam penyusunan laporan tugas akhir ini, banyak bantuan yang diberikan baik berupa materi maupun immateri. Sehingga dalam penyusunannya dapat berjalan dengan baik dan selesai tepat waktu. Dengan segenap ketulusan hati, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada :

1. Prof Dr. Edy Suandi Hamid, Mec., selaku rektor Universitas Islam Indonesia.
2. Ir. Gumbolo HS., M.Sc., selaku dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Agung Nugroho Adi, ST., MT., selaku ketua Jurusan Teknik Mesin, dan sebagai dosen pembimbing I yang telah banyak meluangkan waktu, memberikan petunjuk dan pengarahan.
4. Tri Setia Putra, ST., selaku dosen pembimbing II yang telah banyak meluangkan waktu, memberikan petunjuk dan pengarahan.
5. Segenap dosen dan karyawan Fakultas Teknologi Industri yang telah membantu baik dalam kegiatan akademis, maupun administratif, khususnya jurusan Teknik Mesin atas segala dedikasinya dalam memberikan ilmu kepada penulis serta memberikan bantuan dalam segala hal.

6. Bapak Sugeng Hidayat, Ibu Sri Purwaningsih, Sari Nila Krisna, selaku kedua orang tua dan kakak tercinta yang selalu memberikan perhatian, doa dan dukungan kepada penulis.
7. Semua sahabat dan teman-temanku di Jurusan Teknik Mesin.
8. Semua pihak yang telah ikut membantu kelancaran tugas akhir yang tidak mungkin disebutkan satu persatu.

Semoga dari segala doa, bantuan, bimbingan, semangat, kasih sayang, ketulusan dan pengorbanan yang telah diberikan, akan mendapatkan imbalan dari Allah SWT. Tidak lupa penulis memohon maaf apabila selama melaksanakan tugas akhir ini, terdapat kekhilafan dan kesalahan. Penulis menyadari sepenuhnya akan keterbatasan kemampuan yang dimiliki “*Tak ada gading yang tak retak*”, tidak ada manusia yang sempurna, kesempurnaan hanyalah milik Allah SWT. Oleh karena itu segala kritik dan saran yang sifatnya membangun demi kesempurnaan laporan ini sangat diharapkan.

Akhir kata, penulis berharap semoga laporan tugas akhir ini dapat bermanfaat bagi semua yang membaca, amin ya robal alamin.

Wassalamu 'alaikum Wr.Wb.

Yogyakarta, 21 November 2011

Penulis

ABSTRAK

Tugas akhir ini merupakan penelitian terhadap kemampuan dari sensor kamera CMUcam3 yang berupa pengklasifikasian warna dan penjejakan obyek berwarna, dan pemanfaatan kemampuan tersebut pada robot dua kaki atau robot humanoid untuk menyelesaikan tugasnya yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Penelitian diawali dari mencari tahu cara agar pengendali robot atau sistem kontroller memperoleh nilai dari CMUcam3, setelah itu barulah pengupayaan agar robot mampu menyelesaikan tugasnya dapat dilakukan. Karena untuk membuat robot mampu menyelesaikan tugasnya, sistem kontroller memanfaatkan nilai yang diperoleh dari CMUcam3.

Langkah-langkah yang harus dilakukan oleh sistem kontroller agar memperoleh nilai dari CMUcam3 adalah berkomunikasi dengan CMUcam3, memberikan perintah dengan urutan yang benar pada CMUcam3, dan selanjutnya menerima sekaligus menyimpan data dari CMUcam3. Kemudian langkah-langkah yang harus dilakukan oleh robot agar dapat menyelesaikan tugasnya adalah mengetahui warna dari targetnya, mencari tahu posisi targetnya, dan selanjutnya melakukan aksi untuk mendapatkan targetnya.

Hasil dari penelitian diuji dengan beberapa percobaan, percobaan-percobaan tersebut meliputi percobaan pengklasifikasian warna, percobaan penjejakan obyek berwarna, dan percobaan robot mencari dan mendapatkan targetnya. Hasil dari ketiga percobaan tersebut membuktikan bahwa CMUcam3 dapat digunakan untuk pengklasifikasian warna dan penjejakan obyek berwarna, sistem kontroller berhasil memperoleh nilai dari CMUcam3, dan robot dapat menyelesaikan tugasnya.

Kata kunci : Otomatis, humanoid, CMUcam3, sistem kontroller.

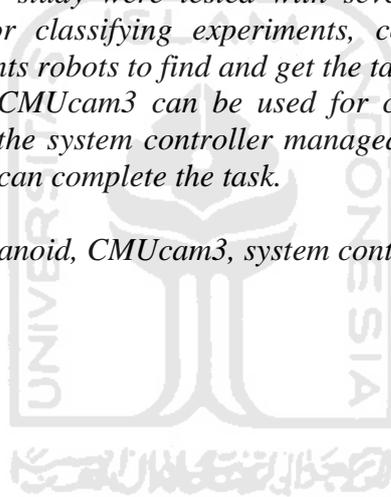
ABSTRACT

This final task is research on the ability of the camera sensor CMUcam3 to classify and track on colored objects, and the utilization of this capacity on two legs or robot humanoid robot to complete its task is to find and get a target in the form of colored objects. The study begins from figuring out how to control a robot or a system controller to obtain the value of CMUcam3, then the pursuit of a robot capable of completing tasks that can be done. Because to make a robot capable of completing the task, the system controller use the values obtained from CMUcam3.

The steps that must be done by the system controller in order to obtain the value of CMUcam3 is communicating with CMUcam3, giving orders in the correct order on CMUcam3, and then receive and store data from CMUcam3. Then the steps that must be performed by robots in order to accomplish its task was to determine the color of the target, find out the position of the target, and then take action to get the target.

The results of the study were tested with several experiments, these experiments include color classifying experiments, colored object tracking experiment, and experiments robots to find and get the target. Results from three experiments proved that CMUcam3 can be used for color classification and tracking colored objects, the system controller managed to obtain the value of CMUcam3, and the robot can complete the task.

keyword : Automatic, humanoid, CMUcam3, system controller.



BAB I

PENDAHULUAN

1.1 Latar Belakang

Keunggulan dalam teknologi robotik tak dapat dipungkiri telah lama dijadikan ikon kebanggaan negara-negara maju di dunia. Kecanggihan teknologi yang dimiliki, gedung-gedung tinggi yang mencakar langit, tingkat kesejahteraan rakyat yang baik, kota-kota yang *modern*, belum terasa lengkap tanpa popularitas kepiawaian dalam dunia robotika.

Robot adalah piranti yang bermanfaat secara umum, yang di dalamnya terdiri dari mekanik, elektronik, sistem controller, sistem gerak, aktuator, dilengkapi dengan sensor-sensor eksternal serta dapat diprogram ulang dan dapat melakukan pekerjaan beraneka ragam. Dengan demikian, dapat disimpulkan bahwa robot adalah suatu mekanisme yang secara otomatis dapat memproses suatu pekerjaan dalam memenuhi suatu permintaan tertentu terhadap suatu sistem.

Untuk perkembangan robotika saat ini dapat dibilang sangat pesat. Riset tidak hanya dilakukan pada robot beroda saja, namun sudah beranjak pada robot berkaki yang mengadopsi kaki hewan atau serangga, hingga robot *humanoid* yang mirip dengan manusia.

Kelebihan robot berkaki dibandingkan dengan robot beroda adalah kemampuan untuk dapat melewati lintasan yang tidak *continue*, misalnya anak tangga. Kelebihan yang lain adalah sedikitnya getaran yang ditimbulkan ketika robot berjalan, karena kaki dapat sekaligus berfungsi sebagai penyerap getaran. Meski demikian dalam perancangannya robot dua kaki atau lebih, jauh lebih sulit dibandingkan robot beroda ataupun berantai, karena mekanisme gerakan yang lebih rumit dan membutuhkan keseimbangan dalam gerakannya.

Sedangkan untuk membuat robot dua kaki atau robot *humanoid* jauh lebih sulit dibandingkan dengan robot yang menggunakan lebih dari dua kaki. Untuk mendapatkan mekanisme gerak dan keseimbangan yang baik pada robot dua kaki atau robot *humanoid*, maka mekanik yang presisi, simpel dan kesesuaian bentuk robot dengan anatomi tubuh manusia harus terpenuhi. Seluruh dimensi

bagian robot juga harus diperhitungkan, dari bagian kaki hingga kepala sampai pemilihan sensor yang efektif dan efisien dalam dimensi dan fungsinya.

Pemilihan sensor pada aplikasi robot merupakan suatu hal yang sangat penting, karena sensor tidak hanya sebagai elemen pembangun sistem yang mempunyai keterikatan yang sangat erat dengan fungsi dan lingkungan kerja robot, namun sensor memberi informasi mengenai apa yang terjadi pada lingkungan kerjanya, sehingga robot memiliki kemampuan secara otomatis untuk melakukan suatu pekerjaan atau tindakan.

Pada banyak aplikasi, sensor yang digunakan adalah sensor yang berupa rangkaian elektronik *analog/digital*. Sebagai contohnya untuk menunjang robot dua kaki atau robot *humanoid* lebih efektif dan efisien menggunakan sensor kamera, karena mempunyai bentuk dan dimensi yang tidak banyak mempengaruhi keseimbangan robot saat berjalan, dan mempunyai kemampuan untuk mengklasifikasi warna hingga menjejak obyek berwarna seperti mata pada manusia sebagai indra penglihatan.

Berdasarkan latar belakang di atas perlu dilakukan penelitian dengan fokus penggunaan sensor kamera untuk robot dua kaki atau robot *humanoid*. Yang nantinya dari proses penelitian tersebut diperoleh informasi yang sesungguhnya tentang sensor kamera, sehingga dari informasi tersebut, sensor kamera dapat benar-benar dimanfaatkan pada robot dua kaki atau robot *humanoid*.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas maka dapat diambil rumusan masalah sebagai berikut :

- a. Bagaimana agar robot dapat melakukan suatu pekerjaan atau tindakan secara otomatis.
- b. Seperti apakah sensor yang efektif dan efisien untuk robot dua kaki atau robot *humanoid*.
- c. Bagaimana cara menggunakan sensor kamera, agar dapat dimanfaatkan secara optimal sebagai sensor penglihatan pada robot dua kaki atau robot *humanoid*.

1.3 Batasan Masalah

Supaya tidak menyimpang dan lebih terarah dari obyek pembahasan maka diperlukan batasan-batasan penelitian yang difokuskan pada :

- a. Penelitian difokuskan pada Pengolahan citra dari sensor kamera yang berupa pengklasifikasian warna dan penjejukan obyek berwarna.
- b. Sensor kamera yang digunakan adalah CMUcam3.
- c. Sistem kontroller yang dipergunakan berbasis ATmega16.
- d. Robot dua kaki atau robot *humanoid* yang dipergunakan merupakan hasil penelitian sebelumnya (Endik, 2011 dan Sriyanto, 2011).

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk membuat robot dua kaki atau robot *humanoid* secara otonom mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna.

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah untuk memberikan kontribusi pada pengembangan ilmu pengetahuan dan teknologi khususnya teknologi robotika yang mungkin saja akan sangat berguna dimasa yang akan datang.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini terdiri dari beberapa bagian, yaitu pada bagian awal berisi halaman judul, lembar pengesahan dosen pembimbing, lembar pengesahan dosen penguji, halaman motto, kata pengantar, abstrak, daftar isi, daftar tabel dan daftar gambar. Kemudian untuk bagian isi terdiri dari beberapa bab, dan untuk bagian akhir berisi tentang daftar pustaka dan lampiran. Bab-bab yang terdapat pada bagian isi terdiri dari bab I sampai dengan bab V.

Bab I adalah pendahuluan, berisikan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan laporan tugas akhir.

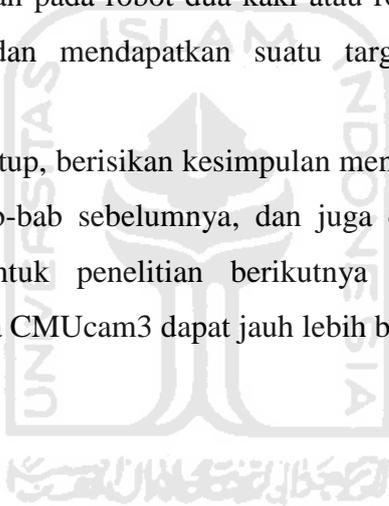
Bab II adalah landasan teori, berisikan teori-teori yang melandasi penelitian dan penyusunan laporan tugas akhir ini. Teori-teori tersebut

menjelaskan tentang mekanik, aktuator, sistem gerak, sistem kontroller dan sensor.

Bab III adalah metodologi penelitian, berisikan penjelasan tentang cara sistem kontroller memperoleh hingga pengolahan data hasil *image processing* yang dilakukan oleh sensor kamera CMUcam3 berupa pengklasifikasian warna dan penjejakan obyek berwarna, dan cara robot dua kaki atau robot *humanoid* untuk menyelesaikan tugasnya yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna.

Bab IV adalah hasil dan pembahasan, berisikan tentang percobaan dan pembuktian bahwa sensor kamera CMUcam3 dapat digunakan untuk mengklasifikasi warna dan menjejak obyek berwarna, dan sensor kamera CMUcam3 dapat digunakan pada robot dua kaki atau robot *humanoid*, sehingga robot mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna.

Bab V adalah penutup, berisikan kesimpulan mengenai semua uraian yang telah dijabarkan pada bab-bab sebelumnya, dan juga dilengkapi dengan saran yang bisa dilakukan untuk penelitian berikutnya agar pemanfaatan dan penggunaan sensor kamera CMUcam3 dapat jauh lebih baik dan optimal lagi.



BAB II

KAJIAN PUSTAKA

2.1 Definisi Robot

Robot adalah suatu alat mekanik yang dapat melakukan tugas fisik, baik menggunakan pengawasan dan kontrol dari manusia, ataupun menggunakan program yang telah didefinisikan terlebih dulu (kecerdasan buatan). Robot berasal dari bahasa Czech, *robota*, yang berarti pekerja, mulai menjadi populer ketika seorang penulis berbangsa Czech (ceko), Karl Capek, membuat pertunjukan dari lakon komedi yang ditulisnya pada tahun 1921 yang berjudul RUR (*Rossum's universal robot*). Ia bercerita tentang mesin yang menyerupai manusia, tetapi mampu bekerja terus-menerus tanpa lelah. Gaung popularitas istilah robot ini kemudian memperoleh sambutan dengan diperkenalkannya robot Jerman dalam film *Metropolis* tahun 1926 yang sempat dipamerkan dalam New York *World's Fair* 1939. Film ini mengisahkan tentang robot yang berjalan mirip manusia beserta hewan peliharaannya. Kembali atas jasa insan film, istilah robot ini makin populer dengan lahirnya C3PO dalam film *Star Wars* pertama pada tahun 1977.

Menurut Fu, et al. (1987), penelitian dan pengembangan pertama yang berbuah produk robotik dapat dilacak mulai dari tahun 1940-an ketika *Argonne National Laboratories* di Oak Ridge, Amerika, memperkenalkan sebuah mekanisme robotik yang dinamai *master-slave* manipulator. Robot ini digunakan untuk menangani material radioaktif. Kemudian produk robot komersial pertama diperkenalkan oleh *unimation incorporated*, Amerika, pada tahun 1950-an. Hingga belasan tahun kemudian langkah komersial ini diikuti oleh perusahaan-perusahaan lain di dunia.

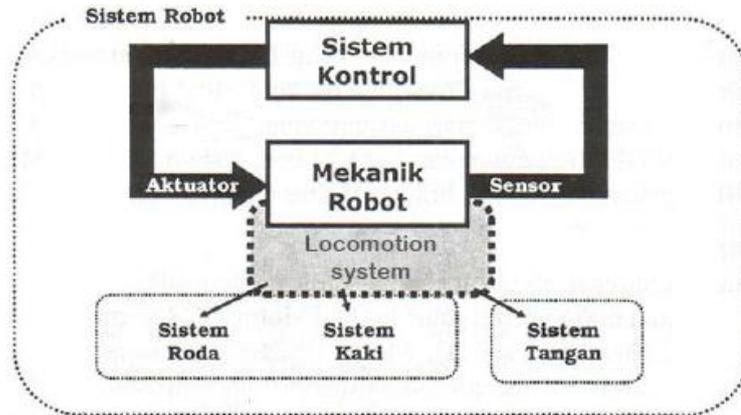
Baru setelah dunia mulai menapak ke jaman industri pada pertengahan tahun 60-an kebutuhan akan otomasi makin menjadi-jadi. Pada saat itulah negara-negara yang telah mapan saat itu seperti Amerika, Inggris, Jerman dan Perancis mulai memunculkan grup-grup riset yang menjadikan robotik sebagai temanya. Tak lama kemudian di Asia yang dimotori oleh pemikir-pemikir dari Jepang, terutama yang baru pulang menimba ilmu di Amerika, bermunculan pula

kelompok-kelompok peneliti dalam bidang robotik. Tak lama kemudian dari catatan bangsa Jepang-lah yang paling produktif dalam pengembangan teknologi robotik di dunia.

Pada awalnya aplikasi robot hampir tak dapat dipisahkan dengan dunia industri, maka kemudian muncul istilah *industrial robot* dan robot manipulator. Definisi robot industri adalah suatu robot tangan (*robot arm*) yang diciptakan untuk berbagai keperluan untuk meningkatkan produksi, memiliki lengan-lengan kaku yang berhubungan secara seri dan sendi yang dapat berputar (*rotasi*), memanjang dan memendek (*translasi* atau *prismatik*). Satu sisi lengan yang disebut sebagai pangkal ditanam pada bidang atau meja yang statis (tidak bergerak), sedangkan sisi yang lain disebut sebagai ujung (*end of effector*) dapat dimuati dengan tool yang sesuai dengan tugas robot. singkatnya dalam dunia mekanik robot manipulator ini memiliki dua bagian, yaitu tangan atau lengan (*arm*) dan pergelangan (*wrist*), pada pergelangan dapat dipasangkan berbagai tool. Begitu diminatinya penggunaan manipulator dalam industri ini, menyebabkan banyak perusahaan besar di dunia menjadikan robot industri sebagai unggulan, bahkan beberapa perusahaan di Jepang masih menjadikan manipulator sebagai produk utama.

Namun saat ini mungkin definisi robot industri sudah tidak sesuai lagi, karena teknologi robot juga sudah dipakai meluas sejak awal 80-an. Seiring itu pula kemudian muncul istilah robot *humanoid* (konstruksi mirip manusia), *animaloid* (mirip binatang), dan sebagainya. Bahkan kini dalam industri spesifik seperti industri perfilman, industri angkasa luar dan industri pertahanan, manipulator tersebut bisa jadi hanya menjadi bagian saja dari sistem robot secara keseluruhan (Pitowarno, 2006).

Artinya sistem robot saat ini tidak hanya tarbangun dari manipulator saja, namun sudah terdiri dari beberapa komponen, seperti yang terlihat pada gambar 2-1. Gambar 2-1 adalah ilustrasi dari sebuah sistem robot *modern*, yang tarbangun dari mekanik robot, aktuator, sistem gerak, sistem kontroller dan sensor.



Gambar 2-1 Sistem Robot

2.2 Mekanik Robot

Mekanik robot adalah sistem mekanik yang terdiri dari setidaknya-tidaknya sebuah fungsi gerak. Jumlah fungsi gerak disebut dengan derajat kebebasan atau *Degree of Freedom (DOF)*. Sebuah sendi yang diwakili oleh sebuah gerak aktuator disebut sebagai satu derajat kebebasan.

Struktur robot sebagian besar dibangun berdasarkan konstruksi mekanik. Robot yang memiliki kemampuan navigasi dan manipulasi, relatif memiliki konstruksi mekanik yang lebih rumit dibandingkan dengan yang berkemampuan navigasi saja, seperti *mobile* robot tanpa tangan yang hanya memiliki roda penggerak. Sedangkan robot berjalan (*walking* robot) seperti misalnya bi-ped (dua kaki) memiliki konstruksi mekanik yang lebih rumit dibandingkan dengan robot tangan planar.

Hal mendasar yang perlu diperhatikan dalam desain mekanik robot adalah perhitungan kebutuhan torsi untuk menggerakkan sendi atau roda. Motor sebagai penggerak utama (*prime-mover*) yang paling sering dipakai umumnya bekerja optimal jika torsi dan kecepatan putarnya ideal. Sebab kebanyakan gerakan yang diperlukan pada sisi anggota badan robot adalah relatif pelan namun bertenaga (Pitowarno, 2006). Maka dari itu konstruksi mekanik robot dan kebutuhan torsi penggerak mekanik robot harus benar-benar diperhitungkan untuk mendapatkan mekanik robot yang baik secara keseluruhan. Contoh dari mekanik robot dapat dilihat pada gambar 2-2.



Gambar 2-2 Robot Dua Kaki atau Robot *Humanoid*

Gambar 2-2 merupakan robot dua kaki atau robot *humanoid*. Robot dua kaki atau robot *humanoid* sering dikenal dengan biped robot, yaitu robot yang pergerakannya menggunakan sistem kaki. Robot dua kaki telah banyak dijadikan objek penelitian karena memiliki karakteristik yang hampir sama dengan sistem kaki pada manusia atau hewan yang memiliki 2 kaki. Berbeda dengan robot 4 atau 6 kaki, robot dua kaki lebih memperhitungkan keseimbangan dalam gerak karena pada saat perpindahannya salah satu kaki harus mampu menopang keseluruhan rangka robot.

2.3 Aktuator

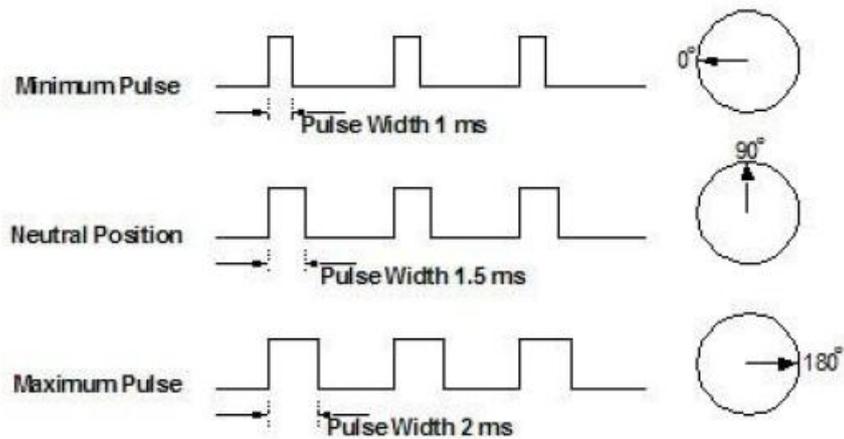
Aktuator adalah perangkat elektromekanik yang menghasilkan daya gerakan. Dapat dibuat dari sistem motor listrik (motor DC magnet permanen, brushless, shunt, series, motor DC servo, motor stepper, dan solenoid), sistem pneumatis (perangkat kompresi berbasis udara atau gas nitrogen) dan perangkat hidrolis (berbasis bahan cair seperti oli). Untuk meningkatkan tenaga mekanik aktuator atau torsi gerakan dapat dipasang *gearbox*, baik *system direct gear* atau *sprocket gear* (Pitowarno. 2006). Dengan meningkatkan kebutuhan akan tenaga dan torsi pada motor atau aktuator, maka dapat menunjang mekanik robot mampu bergerak jauh lebih baik dan sesuai apa yang diharapkan. Salah satu contoh dari aktuator yang banyak dipakai pada robot dapat dilihat pada gambar 2-3.



Gambar 2-3 Motor Servo

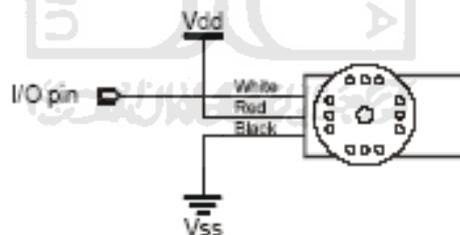
Gambar 2-2 adalah motor servo, motor servo merupakan sejenis motor DC yang beroperasi secara *closed loop*. Poros motor dihubungkan dengan rangkaian kendali, sehingga jika putaran poros belum sampai posisi yang diperintahkan maka rangkaian kendali akan terus mengoreksi posisi hingga mencapai posisi yang diperintahkan (Nugroho Adi, 2010).

Motor servo terdiri dari dua jenis yaitu motor servo standar yang hanya dapat bergerak pada sudut tertentu, biasanya 180° atau 270° , dan motor servo *continuous* yang dapat berputar secara *continue*. Motor servo mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan dapat dikendalikan hanya dengan memberikan pengaturan lebar pulsa pada bagian pin kontrollernya. Lebar pulsa yang diperlukan antara 1 ms hingga 2 ms, contohnya Jika diberikan pulsa dengan lebar 1.5 ms maka motor servo akan berputar menuju 90 derajat (posisi tengah = *middle*). Pulsa dengan lebar 2.0 ms akan membuat poros motor servo menuju 180 derajat (posisi kanan), sedangkan pulsa dengan lebar 1.0 ms akan membuat motor servo menuju 0 derajat (posisi kiri). Hubungan lebar pulsa dengan putaran arah motor servo dapat dilihat pada gambar 2-4.



Gambar 2-4 Hubungan Lebar Pulsa dengan Putaran Arah Motor Servo

Motor servo mempunyai putaran yang lambat, dimana biasanya ditunjukkan oleh rute putaran yang lambat, namun demikian memiliki torsi yang kuat karena internal *gear*-nya. Pada motor servo terdapat tiga pin yaitu pin sinyal, pin tegangan catu daya positif dan negative, susunan ketiga pin tersebut dapat dilihat pada gambar 2-5. Untuk beroperasi dengan baik motor servo sendiri memerlukan tegangan catu daya berkisar 4,8 hingga 6 V.



Gambar 2-5 Susunan Pin Motor Servo

2.4. Sistem Gerak atau *Locomotion System*

Sistem gerak adalah sistem konstruksi robot yang digunakan robot untuk dapat memanipulasi posisi dari satu titik ke titik yang lain. Menurut Pitowarno (2006) ada beberapa sistem gerak yang dapat digunakan dalam desain perancangan robot, yaitu sistem roda, sistem tangan dan sistem kaki.

Sistem roda adalah sistem mekanik yang dapat menggerakkan robot untuk berpindah posisi. Dapat terdiri dari sedikitnya sebuah roda penggerak (*drive* atau

steer), dua roda deferensial (kiri-kanan independen atau sistem belt seperti tank), tiga roda (*synchro drive* atau *holonomic system*), empat roda (*Ackermann model/mobile* seperti robot mobil) ataupun lebih.

Sistem tangan adalah bagian atau anggota badan robot selain sistem roda dan kaki. Dalam konteks *mobile robot*, bagian tangan ini dikenal sebagai manipulator, yaitu sistem gerak yang berfungsi untuk memanipulasi (memegang, mengambil, mengangkat, memindah dan mengolah) obyek. Sedangkan pada robot industri fungsi mengolah ini dapat berupa perputaran (memasang mur baut, *drilling*, dan *milling*), *tracking* (mengelas dan membubut) ataupun mengaduk (*control proses*). Sistem tangan memiliki bagian khusus yang disebut sebagai *gripper* atau *grasper* (pemegang). Untuk *grasper* yang didesain seperti jari tangan manusia, derajat kebebasannya dapat terdiri dari 16 DOF (3 DOF untuk jari kelingking, manis, tengah, telunjuk, dan 4 DOF untuk jari jempol), tidak termasuk gerakan polar pada sendi pergelangan.

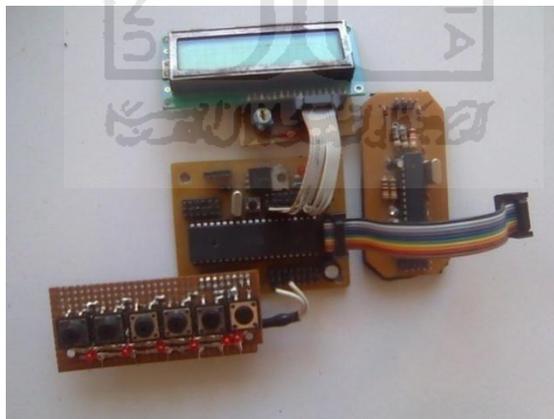
Sistem kaki pada dasarnya adalah gerakan roda yang didesain sedemikian rupa hingga memiliki kemampuan gerak seperti makhluk hidup. Robot dengan sistem dua kaki atau bi-ped robot memiliki struktur seperti kaki manusia setidaknya tidaknya memiliki sendi-sendi gerak yang mewakili pergelangan kaki lutut dan pinggul. Dalam konfigurasi yang ideal pergerakan pada pinggul dapat terdiri dari multi DOF dengan kemampuan gerakan memutar seperti orang menari jaipong. Demikian juga pada pergelangan kaki, idealnya juga memiliki kemampuan gerakan polar. Untuk robot binatang (*animaloid*) seperti serangga, jumlah kaki dapat didesain lebih dari empat. Bahkan robot ular dapat memiliki DOF yang lebih dari 8 sesuai panjang dari robot tersebut.

2.5 Sistem Kontroller

Sistem kontroller adalah kumpulan alat-alat atau setidaknya terdiri dari rangkaian prosesor, untuk mengendalikan, memerintahkan dan mengatur keadaan dari suatu sistem. Bila diperlukan sistem kontroller dapat dilengkapi dengan sistem monitor seperti *seven segment*, LCD (*liquid crystal display*) ataupun CRT (*cathode ray-tube*).

Sedangkan prosesor itu sendiri dapat berupa *microcontroller*. *Microcontroller* adalah sebuah komputer mini yang keseluruhan sistemnya berada dalam satu *chip*. Komponen *Microcontroller* terdiri dari CPU(*Central Processing Unit*) yang bertugas untuk memproses data, RAM (*Random Access Memory*) sebagai tempat menyimpan instruksi yang dijalankan dan data-data yang diperlukan, ROM (*Read-Only Memory*) sebagai tempat dimana program yang sebenarnya disimpan, jalur I/O (*Input/Output*) adalah salah satu jalur yang digunakan *microcontroller* untuk berkomunikasi secara sederhana dengan peralatan lain seperti sensor, LED, toggle switch dan lain-lain, timer untuk *interrupt* berdasarkan waktu, jalur antar muka (*serial* maupun *parallel*) dan terkadang terdapat ADC (*Analogto-Digital Converter*) atau DAC (*Digital-to-Analog Converter*).

Microcontroller banyak digunakan pada sistem kontroller karena mudah diaplikasikan ke peralatan-peralatan yang dimana ruang penempatannya sangat terbatas seperti robot. Contoh *microcontroller* yang sering digunakan dalam sistem kontroller adalah ATmega16. Bentuk dari sistem kontroller berbasis ATmega16 dapat dilihat pada gambar 2-6 di bawah ini.



Gambar 2-6 Sistem Kontroller Berbasis Mikrokontroller ATmega16

Dalam aplikasi robot mikrokontroler tidak hanya berperan sebagai otak, namun mikrokontroler juga sering difungsikan untuk memperoleh suatu data dengan cara berkomunikasi dengan piranti lain. Untuk dapat berkomunikasi dengan piranti lain contohnya mikrokontroller dengan komputer, mikrokontroller dengan mikrokontroller atau dengan yang lain. Mikrokontroller dilengkapi

dengan fasilitas komunikasi antar muka, ada 2 jenis fasilitas komunikasi antar muka yang dikenal, yaitu komunikasi *parallel* dan komunikasi *serial*. Sesuai dengan namanya pada komunikasi *parallel* transfer data dilakukan secara serempak/bersamaan, sedangkan pada komunikasi *serial* data dikirim secara bergantian. Komunikasi secara *parallel* memiliki kelebihan pada kecepatan transfer data, tetapi kualitas suatu komunikasi tidak hanya ditentukan oleh kecepatannya saja, ada faktor lain yang perlu diperhatikan yaitu jarak dan kepraktisan. Komunikasi *parallel* memerlukan jalur data yang lebih banyak, yang berarti pengkabelan (*wiring*) juga akan semakin banyak. Pada komunikasi *serial* biasanya hanya dibutuhkan 2 sampai 3 kabel saja, jadi bisa dikatakan komunikasi *serial* lebih praktis dibanding komunikasi *parallel* apalagi jika komunikasi dilakukan dengan jarak yang jauh.

Untuk melakukan komunikasi *serial* dapat menggunakan *Serial Electrical Standard* berupa RS-232, Secara deskriptif RS-232 adalah sebuah IC *transmitter* dan *receiver interface* yang membutuhkan tegangan +5v. Memiliki 2 pasang pin untuk *transmitter*, serta 2 pasang pin untuk *receiver*. RS-232 juga memiliki fungsi mengubah tegangan menjadi tegangan TTL yang dibutuhkan oleh mikrokontroler dalam pentransferan data *serial*.

Sedangkan pada mikrokontroler, agar komunikasi *serial* dapat berjalan dengan baik dibutuhkan suatu *protocol*/aturan komunikasi. Pada sistem kontroler berbasis ATmega16 terdapat beberapa protokol komunikasi *serial*, salah satunya adalah USART (*universal synchronous and asynchronous serial receiver and transmitter*). USART adalah piranti komunikasi *serial* yang mempunyai kefleksibelan tinggi dan mempunyai fitur sebagai berikut :

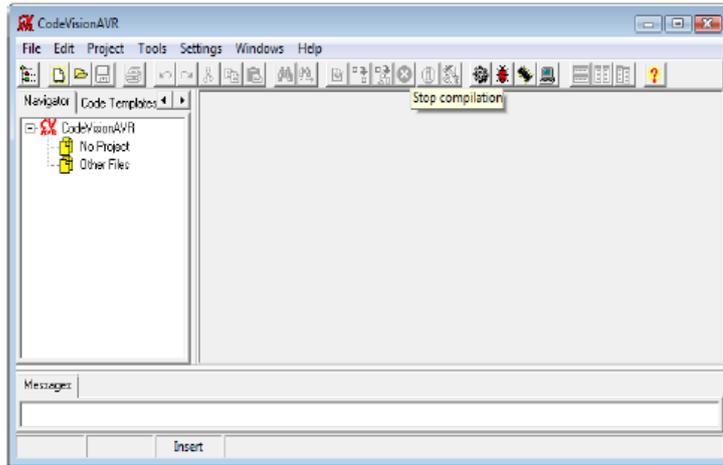
- a. Operasi *full duplex* (*register* pengiriman dan penerimaan data *serial* yang independen).
- b. Mode sinkron dan asikron.
- c. Operasi sinkron yang di-*clock* dari *master* atau *slave*.
- d. Baud rate generator resolusi tinggi.
- e. Mendukung *frame serial* dengan 5,6,7,8, or 9 bit data dan 1 atau 2 bit *stop*.

- f. Pembangkitan parity ganjil atau genap.
- g. Deteksi data *over run*.
- h. Deteksi *framing error*.
- i. Peredaman noise termasuk pendeteksi kesalahan bit *start* dan digital *low pass filter*.
- j. 3 buah interupsi yang terpisah pada *TX complete*, *TX data register empty* dan *RX complete*.
- k. Mode komunikasi multi-processor.
- l. Mode komunikasi asinkron kecepatan ganda.

Dengan fitur di atas memungkinkan sistem kontroller dapat berkomunikasi dengan piranti lain, sehingga dengan terjadinya komunikasi tersebut, maka akan diperolehnya data-data yang diperlukan oleh sistem kontroller untuk menjalankan seluruh sistem contohnya sistem yang ada pada robot agar berjalan dengan baik sesuai perintah yang telah diberikan.

Sedangkan pemrograman mikrokontroller ATmega16 dapat menggunakan *low level language (assembly)* atau *high level language* contohnya bahasa C, Basic, Pascal, JAVA dan lain-lain tergantung *compiler* yang digunakan.

Untuk pemrograman yang sering dilakukan, bahasa yang digunakan adalah bahasa C, dan untuk *software compiler*-nya memakai Code Vision AVR, tampilan dari Code Vision AVR dapat dilihat pada gambar 2-7. Alasan memakai bahasa C dikarenakan bahasa tersebut memiliki keunggulan dibanding bahasa *assembler* (bahasa mesin), yaitu *independent* terhadap *hardware* serta lebih mudah untuk menangani *project* yang besar. Bahasa C juga memiliki keuntungan-keuntungan yang dimiliki bahasa *assembler*, hampir semua operasi yang dapat dilakukan oleh bahasa *assembler* dapat dilakukan dengan bahasa C, bahkan dengan penyusunan program yang lebih sederhana dan mudah.



Gambar 2-7 Tampilan Code Vision AVR

2.6 Sensor

Sensor adalah alat penginderaan pada robot. Sensor juga disebut sebagai sistem pengukuran yang menerima sinyal masukan berupa parameter/besaran fisik dan mengubahnya menjadi sinyal/besaran lain yang dapat diproses lebih lanjut untuk nantinya ditampilkan, direkam ataupun sebagai sinyal umpan balik pada sistem kendali/kontroller. kebanyakan sensor mengubah parameter fisik menjadi elektrik, misalnya tegangan atau arus, sehingga sensor sering juga disebut sebagai transduser, yang berarti piranti pengubah energi dari satu bentuk ke bentuk lain. Piranti ini memberi informasi kepada sistem kontroller mengenai apa yang terjadi di lingkungan, besaran fisik yang diukur antara lain posisi, jarak, gaya, regangan, temperatur, getaran, akselerasi, cahaya, suara dan magnet (Adi, 2010). Sensor dapat diklasifikasikan berdasarkan *outputnya*, yaitu :

- a. *Output biner*: berupa 0 (0 V) atau 1 (5V).
- b. *Output analog*: misalnya 0V hingga 5V.
- c. *Output pewaktu*: misal PWM, waktu RC, waktu pantul.
- d. *Output serial*: misal UART (RS232), I2C, SPI, 1 ware, 2 ware, serial sinkron atau asinkron.
- e. *Output paralel*.

Salah satu contoh dari sensor yang sering digunakan pada aplikasi robot adalah sensor kamera CMUcam3, wujud dari sensor tersebut dapat dilihat pada gambar 2-8.



Gambar 2-8 Sensor Camera CMUcam3

Penggunaan kamera (digital) dalam dunia robotik dikenal sebagai *robotics vision*. Kamera itu sendiri digunakan untuk menangkap gambar 2 dimensi, yang kemudian dari gambar tersebut diolah untuk mendapatkan suatu informasi seperti nilai RGB (*red green blue*) atau nilai posisi dari suatu warna.

Kemampuan kamera digital biasanya diukur dari resolusi tangkapan gambarnya dalam pixels/inch atau pixels/cm. makin besar resolusinya maka makin akurat hasil tangkapan gambarnya, karena kamera digital sekarang ini telah mampu menghasilkan gambar beresolusi hingga 5 Mega pixels lebih. Untuk kegunaan fotografi resolusi ini sangat penting, namun untuk aplikasi *control* dalam *robotic vision*, resolusi yang semakin besar justru membuat kecepatan akses kontroller menjadi menurun. Semakin tinggi resolusinya maka semakin besar data gambar yang harus diidentifikasi dan diolah oleh kontroller, sehingga program akan bekerja lebih lama.

CMUCam3 merupakan sensor visual atau penglihatan yang mampu melakukan pengolahan suatu citra. CMUcam3 memiliki banyak fitur, fitur-fitur tersebut adalah mikroprosesor utama Philips LPC2106, modul kamera omnivision CMOS OV6620 atau OV7630 dengan resolusi 352 x 388 pixels, komunikasi menggunakan RS-232 atau *serial port TTL*, slot tambahan untuk MMC, empat pin *kontroller servo*, jalur antarmuka *wareless*, analog video *output PAL* atau *NTSC*, *programmable*, *open source* untuk windows atau linux, kecepatan *image processing* 26 fps (frame/second), kompresi JPEG *software*, *lua light-weight language interpreter*, *raw images dumps* melalui port *serial*, mampu

membuat histogram dan FIFO *image buffer* untuk *image processing* dengan resolusi tinggi.

Kegunaan yang paling utama sensor ini adalah kemampuannya untuk memonitor warna. Semua itu dapat dicapai pada warna yang mempunyai nilai kontras dan intensitas yang tinggi. Sebagai contoh, CMUcam3 dapat dengan mudah memonitor bola yang berwarna merah pada *background* putih tetapi akan sulit untuk memonitor benda berwarna coklat pada pencahayaan yang tidak tepat.

Pengolahan Citra sering disebut *image processing*, *image* atau citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat *analog* berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat *digital* yang dapat langsung disimpan pada suatu pita magnetik. Citra *digital* tidak selalu merupakan hasil langsung data rekaman suatu sistem. Terkadang hasil rekaman data bersifat *continue* seperti gambar pada monitor televisi, foto sinar-X, dan lain sebagainya. Dengan demikian untuk memperoleh data dari suatu citra *digital* diperlukan suatu proses konversi agar selanjutnya dapat diproses lebih lanjut.

Ada perintah-perintah yang dapat diberikan kepada sensor kamera CMUcam3 untuk melakukan suatu pekerjaan, seperti contohnya *image processing*. Keseluruhan perintah tersebut dapat dilihat pada tabel 2-1 sampai 2-11. Contoh hasil dari pemberian perintah khususnya untuk *image processing*, dapat dilihat pada tabel 2-12. Tabel 2-12 merupakan macam-macam paket data yang di dalamnya mengandung informasi tertentu dari suatu keadaan yang ditangkap oleh sensor kamera CMUcam3.

Tabel 2-1 Buffer Commands

BM	Buffer Mode
RF	Read Frame

Tabel 2-2 Color Statistics Commands

GM	Get Mean
LM	Line Mode

Tabel 2-3 Windowing Commands

SF	Send Frame
DS	Down Sample
VW	Virtual Window
FS	Frame Stream
HR	HiRes Mode
GW	Get Window
PD	Pixels Difference

Tabel 2-4 Servo Commands

SV	Servo Position
SP	Servo Parameters
GS	Get Servo Position
SM	Servo Mask
SO	Servo Output

Tabel 2-5 Camera Module Commands

CR	Camera Register
CP	Camera Power
CT	Camera Type

Tabel 2-6 Auxiliary I/O Commands

GB	Get Button
GI	Get Auxiliary I/O
L0(1)	LED control

Tabel 2-7 Histogram Commands

GH	Get Histogram
HC	Histogram Config
HT	Histogram Track

Tabel 2-8 System Level Commands

SD	Sleep Deeply
SL	Sleep
RS	Reset
GV	Get Version

Tabel 2-9 Color Tracking Commands

TC	Track Color
TI	Track Inverted
TW	Track Window
NF	Noise Filter
LM	Line Mode
GT	Get Tracking Parameters
ST	Set Tracking Parameters

Tabel 2-10 Frame Differencing Commands

FD	Frame Difference
DC	Difference Channel
LF	Load Frame
MD	Mask Difference
UD	Upload Difference
HD	HiRes Difference
LM	Line Mode

Tabel 2-11 Data Rate Commands

DM	Delay Mode
PM	Poll Mode
PS	Packet Skip
RM	Raw Mode
PF	Packet Filter
OM	Output Packet Mask

Tabel 2-12 Tipe Paket Data

Paket Data	Data yang Diterima
F	1 2 r g b r g b ... r g b r g b 2 r g b r g b ... r g b r g b 3
H	H bin0 bin1 bin2 bin3 ... bin26 bin27 \r
T	T mx my x1 y1 x2 y2 pixels confidence\r
S	S Rmean Gmean Bmean Rdeviation Gdeviation Bdeviation \r

Contoh format pengiriman perintah kepada sensor kamera CMUcam3 untuk melakukan suatu pekerjaan seperti contohnya *image processing*, dapat dilakukan dengan format sebagai berikut :

TC [Rmin Rmax Gmin Gmax Bmin Bmax] \r

Kemudian paket data yang akan diterima dari sensor kamera CMUcam3 adalah sebagai berikut :

T mx my x1 y1 x2 y2 pixels *confidence*\r

Perintah yang diberikan kepada sensor kamera CMUcam3 diatas adalah TC (track Color), dimana data yang diperoleh dari perintah tersebut merupakan paket data tipe T, tipe paket data tersebut dapat digunakan untuk memonitor suatu obyek berwarna atau mengandung informasi tertentu, seperti di bawah ini:.

mx – Titik koordinat X dari obyek tampak di kamera.

my – Titik koordinat Y dari obyek tampak di kamera.

x1 – Titik koordinat X paling kiri dari tampak obyek di kamera.

y1 – Titik koordinat Y paling kiri dari tampak obyek di kamera.

x2 – Titik koordinat X paling kanan dari tampak obyek di kamera.

y2 - Titik koordinat Y paling kanan dari tampak obyek di kamera.

pixels – Jumlah pixels di dalam area x1y1 sampai dengan x2y2.

confidence – Kepercayaan terhadap suatu warna.

Dengan memberikan suatu perintah kepada sensor kamera CMUcam3, khususnya perintah untuk melakukan *image processing*. Maka sensor kamera CMUcam3 akan dapat benar-benar dimanfaatkan sebagai sensor penglihatan seperti mata pada manusia.

BAB III

METODOLOGI PENELITIAN

3.1 Pedahuluan

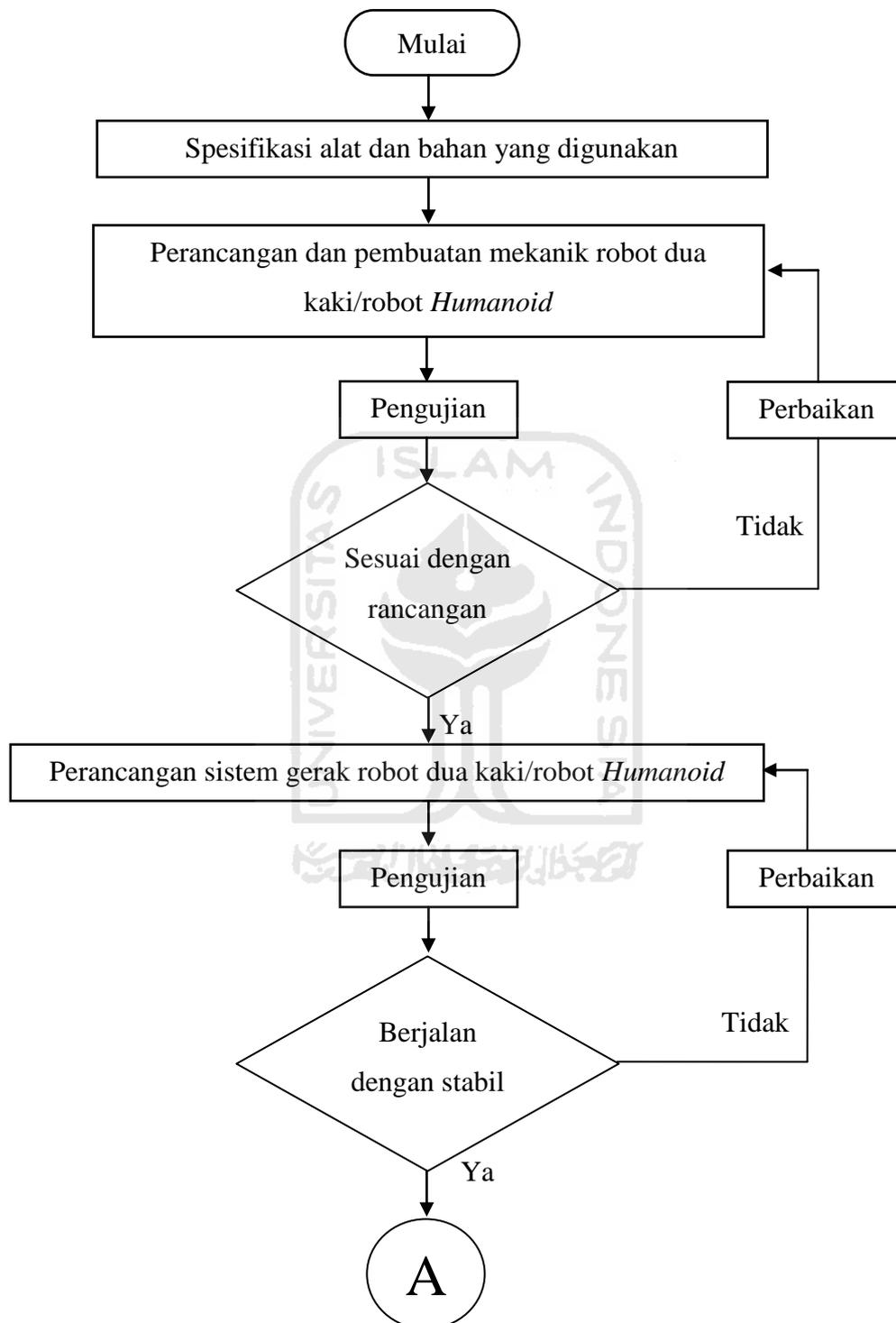
Pada penelitian ini sensor kamera yang dipergunakan adalah CMUcam3, kemudian robot dua kaki atau robot *humanoid* yang dipergunakan untuk mengaplikasikan sensor kamera CMUcam3 merupakan hasil penelitian sebelumnya (Endik, 2011), yang mana sistem gerak dari robot tersebut merupakan hasil penelitian sebelumnya juga (Sriyanto, 2011).

Dari pernyataan di atas, secara tidak langsung memperlihatkan bahwa satu rancangan dengan rancangan yang lain atau satu penelitian dengan penelitian yang lain saling berkaitan. Hal tersebut ditunjukkan ketika melakukan perancangan mekanik robot harus sesuai dengan sistem geraknya. Ataupun setelah kesempurnaan mekanik robot beserta sistem geraknya terpenuhi, maka belum lengkap tanpa sensor yang *compatible*. Atau dapat disimpulkan bahwa robot dua kaki atau robot *humanoid* yang mempunyai mekanik dan sistem gerak yang baik, belum sempurna jika tanpa sensor yang dapat menunjang fungsi keseluruhan dari robot tersebut. *Flowchat* kaitan keseluruhan penelitian dapat dilihat pada gambar 3-1.

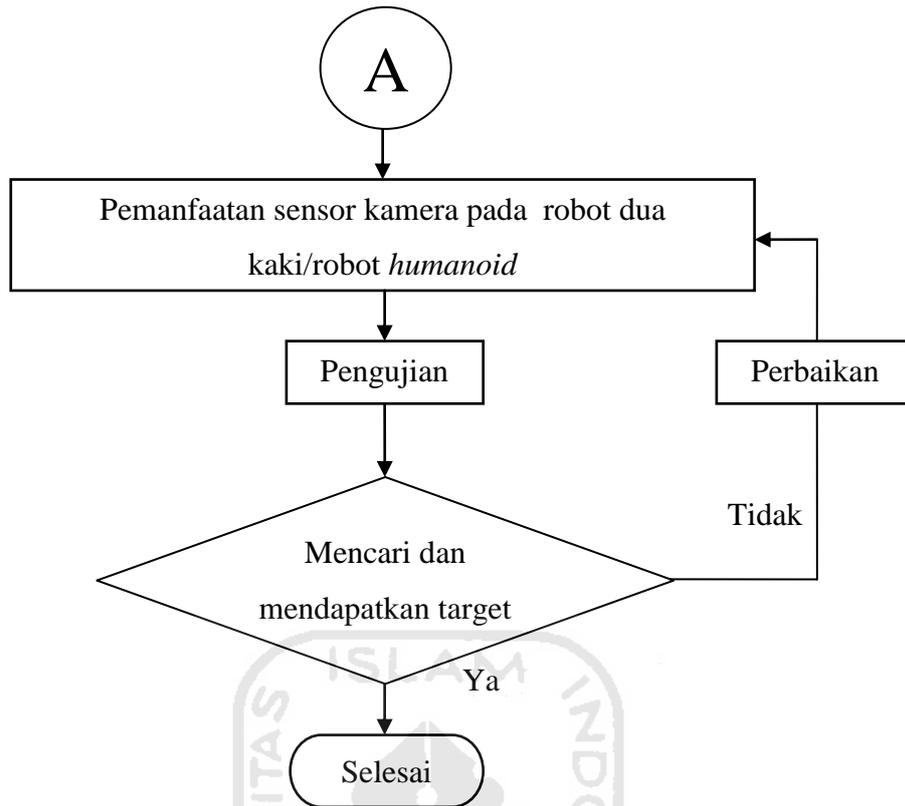
Pada robot dua kaki atau robot *humanoid*, sensor kamera CMUcam3 difungsikan sebagai sensor penglihatan seperti mata pada manusia. Karena sensor kamera CMUcam3 mempunyai kemampuan untuk mengklasifikasi warna, mengikuti atau menjejak obyek berwarna, mengetahui keberadaan obyek berwarna tersebut dan kemudian mendefinisikannya menjadi suatu posisi.

Melihat fungsi dari sensor kamera CMUcam3 dan kaitannya dengan tujuan penelitian ini, yaitu untuk membuat robot dua kaki atau robot *humanoid* secara otonom mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Maka dari itu penelitian ini difokuskan untuk mengetahui cara penggunaan sensor kamera CMUcam3 untuk melakukan pengolahan citra yang berupa pengklasifikasian warna dan penjejakan obyek berwarna. Sehingga dari hasil penelitian, nantinya sensor kamera CMUcam3 dapat benar-benar

dimanfaatkan pada robot dua kaki atau robot *humanoid*, dan sekaligus tujuan dari penelitian dapat tercapai dengan baik.



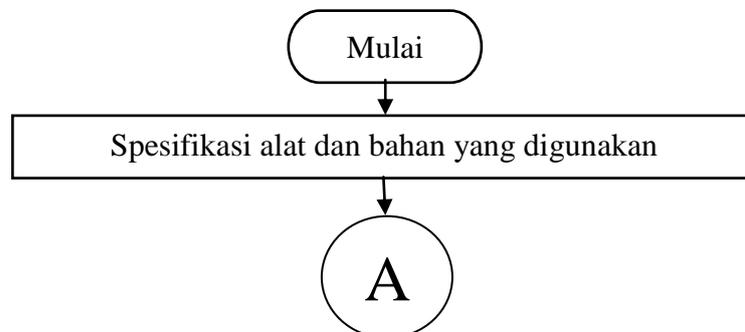
Gambar 3-1 Flowchat Kaitan Keseluruhan Penelitian



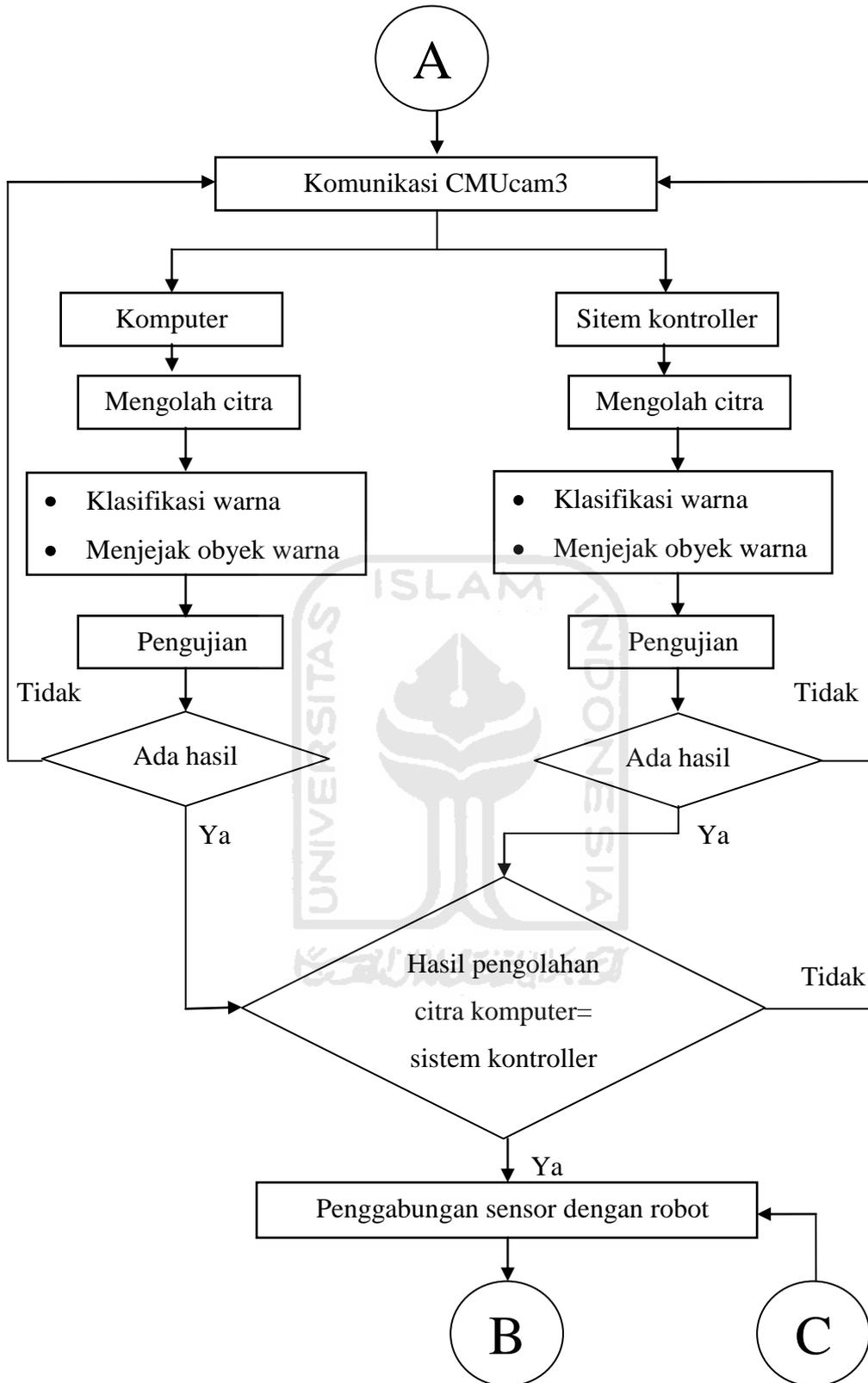
Gambar 3-1 *Flowchat* Kaitan Keseluruhan Penelitian (lanjutan)

3.2 *Flowchat* Penelitian

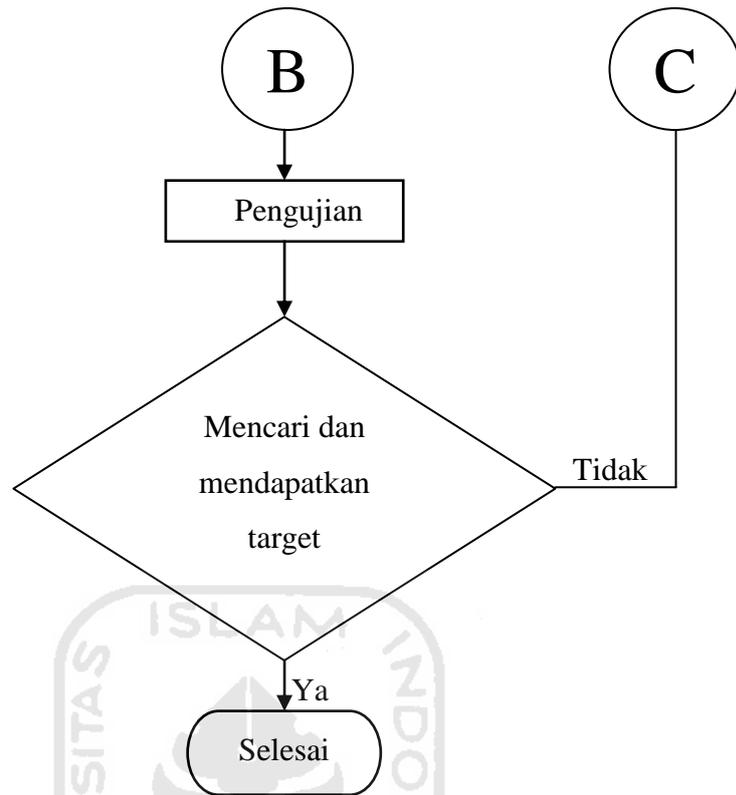
Flowchart penelitian ini berisi tentang langkah-langkah yang dilakukan saat penelitian. Langkah-langkah tersebut berfungsi untuk mempermudah penelitian, sehingga semua tujuan dari penelitian dapat terwujud. Langkah-langkah tersebut dapat dilihat pada gambar 3-2 di bawah ini.



Gambar 3-2 *Flowchart* Penelitian



Gambar 3-2 *Flowchart* Penelitian (lanjutan)



Gambar 3-2 *Flowchart* Penelitian (lanjutan)

3.3 Spesifikasi Alat dan Bahan yang Digunakan

Alat-alat yang digunakan dalam penelitian ini adalah.

- a. Sensor kamera CMUcam3.
- b. Sistem kontroller berbasis ATmega16 dengan *clock* 8 Mhz dan terdapat LCD 16 karakter.
- c. USB ISP *downloader* berbasis ATmega8.
- d. Perangkat komputer.
- e. Mekanik robot dua kaki atau robot *humanoid* dengan aktuator motor servo *standard*.
- f. Catu daya 5 V sampai 12 V (baterai dan adaptor).
- g. Rangkaian 6 saklar *reset*.
- h. Solder, obeng, tank, jangka sorong, penggaris, bor tangan, *cuter*, atraktor, multimeter, mur baut, lem tembak, gergaji, konektor, *black housing*.

Bahan-bahan yang digunakan dalam penelitian ini adalah.

- a. Tenol.
- b. PCB (*project circuit board*) polos.
- c. Kabel.
- d. Pelarut PCB.
- e. PCB berlubang.
- f. *Tubing*.
- g. Lem.

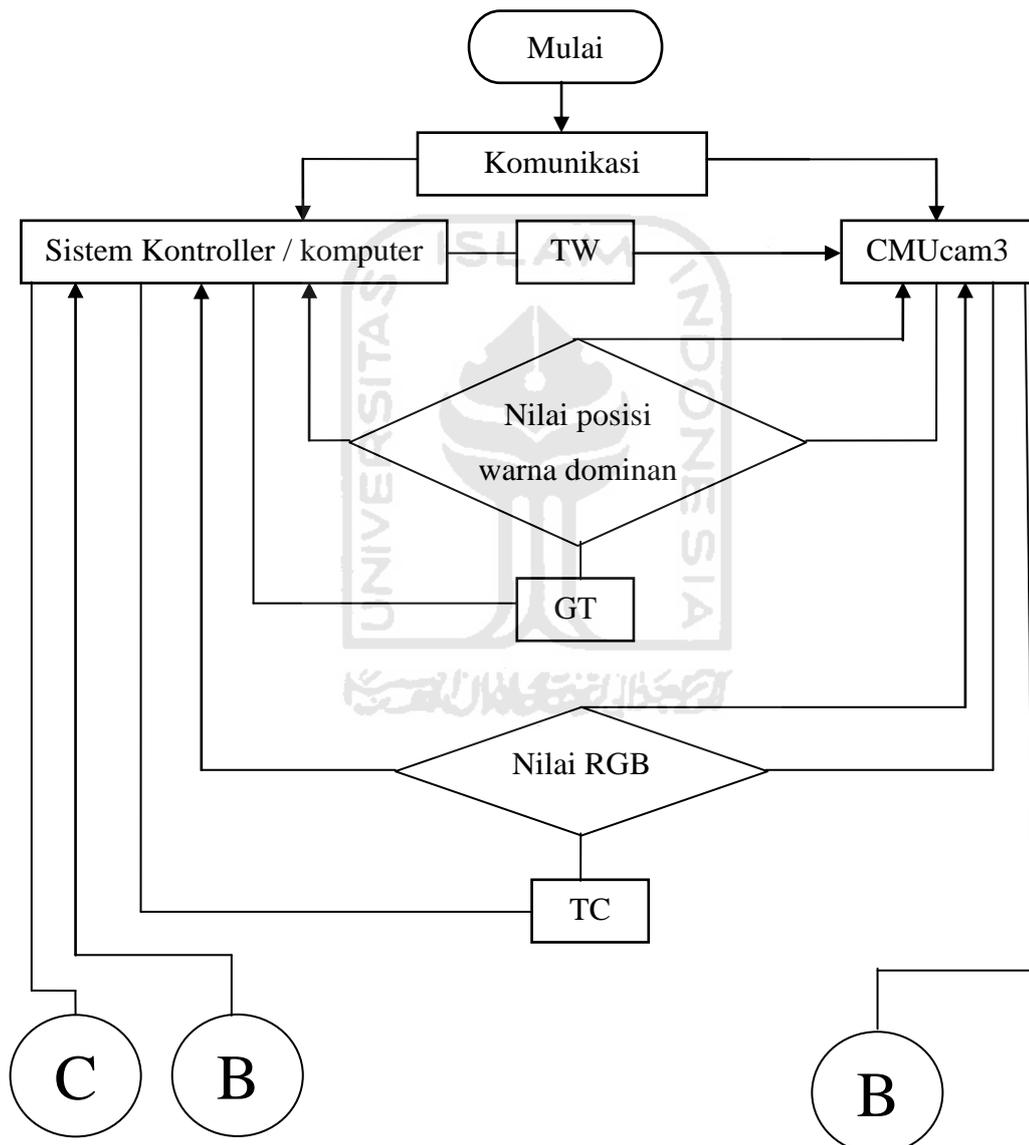
4.4 Komunikasi CMUcam3

Dalam penelitian ini sensor kamera CMUcam3 dikomunikasikan pada 2 piranti, yaitu komputer dan sistem controller yang berbasis ATmega16. Komunikasi sensor kamera CMUcam3 dengan komputer, digunakan untuk melihat hasil kerja dari sensor yang berupa pengolahan citra, dan membandingkan data hasil pengolahan citra yang diperoleh oleh komputer, dengan data hasil pengolahan citra yang diperoleh oleh sistem controller. Membandingkan data hasil pengolahan citra yang diperoleh antara kedua piranti tersebut, dimaksudkan untuk mengoreksi data hasil pengolahan citra yang diterima dan disimpan oleh sistem controller, apakah sudah benar atau tidak.

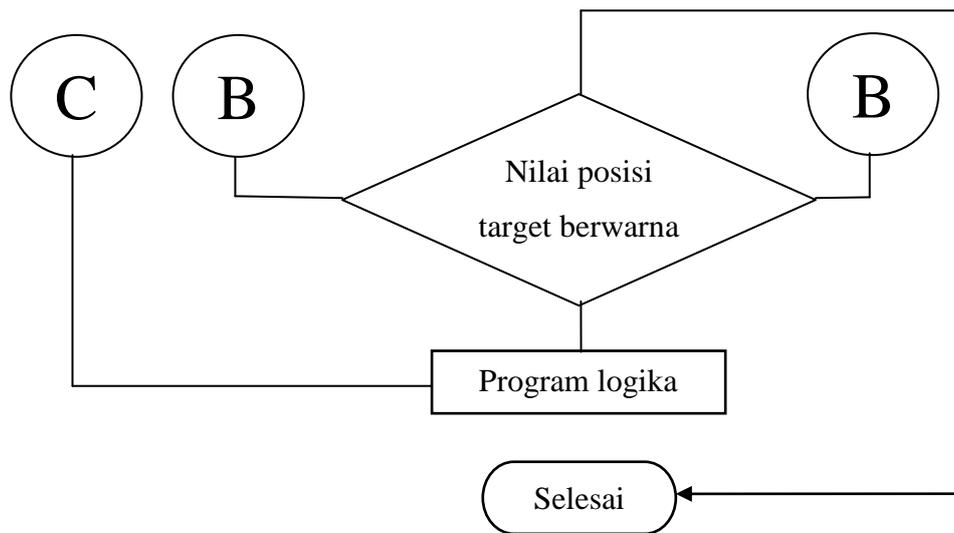
Sedangkan komunikasi sensor kamera CMUcam3 dengan sistem controller, digunakan untuk menerima dan menyimpan data hasil pengolahan citra yang dikirim oleh sensor kamera CMUcam3, yang berupa pengklasifikasian warna dan penjejukan obyek berwarna. Hasil dari pengklasifikasian warna merupakan nilai RGB (*red green blue*) dari suatu warna, dan hasil dari penjejukan obyek berwarna merupakan nilai posisi dari suatu obyek berwarna. Kemudian dari kedua data tersebut, dapat dijadikan informasi oleh sistem controller untuk mengendalikan seluruh sistem yang ada pada robot, dan memberikan intruksi agar robot mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna.

Seperti yang telah dijelaskan pada bab sebelumnya, ada perintah-perintah yang dapat diberikan pada sensor kamera CMUcam3 agar melakukan *Image Processing* atau pengolahan citra. Namun untuk memperoleh hasil pengolahan

citra yang benar, sesuai dengan batasan masalah yaitu pengklasifikasian warna dan penjejukan obyek berwarna. Maka perintah yang diberikan kepada sensor kamera CMUcam3 harus benar dan sesuai dengan urutan. Harus sesuai dengan urutan dikarenakan satu perintah dengan perintah lain saling berkaitan. Urutan dari perintah-perintah tersebut dapat dilihat pada gambar 3-3. Dalam penelitian ini yang dimaksudkan perintah pengklasifikasian warna adalah GT (*get tracking parameters*) dan perintah penjejukan obyek berwarna adalah TC (*track color*).



Gambar 3-3 Urutan Perintah Pengklasifikasian Warna dan Penjejukan Obyek Berwarna



Gambar 3-3 Urutan Perintah Pengklasifikasian Warna dan Penjejakan Obyek Berwarna (lanjutan)

Dari gambar 3-3, kaitan antara perintah satu dengan perintah lain ditunjukkan ketika hasil dari perintah TW (*track window*) yang berupa nilai posisi warna dominan yang ditangkap oleh sensor kamera CMUcam3, digunakan sebagai nilai referensi untuk melakukan perintah GT. Kemudian hasil dari perintah GT yang berupa nilai RGB, digunakan sebagai nilai referensi untuk melakukan perintah TC. Dan sampai akhirnya dari perintah TC, diperoleh nilai posisi dari suatu obyek berwarna, yang selanjutnya dari nilai tersebut dapat digunakan sebagai nilai logika bagi sistem controller untuk mengendalikan robot, sehingga mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna.

Urutan perintah pada gambar 3-3 diatas, secara umum digunakan untuk memperoleh data hasil pengolahan citra dari sensor kamera CMUcam3 yang berupa pengklasifikasian warna dan penjejakan obyek berwarna, ketika berkomunikasi dengan komputer maupun dengan sistem controller.

4.4.1 Komunikasi CMUcam3 dengan Komputer

Yang dimaksud komunikasi CMUcam3 dengan komputer adalah, mengkomunikasikan antara sensor kamera CMUcam3 dengan komputer untuk melihat hasil pengolahan citra dari sensor kamera CMUcam3, yang berupa pengklasifikasian warna dan penjejakan obyek berwarna, dengan menggunakan beberapa aplikasi berbentuk *software* yang sudah tersedia untuk komputer. *Software* tersebut adalah *serial watcher*, hiperterminal, CMUcam2GUI dan CMUcam3 *frame grabber*.

Contoh dari komunikasi CMUcam3 dengan komputer, dapat dilihat pada gambar 3-5. Pada gambar tersebut, *software* yang digunakan adalah hiperterminal. Perintah yang digunakan sehingga memperoleh nilai yang tertera pada gambar tersebut adalah TC (*track color*), atau dalam penelitian ini disebut sebagai perintah penjejakan obyek berwarna. Yang mana hasil dari perintah tersebut dapat diartikan sebagai nilai koordinat dari posisi suatu obyek berwarna. Namun sebelum memberi perintah TC, maka terlebih dahulu harus mempunyai target warna yang berbentuk nilai RGB sebagai referensi untuk melakukan perintah tersebut.

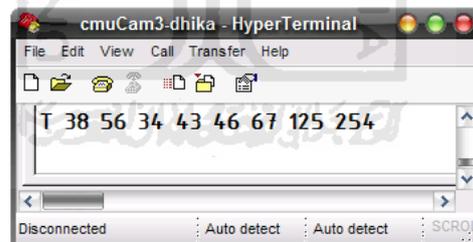
Nilai RGB dapat dicari dengan menggunakan aplikasi *software* adobe photoshop atau tetap menggunakan hiperterminal dengan perintah GT, atau dalam penelitian ini disebut sebagai perintah pengklasifikasian warna. Secara urutan, perintah GT harus diawali dengan Perintah TW. Setelah perintah TW dikirim, maka selanjutnya akan diperoleh nilai posisi warna dominan yang ditangkap oleh sensor kamera CMUcam3. Yang nantinya nilai dari perintah TW tersebut dijadikan nilai referensi untuk melakukan perintah GT. Hasil dari perintah GT adalah paket data S, paket data tersebut adalah paket data yang mengandung informasi nilai Rmin Gmin Bmin Rmax Gmax Bmax dari suatu warna.

Contoh selengkapnya ketika perintah GT dijalankan untuk memperoleh nilai RGB dari suatu warna misalkan warna oranye, dapat dilihat pada gambar 3-4. Dari gambar 3-4, nilai RGB yang diperoleh adalah Rmin =133, Gmin=21, Bmin =16, Rmax =193, Gmax =81, Bmax =46. Dan nilai RGB itulah yang dijadikan referensi untuk melakukan perintah TC.

```
:TW
ACK
T 44 70 0 0 87 142 254 254
:GT
ACK
133 21 16 193 81 46
:
```

Gambar 3-4 Contoh Perintah GT

Seperti yang telah dijelaskan pada bab sebelumnya, nilai yang diperoleh dari perintah TC adalah paket data T yang mengandung informasi nilai mx my $x1$ $y1$ $x2$ $y2$ $pixels$ $confidence$. Contoh hasil data yang diperoleh dari perintah TC terhadap benda berwarna misalkan oranye, dapat dilihat pada gambar 3-5. Nilai yang diperoleh dari perintah tersebut adalah $mx=38$, $my=56$, $x1=34$, $y1=43$, $x2=46$, $y2=67$, $pixels=125$, $confidence=254$. Nilai tersebut dapat diartikan sebagai nilai koordinat dari posisi suatu obyek berwarna yang tangkap oleh sensor kamera CMUcam3. Untuk lebih jelasnya maksud dari nilai koordinat, dapat dilihat pada gambar 3-7.



Gambar 3-5 Nilai Hasil Perintah TC Terhadap Obyek Berwarna *Orange* pada Hiperterminal

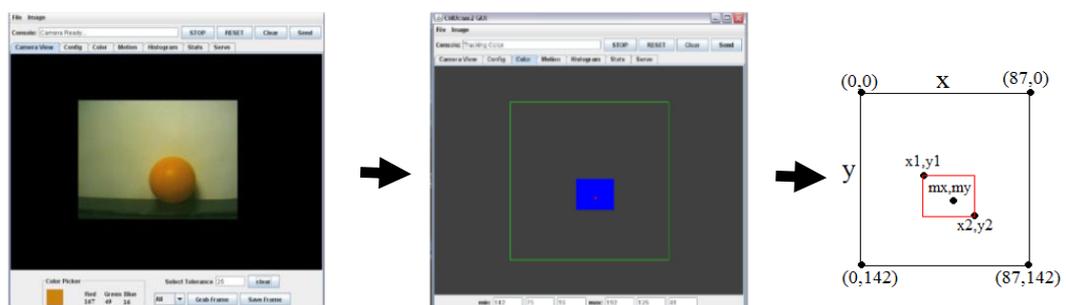
Sedangkan contoh yang lain dari komunikasi CMUcam3 dengan komputer, dapat dilihat pada gambar 3-6 dan 3-7. Pada gambar tersebut *software* yang digunakan adalah CMUcam2GUI. Banyak fasilitas dari CMUcam2GUI yang dapat dimanfaatkan untuk memperoleh informasi pengolahan citra atau *image processing* yang dilakukan oleh sensor kamera CMUcam3. Informasi yang diperoleh dapat seperti pada penggunaan *software* hiperterminal, yaitu berupa nilai RGB ataupun nilai posisi dari suatu obyek berwarna.



Gambar 3-6 Tampak *Real* Obyek Bola Berwarna *Orange* pada CMUcam2GUI

Gambar 3-6 memperlihatkan salah satu fasilitas dari CMUcam2GUI untuk menampilkan bentuk *real* dari suatu obyek berwarna, dan nilai RGB dari obyek berwarna tersebut. Untuk melihat nilai RGB pada CMUcam2GUI, prinsipnya sama seperti perintah GT. Hanya yang membedakan adalah, posisi warna yang ingin diketahui nilai RGB-nya dapat ditentukan sesuai dengan yang diinginkan.

Sedangkan contoh untuk mengetahui nilai posisi dari suatu obyek berwarna pada CMUcam2GUI, dapat dilihat pada gambar 3-7. Prinsipnya sama dengan perintah TC, yaitu menggunakan nilai RGB sebagai referensi untuk memperoleh nilai posisi dari suatu obyek berwarna. Hanya yang membedakan adalah, hasil yang diperoleh dari CMUcam2GUI bukan berbentuk angka, namun sudah dikonversi menjadi suatu tampilan dua dimensi yang lebih mudah untuk dipahami.



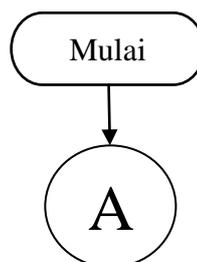
Gambar 3-7 Tampilan dari CMUcam2GUI Hasil *Image Processing* Sensor Kamera CMUcam3

Komunikasi CMUcam3 dengan komputer, dengan menggunakan *software* yang lain seperti *serial watcher* dan CMUcam3 *frame grabber*, dapat dilihat pada gambar 3-14 dan 3-16. Prinsip kerja dari *serial watcher* seperti hiperterminal, dan prinsip kerja dari CMUcam3 *frame grabber* seperti CMUcam2GUI. Dan dari nilai hasil pengolahan citra yang dilakukan oleh sensor kamera CMUcam3 terhadap suatu obyek berwarna yang ditampilkan oleh *software* yang sudah tersedia untuk komputer inilah, yang selanjutnya dijadikan sebagai nilai untuk mengoreksi apakah sudah benar atau tidak, nilai yang diterima dan disimpan oleh sistem kontroller, ketika berkomunikasi dengan sensor kamera CMUcam3.

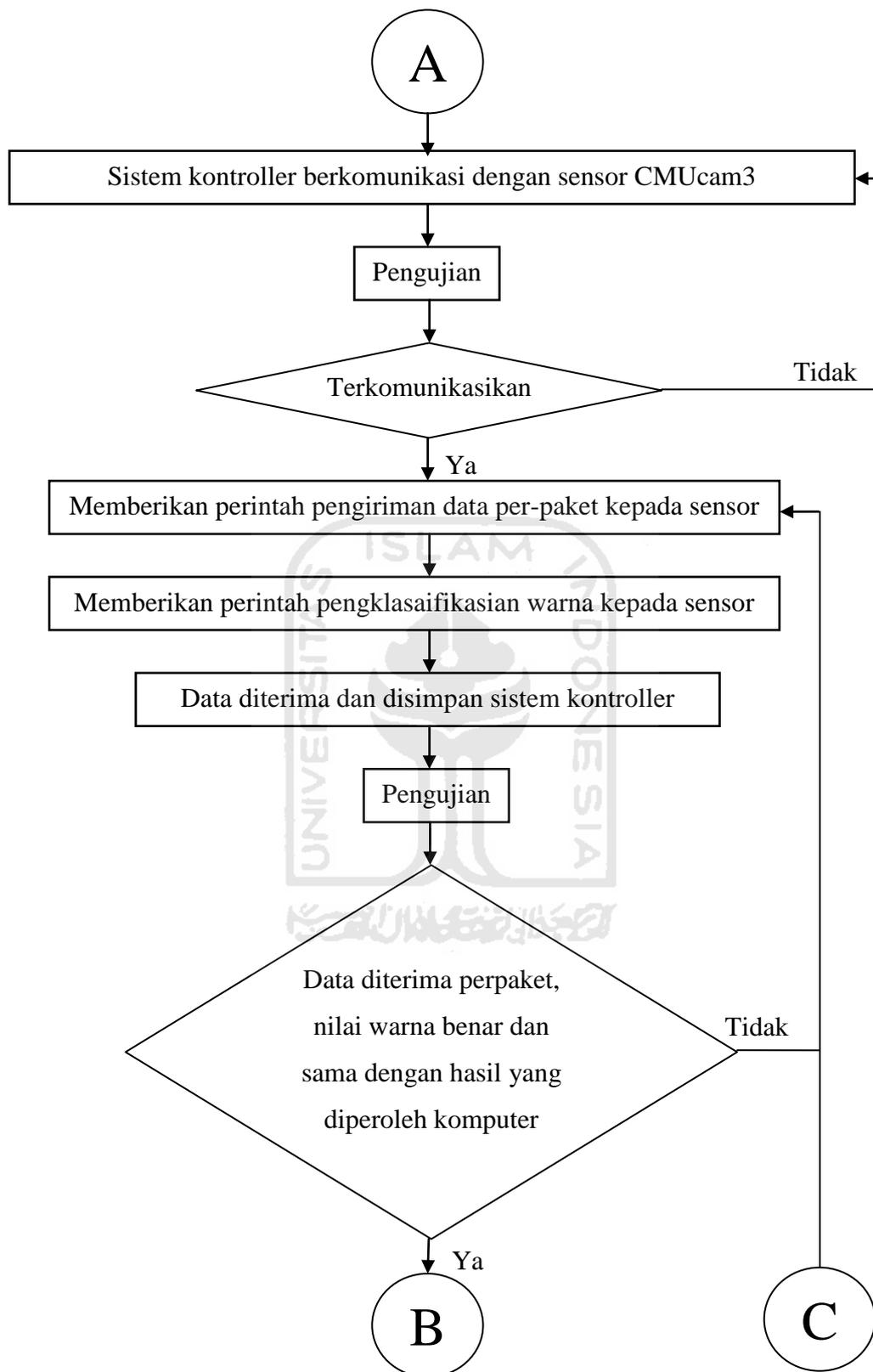
4.4.2 Komunikasi CMUcam3 dengan Sistem Kontroller

Seperti yang telah dijelaskan pada bab sebelumnya, komunikasi sensor kamera CMUcam3 dengan sistem kontroller dimaksudkan untuk memperoleh data hasil dari *image processing* yang dilakukan oleh sensor kamera CMUcam3, yang berupa pengklasifikasian warna dan penjejukan obyek berwarna. Dimana data tersebut kemudian dimanfaatkan oleh sistem kontroller sebagai informasi untuk menjalankan seluruh sistem yang ada pada robot, sehingga robot dapat mencari dan mendapatkan suatu target yang berupa obyek berwarna.

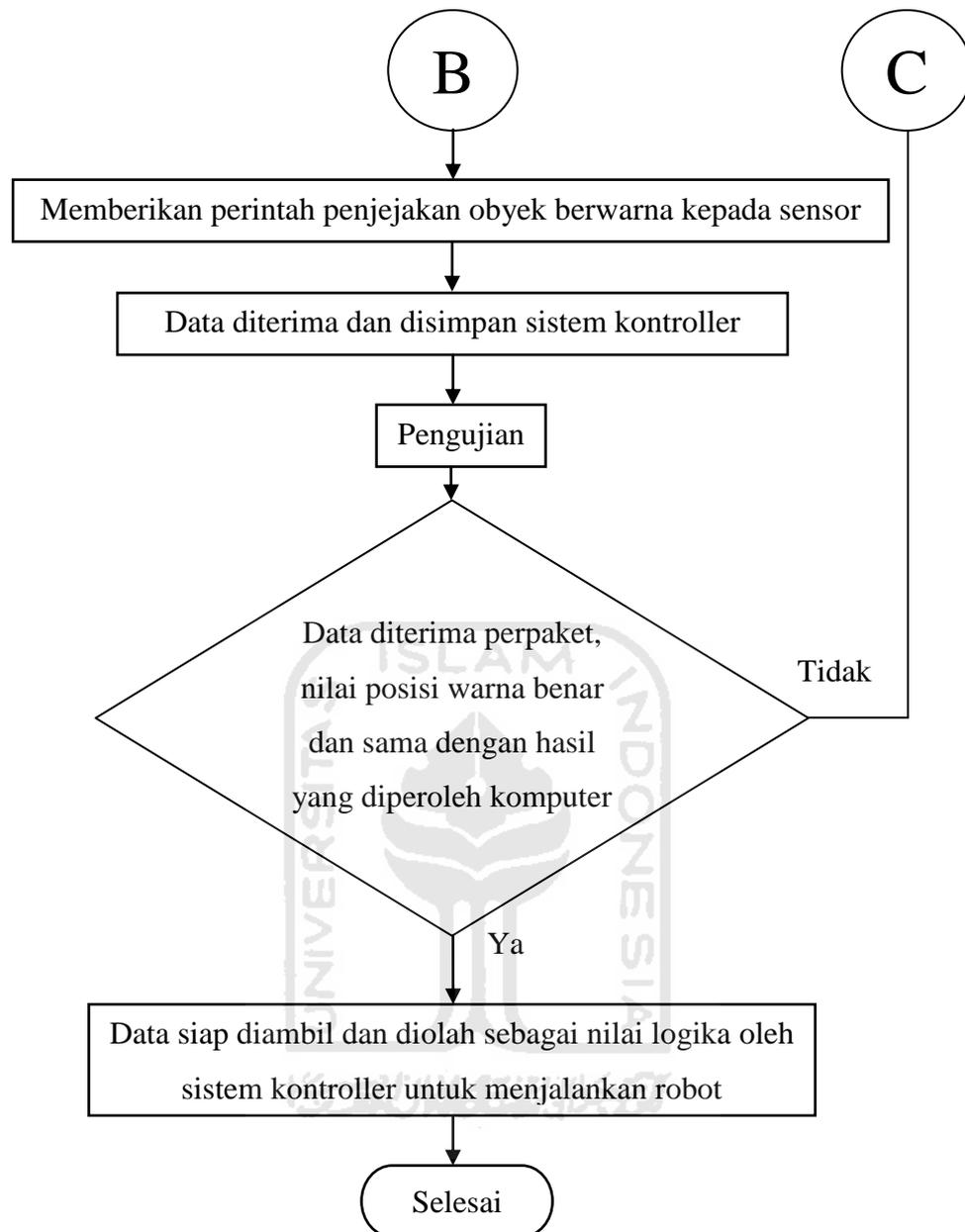
Ada hal yang harus sistem kontroller lakukan secara urut agar data yang ditransfer oleh sensor kamera CMUcam3, dapat diperoleh dan kemudian dapat diolah. Hal tersebut adalah pemberian perintah pengolahan citra, kemudian penerimaan dan penyimpanan data. Setelah data diterima dan disimpan, selanjutnya data dapat diambil, dan kemudian baru dapat diolah oleh sistem kontroller, untuk dimanfaatkan sebagai nilai logika, agar dapat menjalankan seluruh tugas robot dengan baik. Langkah-langkah tersebut dapat dilihat pada gambar 3-8.



Gambar 3-8 *Flowchart* Memperoleh Hingga Mengolah Data



Gambar 3-8 *Flowchart* Memeroleh Hingga Mengolah Data (lanjutan)



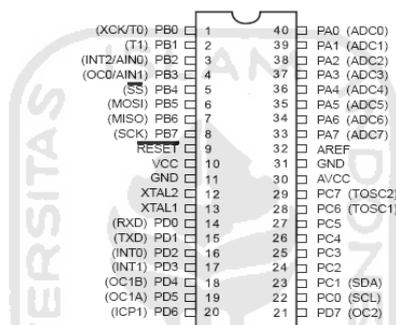
Gambar 3-8 *Flowchart* Mem peroleh Hingga Mengolah Data (lanjutan)

Dari gambar 3-8, hal pertama yang dilakukan sistem kontroller adalah berkomunikasi dengan sensor kamera CMUcam3. Berdasarkan pada bab sebelumnya, sesuai dengan fitur-fitur yang dimiliki oleh mikrokontroller Atmega16 dan sensor kamera CMUcam3. Komunikasi yang dapat digunakan oleh kedua peranti tersebut adalah komunikasi *serial*.

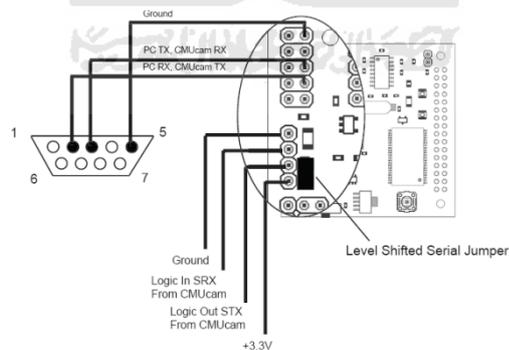
Pada bab sebelumnya, USART ATmega16 memungkinkan transmisi data secara asinkron. Dengan menghubungkan jalur TXD (pengirim) maupun RXD

(penerima), kemudian menyamakan kecepatan transfer data atau sering disebut sebagai *baud rate* pada sistem controller dan sensor kamera CMUcam3, maka kedua piranti tersebut akan dapat berkomunikasi dan saling mentransfer data.

Jalur pengirim dan penerima data pada ATmega16 adalah, pin PORTD.1 dan pin PORTD.0. Sedangkan pada sensor kamera CMUcam3 terdapat pada *serial port* sensor kamera CMUcam3. Untuk lebih jelasnya dapat dilihat pada gambar 3-9 dan gambar 3-10. Dengan terjadinya komunikasi dan transmisi data, maka sistem controller akan memperoleh data-data yang diperlukan untuk menjalankan seluruh sistem yang ada pada robot, sehingga robot dapat menyelesaikan tugasnya dengan baik, sesuai dengan apa yang diharapkan.



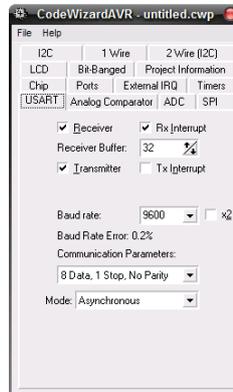
Gambar 3-9 Konfigurasi Pin pada ATmega16



Gambar 3-10 Serial Port Sensor Kamera CMUcam3

Pada mikrokontroler ATmega16, untuk memanfaatkan komunikasi *serial* dapat dilakukan dengan men-*setting* TXD dan RXD pada Wizard Code Vision AVR seperti pada gambar 3-11. Gambar 3-11 merupakan pengaktifan TXD, RXD dan interupsi penerima data pada fasilitas USART dengan *baud rate* yang

digunakan adalah 9600, 8 bit data, 1 *stop* bit dan no parity, dengan mode *asynchronous*.



Gambar 3-11 Konfigurasi USART

Fasilitas interupsi penerima data, digunakan agar data yang dikirimkan oleh sensor kamera CMUcam3 dapat diterima terus menerus oleh sistem kontroller. Selain cara pengaktifanya seperti pada gambar 3-11, cara lainya adalah dapat langsung membuat program pengaktifan interupsi penerima data pada Code Vision AVR seperti pada gambar 3-12.

```
// USART Receiver buffer
#define RX_BUFFER_SIZE 32

char rx_buffer[RX_BUFFER_SIZE];
unsigned char rx_index;
bit rx_flag=0;

// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
    char status;
    status=UCSRA;
    if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)
    {
        rx_buffer[rx_index]=UDR;
        if (rx_buffer[rx_index]!=':')
        {
            rx_buffer[rx_index]=0;
            rx_index=0;
            rx_flag=1;
        }
        else if (rx_buffer[rx_index]==13) rx_buffer[rx_index]=0;
        if (++rx_index == RX_BUFFER_SIZE) rx_index=0;
    }
};
}
```

Gambar 3-12 Program Interupsi Penerima Data

Program pada gambar 3-12 dapat diartikan bahwa, data yang ditransfer oleh sensor kamera CMUcam3 akan diterima terus menerus jika proses penerimaan data tidak mengalami kesalahan. Kesalahan dapat terjadi pada UCSRA, *framing error*, *parity error* dan *data over run*. Program tersebut juga memperlihatkan cara penyimpanan data yang diterima oleh sistem kontroller dari sensor kamera CMUcam3. Yang dimaksud cara menyimpan data adalah, memasukan secara terus menerus dan serentak data hasil perintah *image processing* dari sensor CMUcam3 yang sudah berbentuk paket data, ke dalam *variable array* bernama *rx_buffer[]* yang mempunyai kapasitas 32 elemen. Dimana nantinya data pada setiap elemen tersebut dimanfaatkan untuk program-program yang lain, dengan cara menyimpan ulang pada *variable* lain, agar nilainya tetap tersimpan pada *variable* lain tersebut, sehingga dapat dimanfaatkan secara terus menerus. Untuk lebih jelasnya dapat dilihat dari program pada gambar 3-13.

Hal selanjutnya untuk mempermudah memahami data yang diterima, menyimpan, mengambil dan mengolah data. Maka pemberian perintah pengiriman data perpaket kepada sensor kamera CMUcam3 harus dilakukan, perintah tersebut adalah RM 3 (row mode). Perintah RM 3 membuat data hasil dari *image processing* berupa bilangan ASCII yang harusnya dikirim kepada sistem kontroller, diolah terlebih dahulu oleh chip prosesor yang ada pada sensor kamera CMUcam3 menjadi suatu nilai berupa desimal yang mudah dipahami. Setelah itu nilai tersebut baru dikirim kepada sistem kontroller, sehingga data yang diterima dari sensor kamera CMUcam3, tidak lagi harus diolah oleh sistem kontroller untuk dijadikan nilai yang mudah dipahami dan mudah dimanfaatkan.

Setelah perintah pengiriman data per-paket dilakukan, maka selanjutnya pemberian perintah *image processing* kepada sensor kamera CMUcam3, seperti contohnya perintah pengklasifikasian warna (GT) yang hasilnya berupa nilai RGB, ataupun perintah penjejakan obyek berwarna (TC) yang hasilnya berupa nilai posisi, dapat dilakukan. pada *step* ini untuk memperoleh data yang benar, urutan perintah untuk sensor kamera CMUcam3 seperti pada gambar 3-3, juga harus dilakukan dengan benar. Program untuk sistem kontroller agar dapat memberikan perintah GT dan TC kepada sensor kamera CMUcam3, dapat dilihat

pada gambar 3-13. Mengacu pada pernyataan sebelumnya, program perintah GT dan TC tersebut, dibuat sesuai dengan gambar 3-13. Hal itu dibuktikan dari satu perintah dengan perintah lain yang saling berkaitan. Ditunjukkan dari sebelum melakukan perintah GT, maka perintah TW dilakukan terlebih dahulu. Karena data hasil dari perintah TW, digunakan sebagai nilai referensi untuk melakukan perintah GT. Selanjutnya hasil dari perintah GT yang telah disimpan pada *variable* r1[n] g1[n] b1[n], digunakan sebagai nilai referensi untuk melakukan perintah TC. Dan sampai akhirnya dari perintah TC akan diperoleh nilai posisi dari suatu obyek berwarna, yang kemudian disimpan pada *variable* x[n] y[n] z[n], yang selanjutnya nilai tersebut digunakan sebagai navigasi robot, untuk mendapatkan targetnya yang berupa obyek berwarna.

```

void gt(unsigned char n)
{
    send_com("tw");

    clear_buffer_rx();
    printf("gt");
    putchar(13);

    while(rx_flag==0);

    r1[n]=rx_buffer[1];
    g1[n]=rx_buffer[2];
    b1[n]=rx_buffer[3];

    rh[n]=rx_buffer[4];
    gh[n]=rx_buffer[5];
    bh[n]=rx_buffer[6];

    lcd_clear();
    sprintf(lcd_buffer,"r%3d g%3d b%3d",r1[n],g1[n],b1[n]);
    lcd_gotoxy(0,1);lcd_puts(lcd_buffer);
    sprintf(lcd_buffer,"R%3d G%3d B%3d",rh[n],gh[n],bh[n]);
    lcd_gotoxy(0,0);lcd_puts(lcd_buffer);
}

void tc(unsigned char n)
{
    clear_buffer_rx();
    printf("tc %d %d %d %d",r1[n],g1[n],b1[n],15);
    putchar(13);

    while(rx_flag==0);

    x[n]=rx_buffer[2]; //x posisi
    y[n]=rx_buffer[3]; //y posisi
    z[n]=rx_buffer[8]; //jumlah pixel

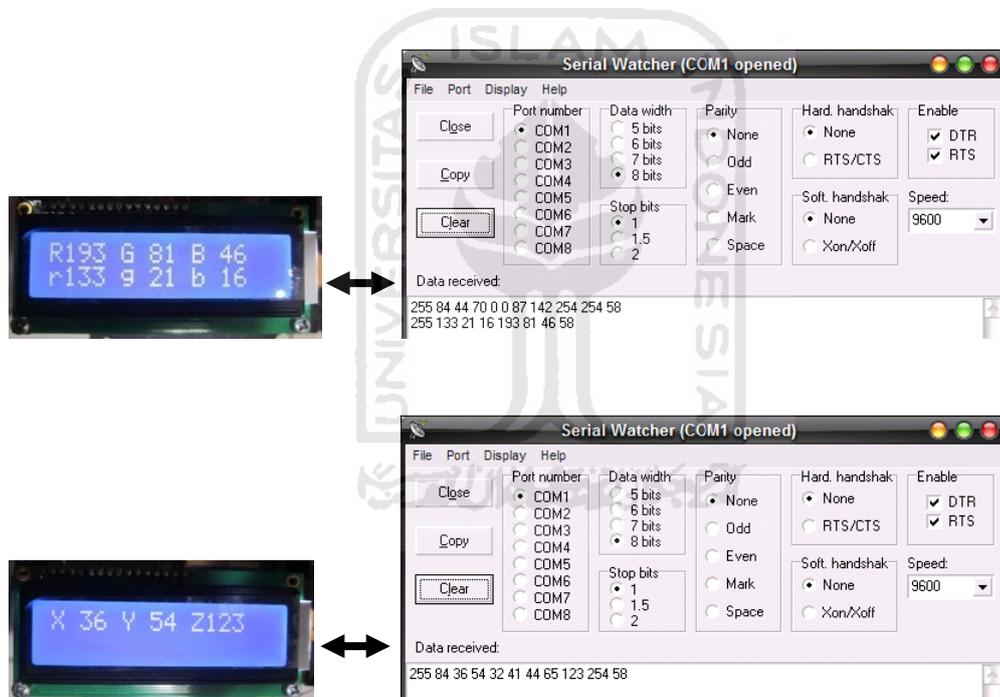
    x0[n]=rx_buffer[4]; //
    y0[n]=rx_buffer[5]; //
    x1[n]=rx_buffer[6]; //
    y1[n]=rx_buffer[7]; //

    lcd_clear();
    sprintf(lcd_buffer,"X%3d Y%3d Z%3d",x[n],y[n],z[n]);
    lcd_gotoxy(0,0);
    lcd_puts(lcd_buffer);
}

```

Gambar 3-13 Program Perintah GT dan TC

Setelah nilai yang diperoleh oleh sistem controller dari perintah GT dan TC dinyatakan benar. Dengan dibuktikan dari hasil perbandingan antara nilai yang diperoleh oleh sistem controller dengan nilai yang diperoleh oleh komputer dari kedua perintah tersebut adalah sama. Maka kemudian nilai tersebut dapat diambil dan diolah oleh sistem controller sebagai nilai logika, untuk menjalankan seluruh sistem yang ada pada robot, sehingga robot dapat menyelesaikan tugasnya dengan baik. Contoh perbandingan antara nilai yang diperoleh oleh sistem controller dengan nilai yang diperoleh oleh komputer yang terbukti hasilnya adalah sama, dapat dilihat pada gambar 3-14. Pada gambar tersebut, sistem controller menampilkan nilai yang diperoleh dari perintah GT dan TC pada LCD, dan untuk komputer pada *software serial watcher*.



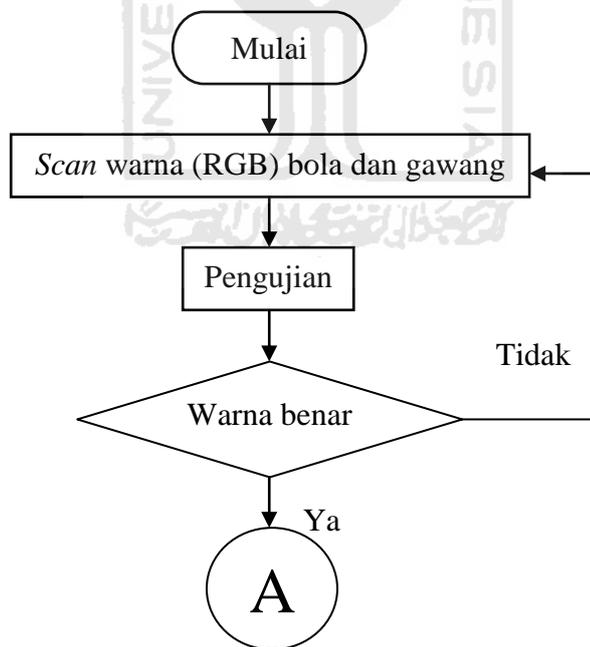
Gambar 3-14 Menbandingkan Nilai RGB dan Nilai Posisi dari Sistem Controller (LCD) dan Komputer (*Serial Watcher*)

Salah satu indikator cara memperoleh hingga mengolah data berhasil digunakan oleh sistem controller untuk memperoleh hingga mengolah data hasil dari *image processing* yang dilakukan oleh sensor kamera CMUcam3, ditunjukkan dari dapat tampilkannya nilai hasil dari *image processing* tersebut ke dalam LCD, seperti ditunjukkan pada gambar 3-14. Namun ketika data yang diharapkan tidak

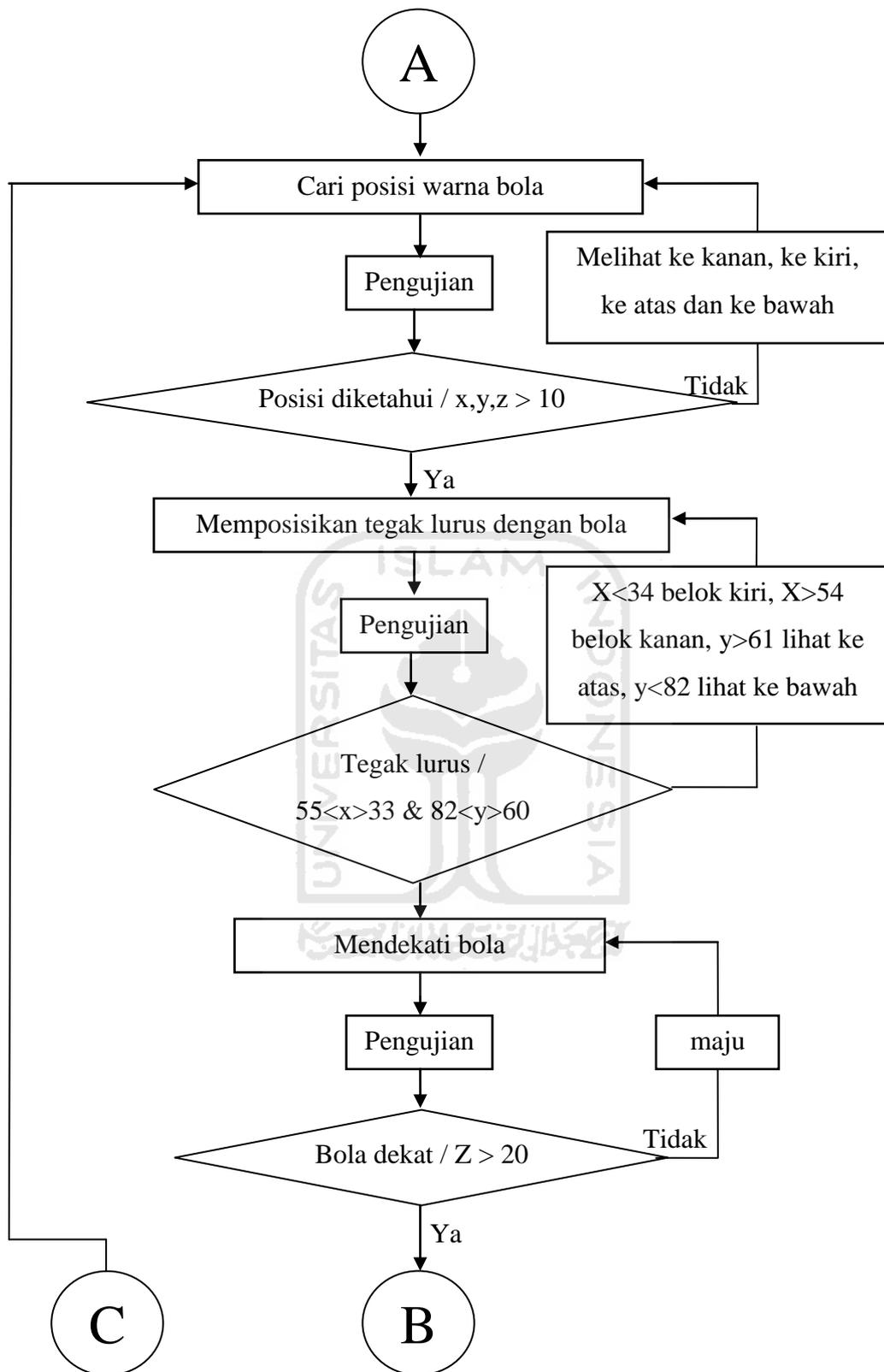
diperoleh, maka bisa dipastikan bahwa ada kesalahan yang dapat berupa kesalahan dalam cara memperoleh, kesalahan saat mengkomunikasikan, kesalahan dalam pemberian perintah *image processing*, ataupun kesalahan dalam mengolah data. Dengan pengujian dan perbaikan pada setiap langkah yang dilewati, maka kesalahan dapat dengan mudah diketahui dan diperbaiki.

4.5 Penggabungan Sensor Kamera CMUcam3 dengan Robot

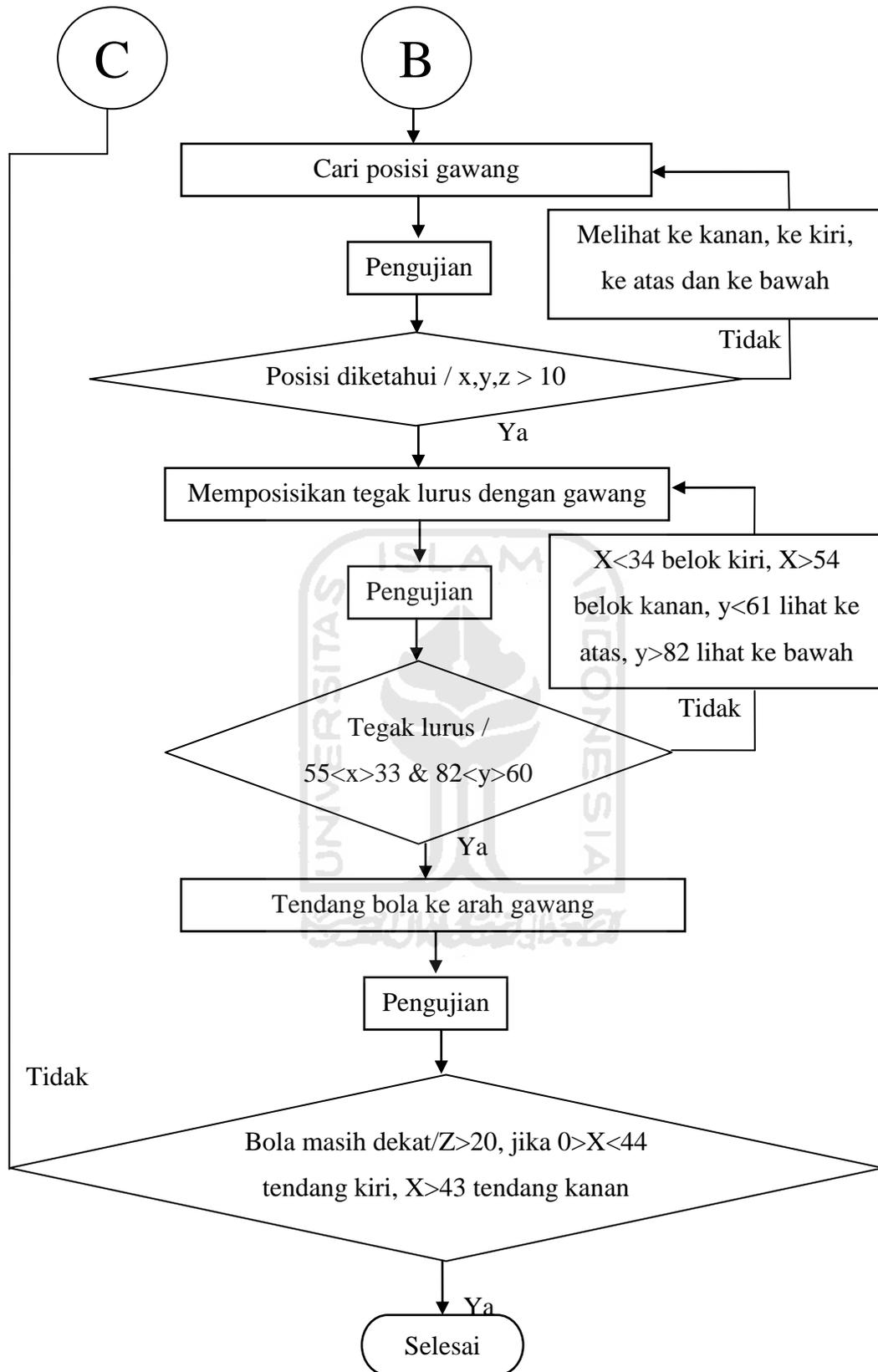
Yang dimaksud dengan penggabungan sensor kamera CMUcam3 dengan robot adalah, mengkomunikasikan antara pengendali gerak yang sudah ada pada robot dengan sistem kontroller yang digunakan untuk memperoleh informasi dari sensor kamera CMUcam3. Penggabungan ini bertujuan untuk membuat robot secara otonom mampu menyelesaikan tugasnya, yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Untuk dapat menyelesaikan tugasnya dengan baik, urutan yang harus dikerjakan oleh robot dapat seperti yang terlihat pada gambar 3-15.



Gambar 3-15 *Flowchart* Penyelesaian Tugas Robot



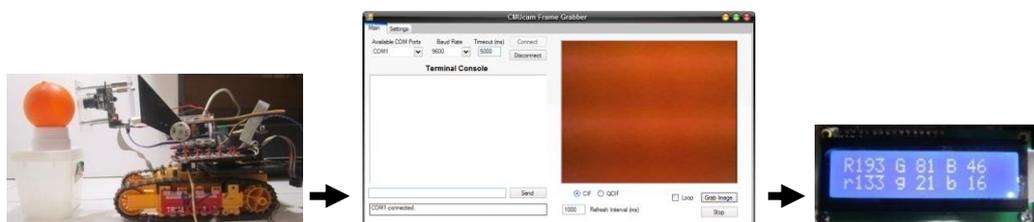
Gambar 3-15 Flowchart Penyelesaian Tugas Robot (lanjutan)



Gambar 3-15 Flowchart Penyelesaian Tugas Robot (lanjutan)

Gambar 3-15 memperlihatkan bahwa, untuk dapat menyelesaikan tugasnya pengendali gerak yang terdapat pada robot sudah memberi kemampuan pada robot untuk dapat bergerak seperti maju, belok kanan, belok kiri, melihat ke kanan, melihat ke kiri, melihat ke atas, melihat ke bawah dan menendang bola. Gambar 3-15 juga memperlihatkan bahwa kemampuan dari sensor kamera CMUcam3 yang berupa pengklasifikasian warna dan penjejakan obyek berwarna yang telah diupayakan dalam penelitian ini agar cara penggunaannya diketahui, sangatlah berguna bagi robot untuk dapat menyelesaikan tugasnya dengan baik. Hal tersebut ditunjukkan dari langkah *Scan* warna (RGB) bola dan gawang, membutuhkan kemampun pengklasifikasian warna yang dimiliki oleh sensor kamera CMUcam3. Kemudian untuk mencari posisi dari bola dan gawang hingga mendapatkannya, membutuhkan kemampuan penjejakan obyek berwarna yang dimiliki oleh sensor kamera CMUcam3.

Langkah pertama pada gambar 3-15 adalah *scan* warna, *scan* warna secara program merupakan perintah GT seperti pada gambar 3-14. Prinsip kerja program tersebut ketika dijalankan adalah, dengan memperlihatkan target yang berupa obyek berwarna pada sensor kamera CMUcam3, dan kemudian memberikan perintah pengklasifikasian warna. Maka pemindaian warna akan dilakukan satu kali sehingga memperoleh satu data berisi nilai RGB. Dan kemudian dari nilai RGB tersebut akan digunakan sebagai referensi untuk menjalankan langkah-langkah selanjutnya, yaitu pencarian posisi dari obyek berwarna. Contoh langkah *scan* warna dapat dilihat pada gambar 3-16, pada gambar tersebut obyek yang digunakan adalah bola berwarna oranye.



Gambar 3-16 *Scan* Warna

Gambar 3-16 memperlihatkan bahwa langkah *scan* warna dilakukan sangat dekat sekitar 1.5 cm. Hal tersebut dimaksudkan agar sensor kamera CMUcam3 dapat fokus terhadap warna dari obyek seperti yang ditampilkan pada CMUcam3 *frame greeber*, sehingga nilai RGB yang diperoleh benar dan akurat seperti yang ditampilkan pada LCD.

Setelah mengetahui nilai RGB dari obyek bola dan gawang, selanjutnya robot dapat mencari posisi target pertamanya yaitu obyek bola. Proses pencarian obyek bola dan gawang merupakan gerakan robot yang meliputi melihat ke kanan, ke kiri, ke atas dan ke bawah. Dan setelah menemukan posisi obyek bola dengan dibuktikan dari terpenuhinya kondisi nilai $x \ y \ z > 10$, maka kemudian robot dapat memposisikan dirinya tegak lurus dengan bola dan mendekati bola tersebut hingga nilai $z > 20$ atau jangkauan tendang robot terhadap bola dapat dilakukan. Kemudian setelah robot mendapatkan obyek bola, maka pencarian posisi target yang kedua yaitu obyek gawang dapat dilakukan. Kemudian setelah menemukan posisi obyek gawang dengan dibuktikan dari terpenuhinya nilai $x \ y \ z > 10$, maka robot dapat memposisikan dirinya tegak lurus dengan gawang dan menendang bola ke arah gawang sehingga bola dapat masuk atau berada di area gawang. Sebelum robot menendang bola, langkah yang dilakukan oleh robot adalah melihat kembali posisi bola, ketika bola berada pada jangkauan tentang kaki kiri atau nilai $x < 44$ maka robot akan menendang dengan kaki kiri, sedangkan ketika bola berada pada jangkauan tentang kaki kanan atau nilai $x > 43$ maka robot akan menendang dengan kaki kanan.

Program pencarian posisi obyek berwarna dapat dilihat pada bagian perintah TC di gambar 3-14. Dengan memanfaatkan *variable* $x[n] \ y[n] \ z[n]$ pada program tersebut sebagai nilai logika dalam program selanjutnya, maka robot dapat bergerak untuk mencari hingga mendapatkan targetnya. Karena *variable* $x[n] \ y[n] \ z[n]$ dapat diartikan sebagai koordinat dari posisi target yang terlihat oleh sensor kamera CMUcam3. Untuk lebih jelasnya maksud dari koordinat dapat dilihat pada gambar 3-7. *Variable* $x[n]$ dan $y[n]$ dapat diartikan sebagai posisi di kanan atau di kiri, lebih tinggi atau lebih rendah obyek berwarna terhadap sensor kamera CMUcam3. Sedangkan *variable* $z[n]$ dapat diartikan sebagai jauh dekatnya, jarak dari obyek berwarna terhadap sensor kamera CMUcam3. Contoh

program yang memanfaatkan *variable* $x[n]$ $y[n]$ $z[n]$ sebagai nilai logika untuk menjalankan robot agar berhasil mencari dan mendapatkan targetnya dapat dilihat pada gambar 3-17.

```

if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

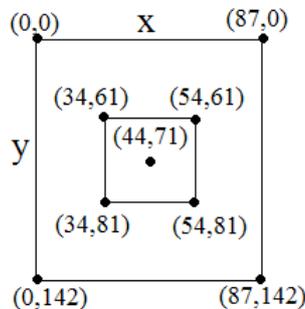
if (x[0]>1&&x[0]<34){a=1;b=0;j=0;};
if (x[0]>33&&x[0]<55){a=0;b=0;};
if (x[0]>54){a=0;b=1;j=0;};
if (a==1&&b==0){kiri();};
if (a==0&&b==1){kanan();};
if (a==0&&b==0){j=1;};

if (j==1&&pwm1>1175){maju();};
if (j==1&&pwm1<1176){tegak();delay_ms(1250);f=7;};

```

Gambar 3-17 Contoh Program Logika

Contoh program pada gambar 3-17, menjelaskan bahwa ketika posisi target yang berupa obyek berwarna tidak berada dalam posisi tengah dari jangkauan penglihatan sensor kamera CMUcam3, atau tidak di dalam koordinat $x=34$ $y=61$ sampai $x=54$ $y=81$ seperti gambar 3-18, atau posisi bola tidak tegak lurus dengan robot, maka robot akan mengkondisikan dirinya selalu tegak lurus dengan targetnya. Dan setelah robot tegak lurus dengan targetnya, namun posisi target masih jauh terhadap letak sensor kamera CMUcam3 yang terpasang pada robot, maka robot akan mendekati targetnya tersebut sampai jangkauan tendang robot terhadap targetnya dapat dilakukan.



Gambar 3-18 Jangkauan Penglihatan Sensor Kamera CMUcam3

Dengan menjalankan langkah-langkah pada gambar *flowchat* 3-15, dipastikan robot akan berhasil menyelesaikan tugasnya dengan baik yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna, atau lebih spesifik lagi robot dapat bermain sepak bola.



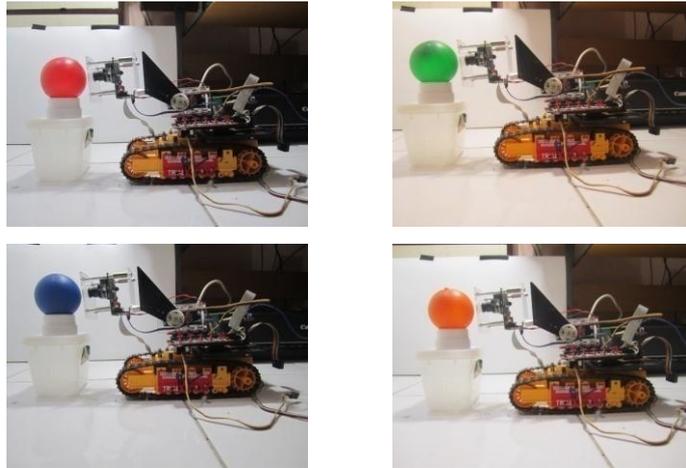
BAB IV

HASIL DAN PEMBAHASAN

Bab IV berisikan tentang beberapa hasil dari percobaan dan sekaligus pembahasannya. Percobaan yang dilakukan meliputi pengklasifikasian warna, penjejukan obyek berwarna dan kemampuan robot mencari dan mendapatkan target. Percobaan-percobaan tersebut dimaksudkan untuk menunjukkan hasil dari penelitian yang telah dilakukan, dan sekaligus membuktikan bahwa sensor kamera CMUcam3 dapat digunakan untuk mengklasifikasi warna dan menjejak obyek berwarna. Kemudian cara memperoleh hingga mengolah data yang dijelaskan pada bab sebelumnya, dapat digunakan oleh sistem kontroller untuk memperoleh hingga mengolah data hasil dari *image processing* yang dilakukan oleh sensor kamera CMUcam3. Dan sensor kamera CMUcam3 dapat digunakan sebagai sensor penglihatan pada robot dua kaki atau robot *humanoid*, sehingga robot mempunyai kemampuan secara otonom mencari dan mendapatkan suatu target yang berupa obyek berwarna.

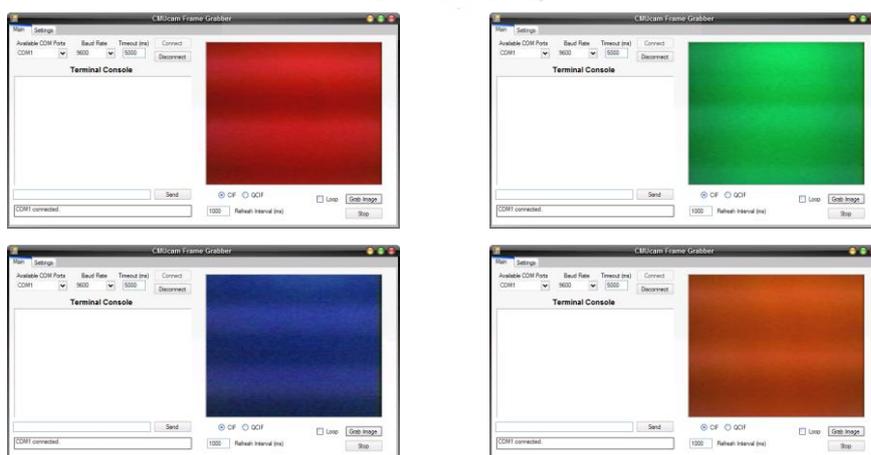
4.1 Pengklasifikasian Warna

Percobaan ini bertujuan untuk memperlihatkan dan membuktikan bahwa sensor kamera CMUcam3 dapat digunakan untuk mengklasifikasi warna, atau membedakan satu warna dengan warna lain. Seperti yang telah dijelaskan pada bab sebelumnya, hasil dari perintah pengklasifikasian warna (TC) adalah nilai RGB (*red green blue*) atau nilai paket data S. Dan dari nilai RGB atau nilai kombinasi warna tersebut, kemudian dapat digunakan sebagai parameter untuk mengidentifikasi dan membedakan satu warna dengan warna lain. Percobaan dilakukan dengan 4 obyek berwarna berbentuk bola, yaitu merah, hijau, biru dan oranye. Dimana warna dari setiap obyek berwarna tersebut ditangkap oleh sensor kamera CMUcam3 pada jarak 1.5 cm, seperti terlihat pada gambar 4-1.



Gambar 4-1 Obyek Percobaan Pengklasifikasian Warna

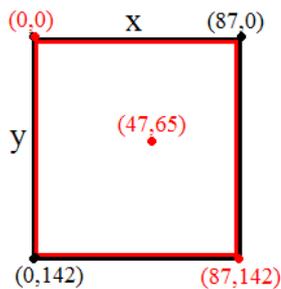
Dengan jarak 1.5 cm, maka warna yang ditangkap oleh sensor kamera CMUcam3 dapat terfokus pada warna dari obyek berwarna tersebut. Untuk lebih jelasnya dapat dilihat pada gambar 4-2. Gambar 4-2 memperlihatkan warna yang sebenar-benarnya ditangkap oleh sensor kamera CMUcam3 yang kemudian ditampilkan oleh *software* CMUcam3 *frame grabber*. Warna seperti pada gambar 4-2 tersebut, yang kemudian diproses oleh sensor kamera CMUcam3, sehingga menjadi nilai RGB. Untuk mendapatkan hasil rekaman warna seperti pada gambar 4-2, dapat dilakukan dengan cara memencet *button grab image* yang terdapat pada sudut kanan bawah dari tampilan CMUcam3 *frame grabber*.



Gambar 4-2 Tampak Warna Merah Hijau Biru dan Oranye dari Obyek Berwarna pada CMUcam3 *frame grabber*

Percobaan pengklasifikasian warna dilakukan dengan tiga kali pengambilan data pada setiap obyek berwarna. Hal ini dimaksudkan untuk mendapatkan hasil yang benar dan meyakinkan. Ketiga data tersebut ditampilkan oleh sistem kontroller dengan LCD, dan komputer dengan *software serial watcher*. Hasil percobaan selengkapnya dapat dilihat pada tabel 4-1. Tabel 4-1 memperlihatkan bahwa sensor kamera CMUcam3 dapat digunakan untuk mengklasifikasi warna. Dibuktikan dari nilai RGB pada setiap warna terlihat berbeda satu dengan yang lain, dan nilai RGB tersebut sesuai dengan warnanya. Sebagai contoh, pada percobaan pertama yang ditampilkan pada LCD untuk warna merah, nilai RGB yang diperoleh adalah $R_{min}=137$ $G_{min}=16$ $B_{min}=16$ $R_{max}=197$ $G_{max}=54$ $B_{max}=48$. Nilai RGB atau nilai kombinasi warna tersebut menunjukkan bahwa warna yang paling dominan adalah warna merah, karena R_{min} dan R_{max} mempunyai nilai terbesar dari nilai yang lain. Sehingga nilai RGB tersebut dapat dinyatakan benar dan sesuai dengan warna merah.

Tabel 4-1 juga memperlihatkan bahwa sistem kontroller berhasil memperoleh dan mengolah data hasil dari *image processing* berupa pengklasifikasian warna, yang dilakukan oleh sensor kamera CMUcam3. Hal tersebut dibuktikan ketika membandingkan data yang ditransfer oleh sensor kamera CMUcam3 kepada komputer dan sistem kontroller, hasilnya sama. Hanya yang membedakan pada komputer menampilkan juga hasil dari perintah TW (*track window*), yang berupa nilai posisi warna dominan yang ditangkap oleh sensor kamera CMUcam3. Sebagai contoh dapat dilihat pada percobaan pertama untuk warna merah, hasil dari perintah TW adalah T 47 65 0 0 87 142 254 254. Nilai tersebut dapat diartinya sebagai koordinat dari posisi warna dominan yang ditangkap oleh sensor kamera CMUcam3. Dengan letak titik paling kiri warna dominan pada koordinat $x_1=0$ $y_1=0$, kemudian titik tengah pada koordinat $m_x=47$ $m_y=65$, dan titik paling kanan pada koordinat $x_2=87$ $y_2=142$. Untuk lebih jelasnya dapat dilihat pada gambar 4-3.

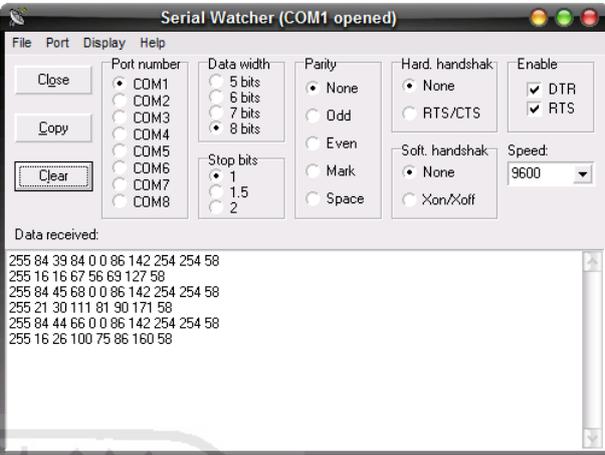
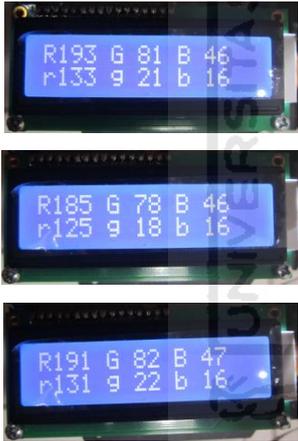
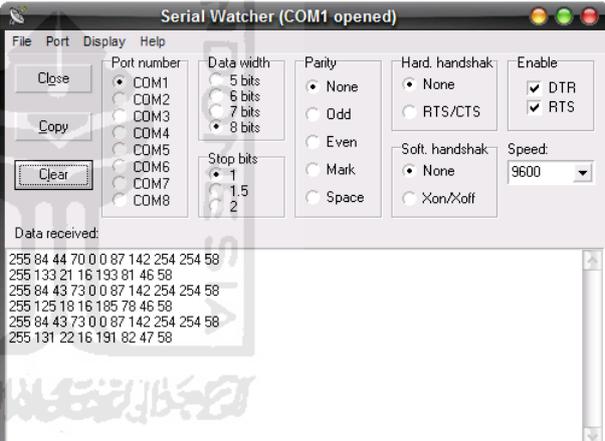


Gambar 4-3 Ilustrasi Hasil dari Perintah TW

Tabel 4-1 Percobaan Pengklasifikasian Warna

Warna	Hasil dari Sistem Kontroller (LCD)	Hasil dari Komputer (<i>software serial watcher</i>)
Merah		
Hijau		

Tabel 4-1 Percobaan Pengklasifikasian Warna (lanjutan)

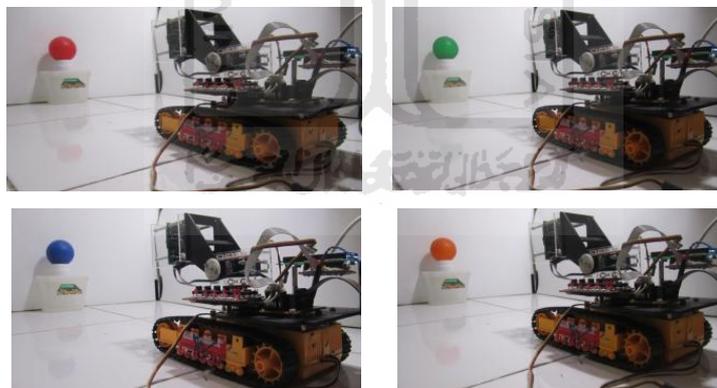
Warna	Hasil dari Sistem Kontroller (LCD)	Hasil dari Komputer (<i>software serial watcher</i>)
Biru		
Oranye		

Adanya perbedaan hasil dari percobaan pada setiap warna dari obyek berwarna yang terdapat pada tabel 4-1, disebabkan oleh faktor cahaya lampu yang dapat berubah-ubah, meskipun untuk mata pada manusia tidak merasakannya. Seperti contohnya nilai RGB untuk warna oranye yang ditampilkan oleh sistem kontroller pada LCD, terlihat berbeda-beda. Dari nilai percobaan pertama adalah $R_{min}=133$ $G_{min}=21$ $B_{min}=16$ $R_{max}=193$ $G_{max}=81$ $B_{max}=46$. Kemudian nilai percobaan kedua $R_{min}=125$ $G_{min}=18$ $B_{min}=16$ $R_{max}=185$ $G_{max}=46$ $B_{max}=46$, dan nilai percobaan ketiga $R_{min}=131$ $G_{min}=22$ $B_{min}=16$ $R_{max}=191$ $G_{max}=82$ $B_{max}=47$. Ketiga nilai yang diperoleh terlihat berbeda, namun masih dapat

diitoleransi dan dianggap benar karena selisih perbedaan satu nilai dengan nilai yang lain tidak terlalu tinggi.

4.2 Penjejakan Obyek Berwarna

Percobaan ini bertujuan untuk memperlihatkan dan membuktikan bahwa sensor kamera CMUcam3 dapat digunakan untuk menjejak suatu obyek berwarna, atau mengetahui posisi dari suatu obyek berwarna. Seperti yang telah dijelaskan pada bab sebelumnya, hasil dari perintah penjejakan obyek berwarna (TC) adalah paket data T yang dapat diartikan sebagai nilai koordinat dari posisi suatu obyek berwarna yang ditangkap oleh sensor kamera CMUcam3. Dan dari nilai tersebut, kemudian dapat digunakan untuk mendefinisikan letak dari suatu obyek berwarna, terhadap sensor kamera CMUcam3. Percobaan dilakukan dengan 4 obyek berwarna berbentuk bola, yaitu merah, hijau, biru dan oranye. Dimana penjejakan pada setiap obyek berwarna tersebut, dilakukan pada jarak 40 cm, dan tegak lurus terhadap sensor kamera CMUcam3. Untuk lebih jelasnya dapat dilihat pada gambar 4-4.

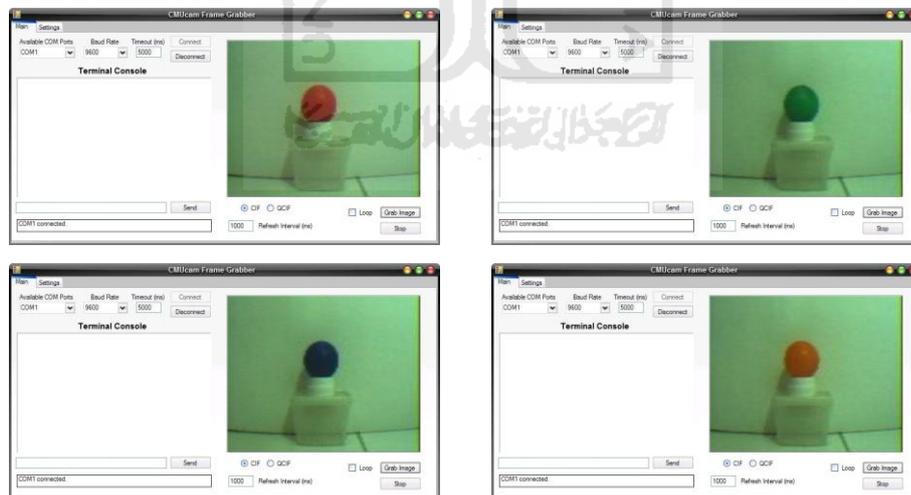


Gambar 4-4 Obyek Percobaan Penjejakan Obyek Berwarna

Sedangkan tampak *real* obyek berwarna yang ditangkap oleh sensor kamera CMUcam3, pada jarak 40 cm dan tegak lurus terhadap sensor kamera CMUcam3 dapat dilihat pada gambar 4-5. Gambar 4-5 memperlihatkan posisi yang sebenar-benarnya dari obyek bola berwarna merah, hijau, biru dan oranye yang ditangkap oleh sensor kamera CMUcam3, yang kemudian ditampilkan oleh *software* CMUcam3 *frame grabber*. Posisi seperti pada gambar 4-5 tersebut, yang

kemudian diproses oleh sensor kamera CMUcam3, sehingga menjadi nilai koordinat dari posisi suatu obyek berwarna.

Seperti pada percobaan sebelumnya, untuk mendapatkan hasil rekaman gambar seperti pada gambar 4-5, dapat dilakukan dengan cara memencet *button grab image* yang terdapat pada sudut kanan bawah dari tampilan CMUcam3 *frame grabber*. Dapat dilihat juga pada gambar 4-5, gambar hasil tangkapan dari sensor kamera CMUcam3 terlihat buram dan obyek bola menjadi sedikit lonjong. Hal tersebut disebabkan oleh pencahayaan yang kurang terang, dan pengaturan fokus lensa kamera yang terdapat pada sensor kamera CMUcam3 kurang tepat. Kedua hal tersebut juga dapat menjadi kendala bagi sensor kamera CMUcam3, ketika melakukan pengklasifikasian warna atau penjejakan obyek berwarna. Namun dengan memberikan pencahayaan yang lebih terang dan juga pengaturan fokus lensa yang tepat, maka gambar hasil tangkapan dari sensor kamera CMUcam3 akan lebih lebih cerah, bentuk obyek berwarna tidak berubah, dan sekaligus hal yang menjadi kendala dari sensor kamera CMUcam3 untuk melakukan pengklasifikasian warna atau penjejakan obyek berwarna akan berkurang dan dapat teratasi.



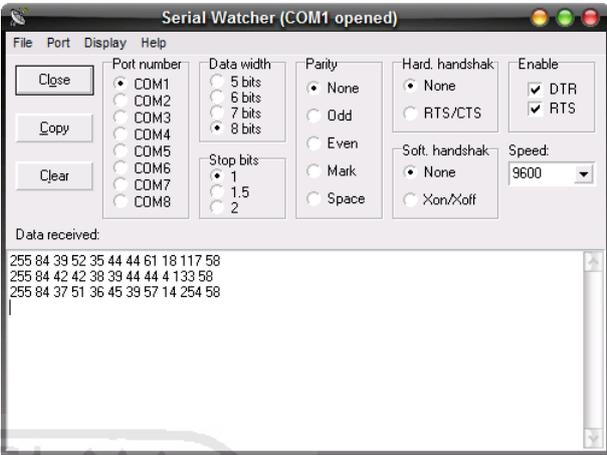
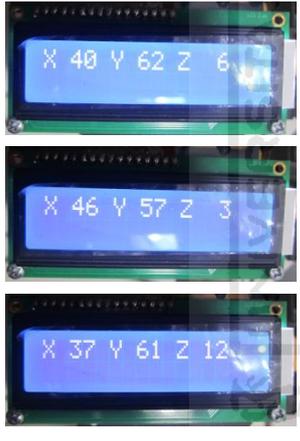
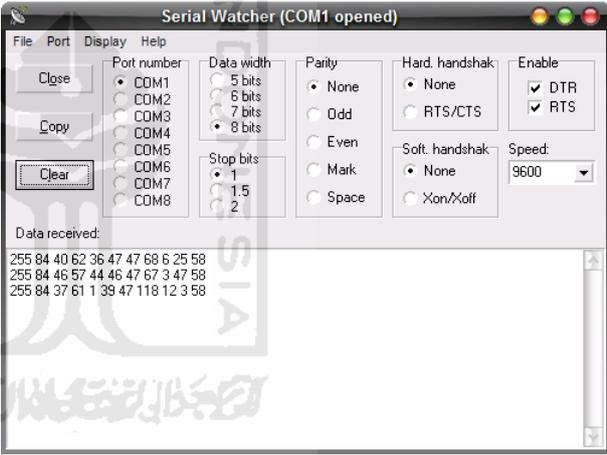
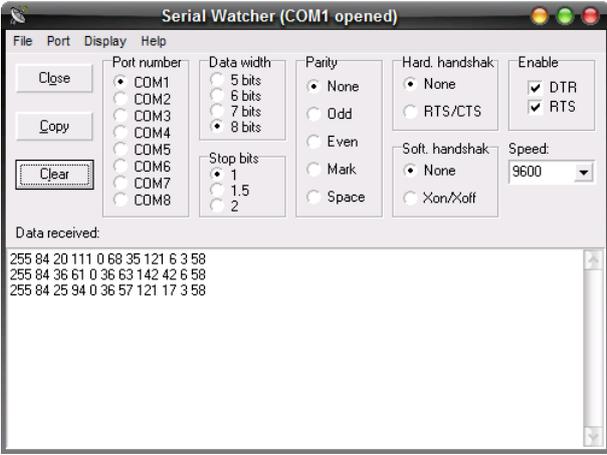
Gambar 4-5 Tampak *Real* Obyek Berwarna Merah Hijau Biru dan Oranye Pada CMUcam3 *frame grabber*

Sama seperti percobaan sebelumnya, percobaan penjejakan obyek berwarna dilakukan dengan tiga kali pengambilan data pada setiap obyek

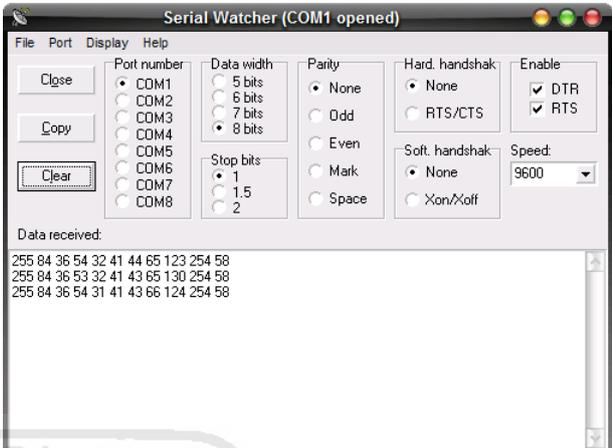
berwarna. Hal ini dimaksudkan untuk mendapatkan hasil yang benar dan meyakinkan. Ketiga data tersebut ditampilkan oleh sistem kontroller dengan LCD dan komputer dengan *software serial watcher*. Hasil percobaan selengkapnya dapat dilihat pada tabel 4-2. Tabel 4-2 memperlihatkan bahwa sensor kamera CMUcam3, dapat digunakan untuk menjejak suatu obyek warna. Dibuktikan dari nilai paket data T pada setiap obyek berwarna terlihat tidak jauh berbeda. Hal tersebut dikarenakan pada setiap obyek berwarna diletakan pada posisi yang hampir sama terhadap sensor kamera CMUcam3. Dibuktikan juga dengan nilai x dan y pada setiap obyek berwarna, ketika dibandingkan dengan nilai x dan y jangkauan penglihatan sensor kamera CMUcam3 seperti pada gambar 3-18, terlihat sesuai. Sebagai contoh, pada percobaan pertama yang ditampilkan pada LCD untuk warna merah, nilai yang peroleh adalah $x=39$ $y=52$ $z=19$. Nilai tersebut ketika dibandingkan dengan gambar 3-18, dapat diartikan bahwa obyek bola berwarna merah berada pada area tengah pandangan sensor kamera CMUcam3, sama seperti yang ditampilkan oleh CMUcam3 *frame grabber* pada gambar 4-5.

Tabel 4-2 juga memperlihatkan bahwa sistem kontroller berhasil memperoleh dan mengolah data hasil dari *image processing* berupa penjejukan obyek berwarna, yang dilakukan oleh sensor kamera CMUcam3. Hal tersebut dibuktikan ketika membandingkan data yang ditransfer oleh sensor kamera CMUcam3 kepada komputer dan sistem kontroller, hasilnya sama. Hanya yang membedakan pada komputer menampilkan juga seluruh hasil dari perintah TC. Sebagai contoh, pada percobaan pertama untuk warna merah, Hasil dari perintah TC yang ditampilkan oleh komputer adalah T 39 52 35 44 44 61 18 117. Sedangkan pada sistem kontroller hanya menampilkan nilai mx, my dan pixels, atau 39 52 18. Hal tersebut dikarenakan nilai mx, my dan pixels, sudah menunjukkan posisi dari suatu obyek berwarna dan lebih sering digunakan untuk logika pemrograman dibandingkan nilai-nilai yang lain. Untuk contoh progamnya dapat dilihat pada gambar 3-17.

Tabel 4-2 Percobaan Pengjejakan Obyek Berwarna

Warna	Hasil dari Sistem Kontroller (LCD)	Hasil dari Komputer (<i>software serial watcher</i>)
Merah		
Hijau		
Biru		

Tabel 4-2 Percobaan Pengejekan Obyek Berwarna (lanjutan)

Warna	Hasil dari Sistem Kontroller (LCD)	Hasil dari Komputer (<i>software serial watcher</i>)
Oranye		

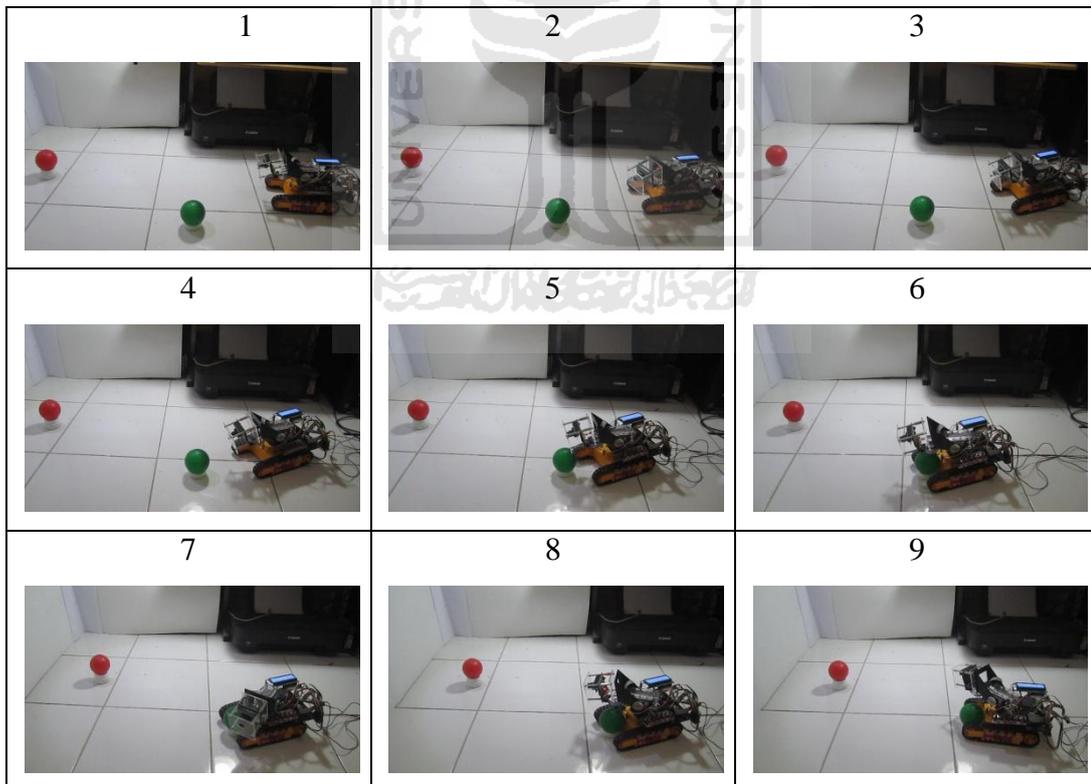
Adanya perbedaan hasil dari percobaan pada setiap posisi dari obyek berwarna yang terdapat pada tabel 4-2, disebabkan oleh faktor cahaya lampu yang dapat berubah-ubah, meskipun untuk mata pada manusia tidak merasakannya. Seperti contohnya nilai posisi obyek berwarna oranye yang ditampilkan oleh sistem kontroller pada LCD, terlihat berbeda-beda. Dari nilai percobaan pertama adalah $mx=36$ $my=54$ $pixels=123$. Kemudian nilai percobaan kedua $mx=36$ $my=53$ $pixels=130$ dan nilai percobaan $mx=36$ $my=54$ $pixels=124$. Ketiga nilai yang diperoleh terlihat berbeda, namun masih dapat ditoleransi dan dianggap benar karena selisih perbedaan satu nilai dengan nilai yang lain tidak terlalu tinggi.

4.3 Kemampuan Robot Mencari dan Mendapatkan Target

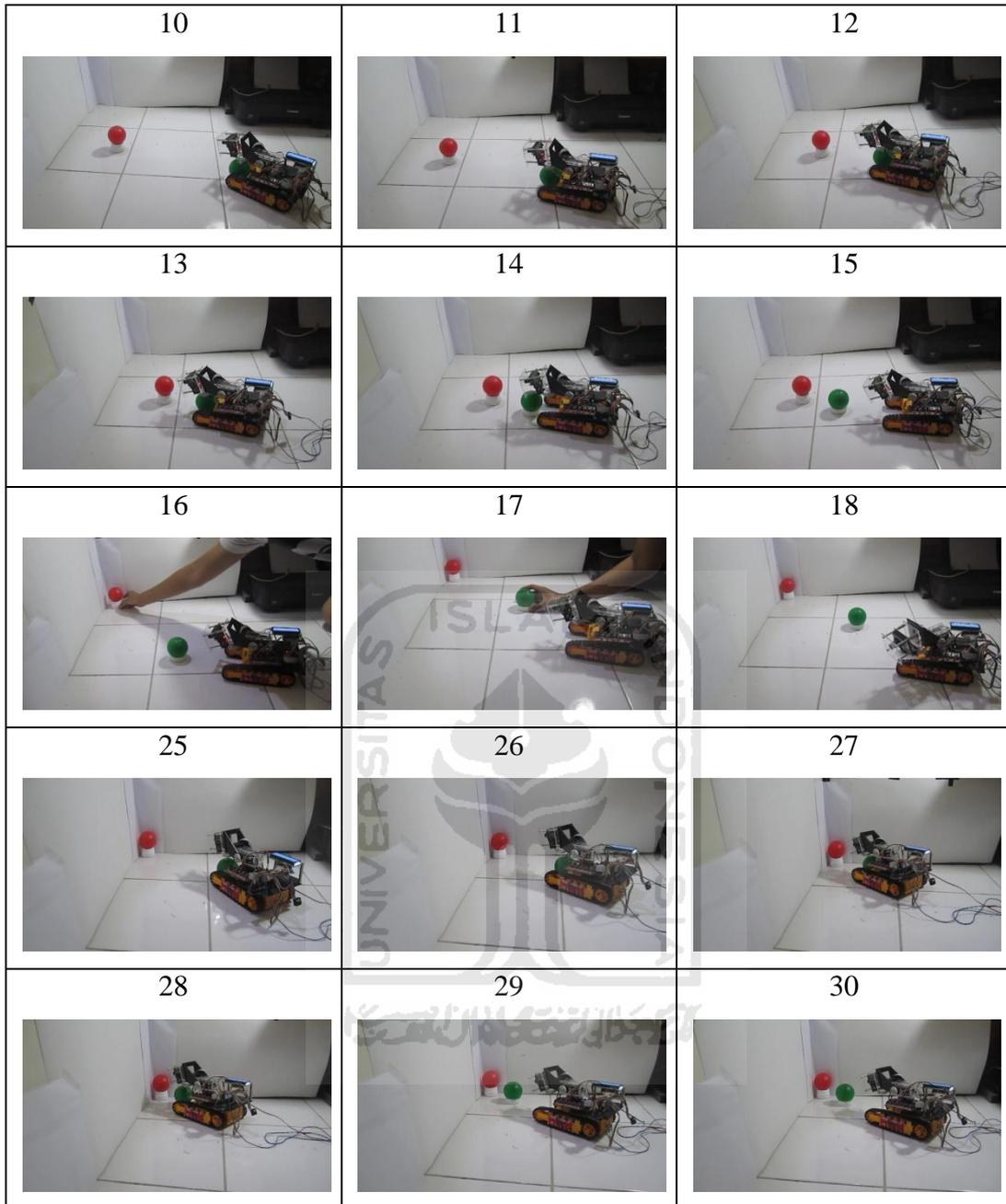
Percobaan ini bertujuan untuk memperlihatkan dan membuktikan bahwa dengan memanfaatkan kemampuan pengklasifikasian warna dan penjejakan obyek berwarna dari sensor kamera CMUcam3, maka robot dua kaki atau robot *humanoid* secara otonom mampu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Percobaan dilakukan dua tahap dengan menggunakan dua robot. Percobaan pertama menggunakan robot beroda, dan percobaan kedua

menggunakan robot yang sesungguhnya yaitu robot dua kaki atau robot *humanoid*.

Percobaan pertama dengan menggunakan robot beroda, dimaksudkan untuk mencoba dua hal. Yaitu hal yang pertama adalah, mencoba kemampuan dari sensor kamera CMUcam3 yang berupa pengklasifikasian warna dan penjejakan obyek berwarna, untuk menavigasi robot dalam mencari dan mendapatkan targetnya. Kemudian hal yang kedua adalah, mencoba untuk melihat tingkat keberhasilan langkah-langkah robot untuk menyelesaikan tugasnya seperti pada gambar 3-15, ketika diterapkan pada sebuah robot. Setelah kedua hal tersebut dicoba pada robot beroda yang relatif lebih sederhana dibandingkan dengan robot dua kaki atau robot *humanoid*, dan robot beroda terbukti berhasil mencari kemudian mendapatkan targetnya. Maka dua hal tersebut baru dapat digunakan pada robot dua kaki atau robot *humanoid*, dan percobaan kedua dapat segera dilakukan. Hasil percobaan yang pertama dapat dilihat pada gambar 4-6.



Gambar 4-6 Robot Beroda Mencari dan Mendapatkan Target



Gambar 4-6 Robot Beroda Mencari dan Mendapatkan Target (lanjutan)

Gambar 4-6 memperlihatkan bahwa robot beroda mampu untuk menyelesaikan tugasnya, yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Gambar 4-6 juga memperlihatkan langkah-langkah robot beroda dalam menyelesaikan tugasnya. Dimulai dari yang pertama, untuk kolom nomor 1, 2 dan 3 memperlihatkan langkah pencarian robot terhadap posisi dari target pertama, yaitu obyek bola berwarna hijau. Langkah pencarian obyek pertama ini,

merupakan gerakan dari kepala robot yang mampu berputar ke kanan kemudian ke kiri, selanjutnya naik dan turun. Seperti menggeleng-gelengkan atau mengangguk-anggukan kepala pada manusia. Langkah tersebut akan selalu dilakukan sampai posisi obyek pertama ditemukan. Kekurangan dari masing-masing langkah tersebut adalah jangkauan gerakannya yang hanya mempunyai sudut 50° dari posisi awal. Sehingga hal tersebut akan menyebabkan ketika target berada tidak pada jangkauan pencarian dari robot, maka robot tidak akan dapat menemukan targetnya. Namun setelah robot menemukan dan mengetahui posisi dari targetnya, maka robot akan memposisikan dirinya tegak lurus terhadap target tersebut, seperti kolom nomor 4. Kemudian setelah robot tegak lurus dengan target, maka langkah pada kolom 5 dan 6 akan dilakukan. Langkah tersebut adalah menghampiri target dan kemudian mengambilnya. Langkah pengambilan adalah langkah memasukan bola ke dalam rongga yang ada diantara roda dari robot. Hal tersebut dimaksudkan agar target pertama mudah untuk diarahkan atau dibawa ke target yang kedua, yaitu obyek bola berwarna merah. Atau pada percobaan pertama ini, diartikan sebagai gawang.

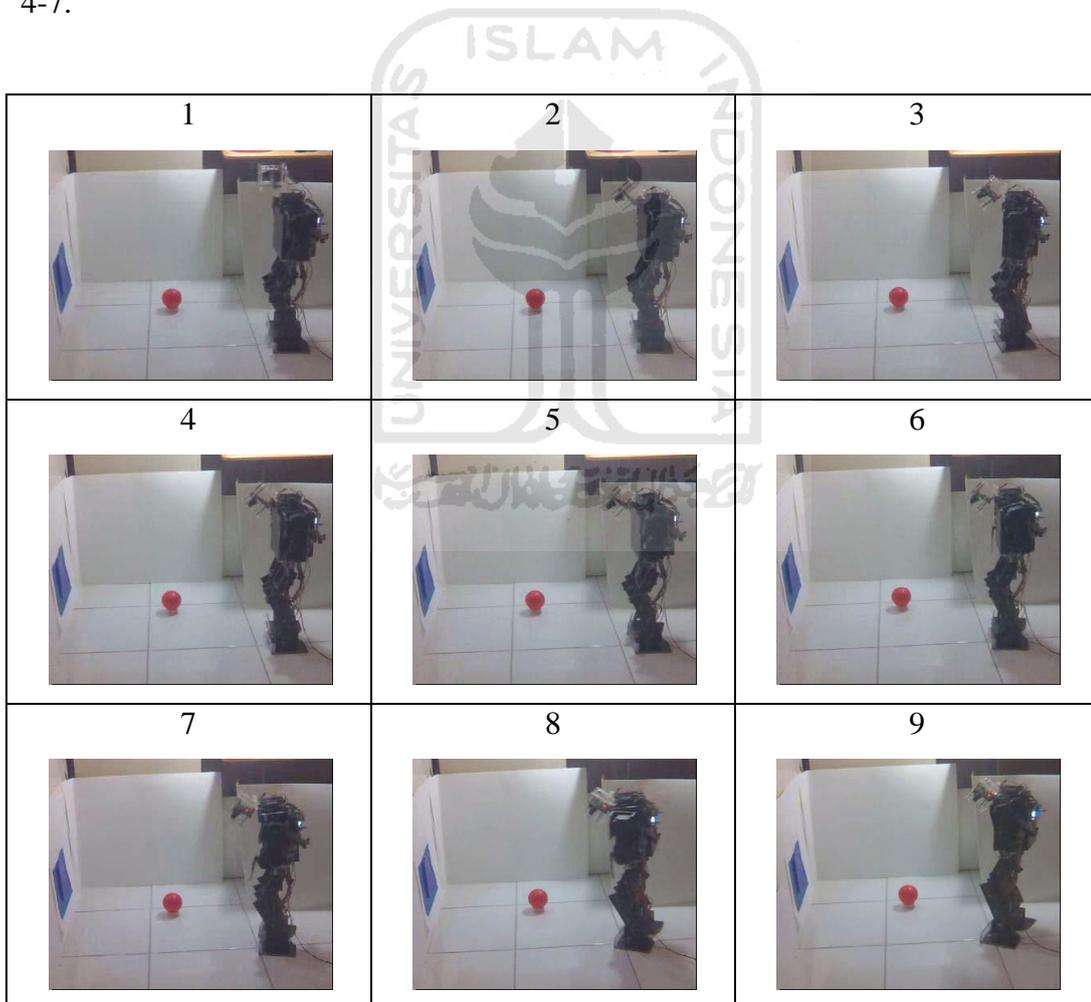
Kemudian setelah robot berhasil mendapatkan target pertamanya, maka pencarian target kedua dapat dilakukan. Prinsip mencari hingga mendapatkan target kedua hampir sama seperti target pertama, hal tersebut ditunjukkan oleh kolom nomor 7 sampai dengan 12. Hanya yang membedakan adalah, setelah posisi target kedua ditemukan, robot hanya menghampirinya untuk meletakkan target pertama di dekat target kedua, seperti ditunjukkan oleh kolom nomor 14,15 dan 16. Hal tersebut dapat diartikan bahwa bola telah masuk pada area gawang, atau dapat disebut sebagai *goal*.

Untuk kolom nomor 16 sampai dengan 30, gerakan keseluruhan robot terlihat hampir sama seperti kolom nomor 1 sampai dengan nomor 15. hanya yang membedakan, untuk kolom nomor 16 sampai dengan 30 dimaksudkan untuk menunjukkan bahwa robot beroda mampu meletakkan target satu selalu berada di dekat target dua, walaupun dipindah ke posisi yang berbeda.

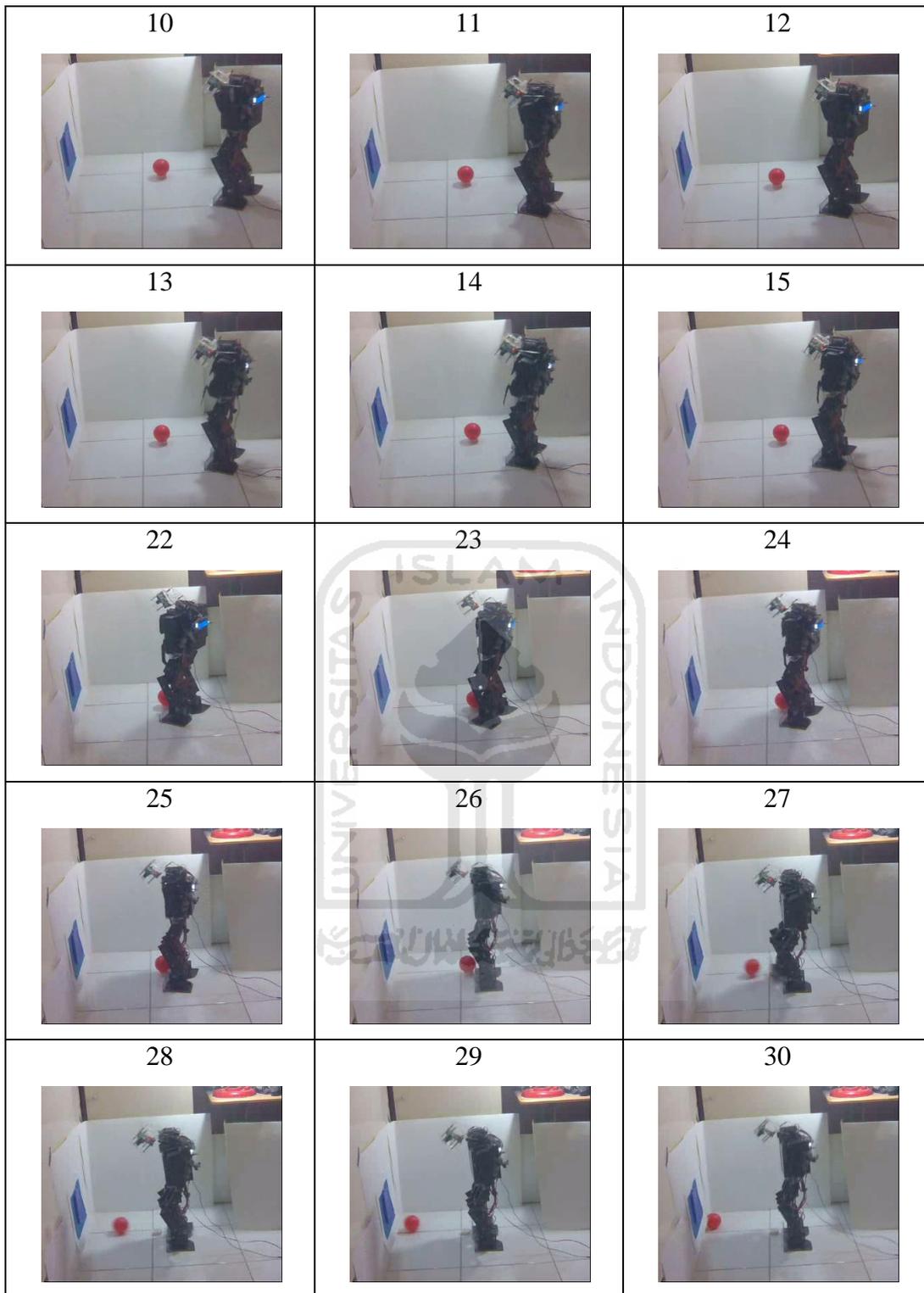
Dari pemaparan langkah-langkah penyelesaian tugas robot beroda di atas, dapat disimpulkan bahwa untuk langkah-langkah penyelesaian tugas robot beroda sedikit berbeda dengan langkah-langkah penyelesaian tugas robot seperti

pada gambar 3-15. Perbedaannya hanya ketika robot beroda telah mendapatkan target pertamanya yaitu bola berwarna hijau, maka bola tidak ditendang ke arah target kedua atau gawang. Namun robot beroda akan membawa bola tersebut mendekati gawang hingga jarak yang paling dekat dengan gawang. Prinsip penyelesaian tugas robot tersebut sedikit berbeda namun mempunyai maksud akhir yang sama, yaitu warna target satu dan warna target dua berdekatan atau warna target satu berada pada area warna target kedua.

Setelah melihat hasil dari percobaan pertama yang membuktikan bahwa robot beroda secara otonom mampu mencari dan mendapatkan targetnya. Maka selanjutnya percobaan kedua dengan menggunakan robot dua kaki atau robot *humanoid*, dapat dilakukan. Hasil dari percobaan kedua dapat dilihat pada gambar 4-7.



Gambar 4-7 Robot Dua Kaki atau Robot *Humanoid* Mencari dan Mendapatkan Target



Gambar 4-7 Robot Dua Kaki atau Robot *Humanoid* Mencari dan Mendapatkan Target (lanjutan)

Gambar 4-7 memperlihatkan bahwa robot dua kaki atau robot *humanoid* secara otonom mampu mencari dan mendapatkan targetnya. Gambar 4-7 juga membuktikan bahwa sensor kamera CMUcam3 dapat digunakan sebagai sensor penglihatan pada robot dua kaki atau robot *humanoid* seperti mata pada manusia. Dengan menggunakan kemampuan sensor kamera CMUcam3 yang berupa pengklasifikasian warna, maka robot dapat mengetahui dan membedakan warna dari setiap targetnya, yaitu warna merah sebagai warna target pertama, dan warna biru sebagai warna target kedua. Pada percobaan ini warna merah dianggap sebagai warna dari obyek bola, dan warna biru dianggap sebagai warna dari obyek gawang. Kemudian dengan kemampuan dari sensor kamera CMUcam3 yang berupa penjejakan obyek berwarna, maka robot dapat mengetahui posisi dari obyek bola dan gawang. Sehingga robot mampu untuk memasukan bola kedalam area gawang.

Dapat dilihat bahwa langkah-langkah pergerakan robot untuk menyelesaikan tugasnya pada Gambar 4-7, sama seperti langkah-langkah penyelesaian tugas robot pada gambar 3-15. Hal tersebut terbukti ketika melihat satu per satu gambar di setiap kolom pada Gambar 4-7. Dari yang pertama, untuk kolom nomor 1 dan 2 memperlihatkan langkah pencarian robot terhadap posisi dari bola. Sama seperti robot beroda, untuk robot berkaki atau robot *humanoid* ketika tidak menemukan posisi dari bola, maka robot akan terus mencari hingga posisi bola ditemukan. Untuk gerakannya hampir sama seperti robot beroda, hanya yang membedakan pada robot dua kaki atau robot *humanoid*, untuk memindahkan pandangan ke kanan dan ke kiri, dilakukan dengan cara memutar pinggul dari robot, dan dengan jangkauan putaran masing-masing gerakan yang hanya 40° dari posisi awal. Kemudian perbedaan yang lain adalah, pada robot dua kaki atau robot *humanoid*, langkah pencarian obyek bola diawali dari melihat ke atas, kemudian ke bawah, dan selanjutnya ke kanan dan ke kiri.

Setelah posisi bola ditemukan, maka selanjutnya robot akan memposisikan dirinya tegak lurus terhadap bola, dan kemudian mendekatinya sampai jangkauan tendang robot terhadap bola dapat dilakukan, seperti terlihat pada kolom nomor 3 sampai dengan 21. Setelah kondisi tersebut terpenuhi, maka robot akan melakukan langkah selanjutnya, yaitu mencari posisi dari gawang

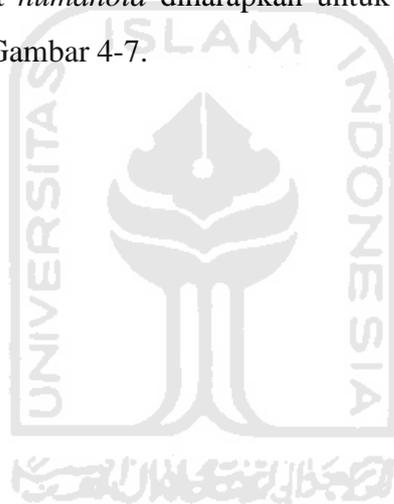
seperti kolom nomor 22 sampai dengan 24. Pencarian posisi dari gawang sama seperti pencarian posisi dari bola, yaitu dengan gerakan mencari yang diawali dari melihat ke atas kemudian ke bawah, dan selanjutnya ke kanan dan ke kiri. Kemudian setelah posisi gawang diketahui, maka selanjutnya robot akan memposisikan dirinya tegak lurus terhadap bola dan gawang, dan selanjutnya robot akan menendang bola ke arah gawang seperti kolom nomor 25 sampai dengan 29. Gambar pada kolom nomor 25 memperlihatkan bahwa ketika robot akan melakukan tendangan terhadap bola ke arah gawang, maka sebelumnya robot akan melihat kembali posisi dari bola, dimana ketika bola terlihat berada pada area jangkauan tendang kaki kiri, maka robot akan menendang dengan kaki kiri. Sebaliknya ketika bola terlihat berada pada area jangkauan tendang kaki kanan, maka robot akan menendang dengan kaki kanan. Dan sampai akhirnya, untuk kolom nomor 30 memperlihatkan bahwa bola sudah berada pada area gawang atau dapat disebut sebagai *goal*. Dari pernyataan-pernyataan diatas, dapat disimpulkan bahwa gambar 4-7 juga membuktikan langkah-langkah penyelesaian tugas robot pada gambar 3-15, dapat diterapkan pada robot dua kaki atau robot *humanoid* untuk menyelesaikan tugasnya dengan baik.

Walaupun robot kaki atau robot *humanoid* mampu mencari dan mendapatkan targetnya, atau dapat dinyatakan bahwa robot dua kaki atau robot *humanoid* dapat bermain sepak bola. Namun ada beberapa kekurangan dari robot yang dapat menghambat dalam melakukan itu semua. Kekurangan yang pertama adalah daya lihat dari robot yang sangat terbatas. Walaupun sudah diantisipasi dengan kemampuan dari kepala robot yang dapat naik turun, dan kemampuan dari pinggul yang dapat berputar ke kanan dan ke kiri untuk memperluas sudut pandangan dari robot. Namun robot hanya mampu melihat suatu target dengan jangkauan sudut tidak lebih dari 90° dan dengan jarak maksimal 60 cm. sehingga hal tersebut sering membuat robot kehilangan atau tidak menemukan posisi dari targetnya, ketika mencari ataupun menghampiri.

Kemudian kekurangan yang kedua adalah, terwujudnya ketidakpresisian pergerakan robot ketika bergerak maju, belok kanan, belok kiri, tendang kanan maupun tendang kiri. Sebagai contoh, Hal tersebut akan membuat ketidaksengajaan robot menyenggol bola ketika melakukan suatu gerakan,

sehingga bola akan bergerak liar dan robot akan lebih sering kehilangan posisi dari bola tersebut. Contoh yang lain adalah, robot akan bergerak berlebih atau sangat kurang, sehingga ketika robot berjalan menuju bola atau memposisikan dirinya tegak lurus terhadap bola, akan terlihat kesulitan.

Sehingga dapat disimpulkan bahwa, ketika robot dapat menyelesaikan tugasnya seperti pada Gambar 4-7. Berarti pada saat itu posisi bola berada dalam jangkauan penglihatan robot pada saat dicari. Kemudian pergerakan robot untuk memposisikan dirinya tegak lurus dengan bola, gawang ataupun keduanya adalah pas. Hingga saat robot akan menendang bola, posisi dari bola terhadap jangkauan tendang robot juga adalah pas. Sehingga dapat disimpulkan lagi bahwa, dari beberapa hal dalam paragraf ini dapat menjadi syarat yang harus terpenuhi, ketika robot dua kaki atau robot *humanoid* diharapkan untuk menyelesaikan tugasnya dengan baik seperti pada Gambar 4-7.



BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari penelitian terhadap sensor kamera CMUcam3, dan hasil dari percobaan-percobaan yang telah dilakukan. Maka dapat ditarik kesimpulan bahwa, robot dua kaki atau robot *humanoid* mampu secara otonom menyelesaikan tugasnya, yaitu mencari dan mendapatkan suatu target yang berupa obyek berwarna. Namun agar robot dapat menyelesaikan tugasnya dengan baik, maka faktor-faktor yang mendukung itu semua harus terpenuhi. Faktor-faktor tersebut adalah, kinerja dari sensor kamera CMUcam3 yang baik, agar mendapatkan hasil yang akurat. Kemudian gerakan dari robot yang presisi, agar tidak terjadi kesalahan dalam posisi. Dan program keseluruhan dari robot yang tepat, agar dapat merealisasikan fungsi robot yang diinginkan.

5.2 Saran

Untuk mendapatkan hasil yang lebih maksimal pada penelitian yang selanjutnya, maka berikut ini adalah saran dan perbaikan yang perlu dipertimbangkan.

1. Untuk mendapatkan hasil *image processing* dari sensor kamera CMUcam3 yang akurat, maka pencahayaan yang baik harus terpenuhi.
2. Sensor kamera CMUcam3 memiliki sensitifitas pada intensitas cahaya dan kontras warna. Oleh karena itu, perlu dilakukan kalibrasi nilai warna (RGB) untuk melakukan penjejakan obyek berwarna di setiap tempat yang berbeda.
3. Agar robot dapat mengikuti warna lebih cepat, maka diperlukan sensor kamera yang memiliki kecepatan dan resolusi yang lebih tinggi.
4. Penambahan sensor keseimbangan pada robot dua kaki atau robot *humanoid* perlu dipertimbangkan.
5. Dalam pemrograman, penyimpanan nilai posisi terakhir dari suatu obyek berwarna sangat penting dilakukan. Karena nilai tersebut dapat digunakan

oleh robot, untuk mengingat kembali posisi terakhir dari suatu obyek berwarna saat kehilangan. Sehingga robot dapat menemukan kembali obyek berwarna tersebut.



DAFTAR PUSTAKA

- Adi, A.N. (2010). *Mekatronika*. Graha Ilmu. Yogyakarta.
- Andrianto, H. (2008). *Pemrograman Mikrokontroler AVR ATMEGA16*. Informatika, Bandung.
- Bejo, A. (2008). *C&AVR Rahasia Kemudahan Dalam Mikrokontroler ATmega8535*. Graha Ilmu. Yogyakarta.
- Datasheet ATmega8535.
- Datasheet ATmega16.
- Datasheet CMUcam2.
- Datasheet CMUcam3.
- Endik, S. (2011). *Perancangan Mekanik Robot Dua Kaki*. Skripsi. tidak diterbitkan. Teknik Mesin Universitas Islam Indonesia Yogyakarta.
- Fu, et al. (1987). *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill, ISBN:0-07-100421-1. New York.
- Halim, F.A., & Tjen, A. (2006). Perancangan Robot Humanoid dengan Kemampuan Pergerakan Autonomous Mengikuti Obyek, *Seminar Nasional Sistem dan Informatika 2007*. Bali. Indonesia.
- Haryanto, M.A., & Prasetyanto, W.A. (2008). *Pemrograman Bahasa C untuk Mikrokontroler ATMEGA8535*. Andi Yogyakarta. Yogyakarta.
- Murni, A., & Cahyati, D. (2003). *Pengolahan Citra Digital : Peningkatan Mutu Domain Spesial*. Jurnal.
- Prasetyo, B., Ningrum, S.E., & Alasiry, A.H. (2009). *Implementasi Metode Virtual Force Field Untuk Kontrol Pergerakan Autonomous Mobile Robot Pada Aplikasi Soccer Robot*. Jurnal.
- Pitowarno, E. (2006). *ROBOTIKA : Desain, Kontrol dan Kecerdasan*. Andi Yogyakarta. Yogyakarta.
- Salfikar, I., Sulistijono, I.A., & Setiaji, O. (2009). *Humanoid Soccer Robot : System Kendali Visual Menggunakan Metode Fuzzy Sebagai Penjejak Objec*. Jurnal.
- Sigit, R. (2007). *Robotika, Sensor & Atuator Persiapan Lomba Kontes Robot Indonesia dan Robot Cerdas Indonesia*. Graha Ilmu. Yogyakarta.

- Sriyanto, J. (2011). *Perancangan Sistem Gerak Robot Dua Kaki*. Skripsi. tidak diterbitkan. Teknik Mesin Universitas Islam Indonesia Yogyakarta.
- Permadi, R., & Murti, M.A. (2006). Aplikasi Image Processing Dalam Penggunaan Kamera Sebagai Sensor Api. *Seminar Nasional Sistem dan Informatika 2006*. Bali. Indonesia.
- Petrus, P. (2007). *Pengolahan Citra*. Bahan Mata Kuliah Komputer Grafis. UKDW.
- Putra, T.S. (2008). *Perancangan Robot Dua Kaki dengan Tiga Derajat Kebebasan*. Skripsi. tidak diterbitkan. Teknik Mesin Universitas Islam Indonesia Yogyakarta.



LAMPIRAN



Program Robot

```
/*
Chip type      : ATmega8535
Program type   : Application
Clock frequency : 8,000000 MHz
Memory model   : Small
External SRAM size : 0
Data Stack size : 512
*/

#include <mega32.h>
#include <delay.h>
#include <stdio.h>

// Alphanumeric LCD Module functions
#asm
.equ __lcd_port=0x1B ;PORTA
#endasm
#include <lcd.h>

#pragma used+

#define RXB8 1
#define TXB8 0
#define UPE 2
#define OVR 3
#define FE 4
#define UDRE 5
#define RXC 7

#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<OVR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)

// USART Receiver buffer
#define RX_BUFFER_SIZE 32

char rx_buffer[RX_BUFFER_SIZE];
unsigned char rx_index;
bit rx_flag=0;

// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
    char status;
    status=UCSRA;
    if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)
    {
```

```

rx_buffer[rx_index]=UDR;
if(rx_buffer[rx_index]!=':')
{
    rx_buffer[rx_index]=0;
    rx_index=0;
    rx_flag=1;
}
else if(rx_buffer[rx_index]==13)rx_buffer[rx_index]=0;
if (++rx_index == RX_BUFFER_SIZE)rx_index=0;
};
}

```

```

void clear_buffer_rx()
{
    char i;
    rx_index=0;
    for(i=0;i<RX_BUFFER_SIZE;i++)rx_buffer[i]=0;
    rx_flag=0;
}

```

```
char lcd_buffer[32];
```

```

unsigned char x[4];
unsigned char y[4];
unsigned char z[4];
unsigned char x0[4];
unsigned char y0[4];
unsigned char x1[4];
unsigned char y1[4];
unsigned char z0[4];
bit a,b,c,d,h,k;
char e,f,g,j;

```

```

eeprom unsigned char rl[4];
eeprom unsigned char rh[4];

```

```

eeprom unsigned char gl[4];
eeprom unsigned char gh[4];

```

```

eeprom unsigned char bl[4];
eeprom unsigned char bh[4];

```

```
unsigned char rgb_mean[3];
```

```

void send_com(char flash *com)
{
    char i;
    lcd_clear();lcd_putsf(com);

    clear_buffer_rx();

    printf(com);
}

```



```

putchar(13);

while(rx_flag==0);

lcd_gotoxy(0,1);
lcd_putsf(" ");
lcd_puts(rx_buffer);
delay_ms(100);
}

void gm()
{
clear_buffer_rx();
printf("gm");
putchar(13);

while(rx_flag==0);

rgb_mean[0]=rx_buffer[2];
rgb_mean[1]=rx_buffer[3];
rgb_mean[2]=rx_buffer[4];

lcd_clear();
sprintf(lcd_buffer,"R%3d G%3d B%3d",rgb_mean[0],rgb_mean[1],rgb_mean[2]);
lcd_gotoxy(0,0);lcd_putsf(" Get Mean ");
lcd_gotoxy(0,1);lcd_puts(lcd_buffer);
//lcd_puts(rx_buffer);
}

void gt(unsigned char n)
{
send_com("tw");

clear_buffer_rx();
printf("gt");
putchar(13);

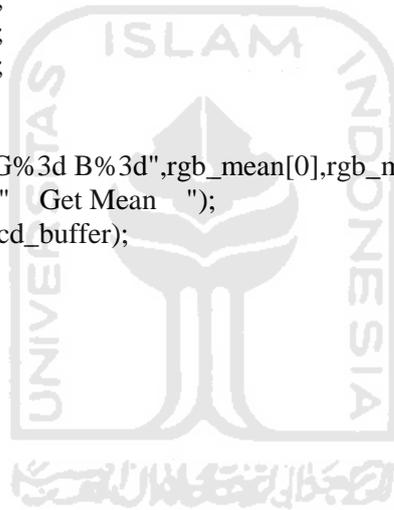
while(rx_flag==0);

rl[n]=rx_buffer[1];
gl[n]=rx_buffer[2];
bl[n]=rx_buffer[3];

rh[n]=rx_buffer[4];
gh[n]=rx_buffer[5];
bh[n]=rx_buffer[6];

lcd_clear();
sprintf(lcd_buffer,"r%3d g%3d b%3d",rl[n],gl[n],bl[n]);
lcd_gotoxy(0,1);lcd_puts(lcd_buffer);
sprintf(lcd_buffer,"R%3d G%3d B%3d",rh[n],gh[n],bh[n]);
lcd_gotoxy(0,0);lcd_puts(lcd_buffer);
}

```



```

void tc(unsigned char n)
{
    clear_buffer_rx();
    printf("tc %d %d %d %d",rl[n],gl[n],bl[n],15);
    putchar(13);

    while(rx_flag==0);

    x[n]=rx_buffer[2]; //x posisi
    y[n]=rx_buffer[3]; //y posisi
    z[n]=rx_buffer[8]; //jumlah pixel

    // x0[n]=rx_buffer[4]; //
    // y0[n]=rx_buffer[5]; //
    // x1[n]=rx_buffer[6]; //
    // y1[n]=rx_buffer[7]; //

    // z0[n]=(x1[n]-x0[n])+(y1[n]-y0[n]);
    // //(-1.55*z0[n])+79;

    lcd_clear();
    sprintf lcd_buffer,"X%3d Y%3d Z%3d",x[n],y[n],z[n];
    lcd_gotoxy(0,0);
    lcd_puts lcd_buffer);
    sprintf lcd_buffer,"r%3d g%3d b%3d",rl[n],gl[n],bl[n]);
    lcd_gotoxy(0,1);
    lcd_puts lcd_buffer);

    //lcd_puts(rx_buffer);
}

```

```

void orange()
{
    char n=0;
    clear_buffer_rx();
    printf("tc %d %d %d %d",122,34,16,15);
    putchar(13);

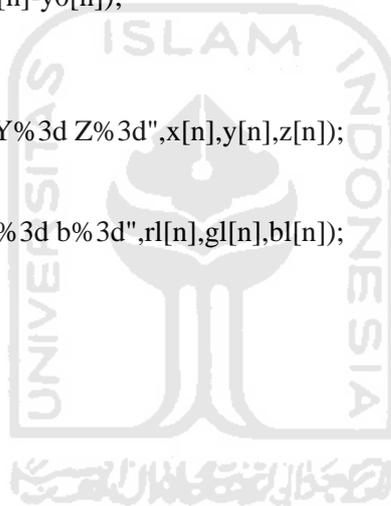
    while(rx_flag==0);

    x[n]=rx_buffer[2];
    y[n]=rx_buffer[3];
    z[n]=rx_buffer[8];

    x0[n]=rx_buffer[4]; //
    y0[n]=rx_buffer[5]; //
    x1[n]=rx_buffer[6]; //
    y1[n]=rx_buffer[7]; //

    z0[n]=(x1[n]-x0[n])+(y1[n]-y0[n]);
    //(-1.55*z0[n])+79;

```



```

    lcd_clear();
    sprintf(lcd_buffer,"X%3d Y%3d Z%3d",x[n],y[n],z[n]);
    lcd_gotoxy(0,0);
    lcd_puts(lcd_buffer);
    sprintf(lcd_buffer,"z%3d %2d",z0[n],f);
    lcd_gotoxy(0,1);
    lcd_puts(lcd_buffer);
}

void merah()
{
    char n=0;
    clear_buffer_rx();
    printf("tc %d %d %d %d",77,25,16,15); //126,16,16
    putchar(13);

    while(rx_flag==0);

    x[n]=rx_buffer[2];
    y[n]=rx_buffer[3];
    z[n]=rx_buffer[8];

    x0[n]=rx_buffer[4]; //
    y0[n]=rx_buffer[5]; //
    x1[n]=rx_buffer[6]; //
    y1[n]=rx_buffer[7]; //

    z0[n]=(x1[n]-x0[n])+(y1[n]-y0[n]);
    //(-1.55*z0[n])+79;

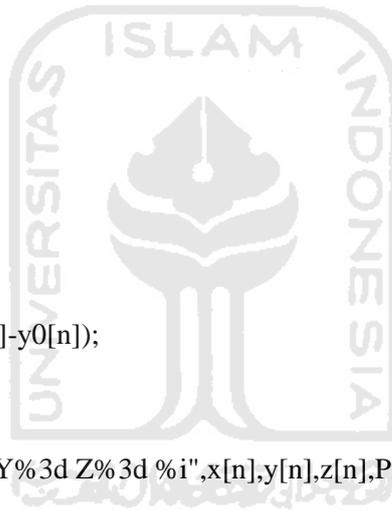
    lcd_clear();
    sprintf(lcd_buffer,"X%3d Y%3d Z%3d %i",x[n],y[n],z[n],PIND.7);
    lcd_gotoxy(0,0);
    lcd_puts(lcd_buffer);
    sprintf(lcd_buffer,"z%3d %i%i%i %2d %i
%i%i",z0[n],PINB.0,PINB.1,PINB.2,f,j,PIND.5,PIND.6);
    lcd_gotoxy(0,1);
    lcd_puts(lcd_buffer);
}

void biru()
{
    char n=1;
    clear_buffer_rx();
    printf("tc %d %d %d %d",16,22,77,15);
    putchar(13);

    while(rx_flag==0);

    x[n]=rx_buffer[2];
    y[n]=rx_buffer[3];
    z[n]=rx_buffer[8];

```



```

x0[n]=rx_buffer[4]; //
y0[n]=rx_buffer[5]; //
x1[n]=rx_buffer[6]; //
y1[n]=rx_buffer[7]; //

z0[n]=(x1[n]-x0[n])+(y1[n]-y0[n]);
//(-1.55*z0[n])+79;

lcd_clear();
sprintf(lcd_buffer,"X%3d Y%3d Z%3d %i",x[n],y[n],z[n],PIND.7);
lcd_gotoxy(0,0);
lcd_puts(lcd_buffer);
sprintf(lcd_buffer,"z%3d %i%i%i %2d %i
%i%i",z0[n],PINB.0,PINB.1,PINB.2,f,j,PIND.5,PIND.6);
lcd_gotoxy(0,1);
lcd_puts(lcd_buffer);
}
void hijau()
{
char n=0;
clear_buffer_rx();
printf("tc %d %d %d %d",16,120,30,15);
putchar(13);

while(rx_flag==0);

x[n]=rx_buffer[2];
y[n]=rx_buffer[3];
z[n]=rx_buffer[8];

x0[n]=rx_buffer[4]; //
y0[n]=rx_buffer[5]; //
x1[n]=rx_buffer[6]; //
y1[n]=rx_buffer[7]; //

z0[n]=(x1[n]-x0[n])+(y1[n]-y0[n]);
//(-1.55*z0[n])+79;

lcd_clear();
sprintf(lcd_buffer,"X%3d Y%3d Z%3d",x[n],y[n],z[n]);
lcd_gotoxy(0,0);
lcd_puts(lcd_buffer);
sprintf(lcd_buffer,"z%3d %d ",z0[n],f);
lcd_gotoxy(0,1);
lcd_puts(lcd_buffer);
}

#define servo1_pin PORTD.2
#define servo2_pin PORTD.3
unsigned int pwm1=1600;
unsigned int pwm2=1480;
unsigned char servo_index=0;

```



```

// Timer 1 output compare A interrupt service routine
interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{

    if(servo_index==0)
    {
        servo1_pin=1;
        OCR1A=pwm1;
        servo_index=1;
    }
    else if(servo_index==1)
    {
        servo1_pin=0;           // Interup Timer 1
        servo2_pin=1;
        OCR1A=pwm2;
        servo_index=2;
    }

    else if(servo_index==2)
    {
        servo2_pin=0;
        OCR1A=2000;
        servo_index=0;
    }
}

void tegak(){PORTC.1=1;PORTC.6=1;PORTC.7=1;}//tegak
void maju(){PORTC.1=0;PORTC.6=0;PORTC.7=0;}// maju
void kiri(){PORTC.1=1;PORTC.6=0;PORTC.7=0;}// kiri
void kanan(){PORTC.1=0;PORTC.6=0;PORTC.7=1;}// kanan
void lihat_kiri(){PORTC.1=1;PORTC.6=0;PORTC.7=0;}// lihat kiri
void lihat_kanan(){PORTC.1=0;PORTC.6=1;PORTC.7=1;}// lihat kanan
void tendang_kanan(){PORTC.1=0;PORTC.6=1;PORTC.7=0;}// tendang kanan
void tendang_kiri(){PORTC.1=1;PORTC.6=0;PORTC.7=1;}// tendang kiri

void main(void)
{
    PORTB=0x07;
    DDRC=0xFF;
    PORTD=0xE0;
    DDRD=0x0C;
    PORTC.7=1;PORTC.6=1;PORTC.1=1;

    // Timer/Counter 1 initialization
    // Clock source: System Clock
    // Clock value: 1382.400 kHz
    // Mode: CTC top=OCR1A
    // OC1A output: Discon.
    // OC1B output: Discon.
    // Noise Canceler: Off
    // Input Capture on Falling Edge
    // Timer 1 Overflow Interrupt: Off

```

```

// Input Capture Interrupt: Off
// Compare A Match Interrupt: On
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x0A;
OCR1A=5000;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x10;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud rate: 9600
UCSRA=0x00;
UCSRB=0x98;
UCSRC=0x86;
UBRRH=0x00;
// UBRRL=0x5C; //14.318Mhz
// UBRRL=0x4D; //12Mhz
// UBRRL=0x47; //11.0592Mhz
UBRRL=0x33; //8Mhz
// UBRRL=0x67; //16Mhz

ACSR=0x80;
// LCD module initialization dengan jumlah kolom perbaris 16
lcd_init(16);

// Global enable interrupts
#asm("sei")

lcd_clear();lcd_putsf(" * Dhika * ");
lcd_gotoxy(0,1);lcd_putsf("Teknik Mesin UII");
//printf("Dhika Teknik Mesin UII");
delay_ms(1000);

lcd_clear();lcd_putsf("Cek Kamera.....");
delay_ms(200);
send_com("rs");

lcd_clear();lcd_putsf("Kamera Ok");delay_ms(500);

//send_com("om 0 67");delay_ms(100);
send_com("pm 1");delay_ms(100);
send_com("rm 3");delay_ms(100);

while (1)
{
if(PIND.7==0){e=0;delay_ms(200);while(PIND.7==0);gt(0);delay_ms(500);e=1;};

```

```

//if(PINC.1==0){e=1;f=0;pwm1=1600;};
if(PIND.6==0){e=2;f=1;pwm1=1600;g=0;j=0;h=0;k=0;};
if(PIND.5==0){e=3;f=1;pwm1=1600;g=0;j=0;h=0;k=0;};

if(e==1){tc(0);f=0;pwm1=1600;};
if(e==2){merah(0);};
if(e==3){biru(0);};

if(pwm1>1800){pwm1=1800;};
if(pwm1<1100){pwm1=1100;};

    if(f==0){e=2;tegak(0);g=0;j=0;h=0;k=0;}

    else if(f==1){

        if(x[0]<11&&g==0){j=1;tegak(0);g=1;};
        if(x[0]<11&&g==1){j=2;lihat_kanan(0);if(PINB.0==0&&PINB.1==1){g=2;}};
        if(x[0]<11&&g==2){j=3;lihat_kiri(0);if(PINB.0==1&&PINB.1==0){g=3;}};

if(x[0]<11&&g==3&&pwm1<1800){j=4;tegak(0);pwm1=pwm1+25;if(pwm1>1799){g=4
};};
    if(x[0]<11&&g==4&&pwm1>1100){j=5;pwm1=pwm1-25;if(pwm1<1101){g=5;}};
    if(x[0]<11&&g==5){g=0;pwm1=1600;};

    if(x[0]>10&&g<1){f=2;j=1;};

    if(x[0]>10&&g>0){f=2;};

    }

    else if(f==2){

        if (j==1){f=3;};
        if (j==2&&PINB.0==0&&PINB.1==1){f=3;};//h=1;k=0;
        if (j==3&&PINB.0==1&&PINB.1==0){f=3;};//h=0;k=1;
        if (j==4){f=3;};
        if (j==5){f=3;};
        }

    else if(f==3){

        if (j==1){tegak(0);};
        if (j==2){kanan(0);if(x[0]>10&&x[0]<78){tegak(0);}};
        if (j==3){kiri(0);if(x[0]>10&&x[0]<78){tegak(0);}};
        if (j==4){tegak(0);};
        if (j==5){tegak(0);};

        if (y[0]>1&&y[0]<61){c=1;d=0;};
        if (y[0]>60&&y[0]<82){c=0;d=0;};
        if (y[0]<1){c=0;d=0;};
        if (y[0]>81){c=0;d=1;};
        if (c==1&&d==0){pwm1=pwm1+25;};
    }

```

```

if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (c==0&&d==0&&x[0]>10){tegak();delay_ms(1000);f=4;};
if (j==1||j==4||j==5){if(x[0]<11){f=1;};};

}

else if(f==4){

if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (x[0]>1&&x[0]<24){a=1;b=0;};
if (x[0]>23&&x[0]<65){a=0;b=0;};
if (x[0]>64){a=0;b=1;};
if (a==1&&b==0){kiri();};
if (a==0&&b==1){kanan();};
if (a==0&&b==0){tegak();delay_ms(1250);f=5;};

j=0;
}

else if(f==5){

if (PINB.0==1&&PINB.1==1)
{
if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;f=6;};
if (y[0]<1){c=0;d=0;f=6;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};
}

}

else if(f==6){

if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};
}
}

```

```

if (x[0]>1&&x[0]<14){a=1;b=0;j=0;};
if (x[0]>13&&x[0]<75){a=0;b=0;};
if (x[0]>74){a=0;b=1;j=0;};
if (a==1&&b==0){kiri();};
if (a==0&&b==1){kanan();};
if (a==0&&b==0){j=1;};

if (j==1&&pwm1>1175){maju();};
if (j==1&&pwm1<1176){tegak();delay_ms(1250);f=7;};

}
////////////////////////////////////
else if(f==7){
if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (x[0]>1&&x[0]<34){a=1;b=0;};
if (x[0]>33&&x[0]<55){a=0;b=0;};
if (x[0]>54){a=0;b=1;};
if (a==1&&b==0){kiri();};
if (a==0&&b==1){kanan();};
if (a==0&&b==0){tegak();delay_ms(500);f=8;};
}
////////////////////////////////////
else if(f==8){ if
(PINB.0==1&&PINB.1==1){tegak();delay_ms(1000);pwm1=1600;e=3; g=0;j=0;f=9;};}
////////////////////////////////////

else if(f==9){

if(x[1]<11&&g==0){j=1;tegak();g=1;};
if(x[1]<11&&g==1){j=2;lihat_kanan();if(PINB.0==0&&PINB.1==1){g=2;}};
if(x[1]<11&&g==2){j=3;lihat_kiri();if(PINB.0==1&&PINB.1==0){g=3;}};

if(x[1]<11&&g==3&&pwm1<1800){j=4;tegak();pwm1=pwm1+25;if(pwm1>1799){g=4
;};};
if(x[1]<11&&g==4&&pwm1>1100){j=5;pwm1=pwm1-25;if(pwm1<1101){g=5;}};
if(x[1]<11&&g==5){g=0;pwm1=1600;};

if(x[1]>10&&g<1){j=1;tegak();f=10;};

if(x[1]>10&&g>0){f=10;};

}

else if(f==10){

if (j==1){f=11;};

```

```

if (j==2&&PINB.0==0&&PINB.1==1){f=11;};
if (j==3&&PINB.0==1&&PINB.1==0){f=11;};
if (j==4){f=11;};
if (j==5){f=11;};
}

else if(f==11){

if (j==1){tegak();};
if (j==2){kanan();if(x[1]>10&&x[1]<78){tegak();};};
if (j==3){kiri();if(x[1]>10&&x[1]<78){tegak();};};
if (j==4){tegak();};
if (j==5){tegak();};

if (y[1]>1&&y[1]<61){c=1;d=0;};
if (y[1]>60&&y[1]<82){c=0;d=0;};
if (y[1]<1){c=0;d=0;};
if (y[1]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (c==0&&d==0&&x[1]>10){tegak();delay_ms(1250);f=12;};
if (j==1||j==4||j==5){if(x[1]<11){f=9;};};

}

else if(f==12){

if (y[1]>1&&y[1]<61){c=1;d=0;};
if (y[1]>60&&y[1]<82){c=0;d=0;};
if (y[1]<1){c=0;d=0;};
if (y[1]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (x[1]>1&&x[1]<34){a=1;b=0;};
if (x[1]>33&&x[1]<55){a=0;b=0;};
if (x[1]>54){a=0;b=1;};
if (a==1&&b==0){kiri();};
if (a==0&&b==1){kanan();};
if (a==0&&b==0){tegak();};

if (a==0&&b==0&&c==0&&d==0){tegak();delay_ms(500);f=13;};

}

////////////////////////////////////
else if(f==13){ if
(PINB.0==1&&PINB.1==1){pwm1=1200;tegak();delay_ms(1000);e=2;g=0;j=0;k=1;f=1
4;};}
////////////////////////////////////

```

```

else if(f==14){

if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;f=15;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

}
////////////////////////////////////
else if(f==15){

if (y[0]>1&&y[0]<61){c=1;d=0;};
if (y[0]>60&&y[0]<82){c=0;d=0;};
if (y[0]<1){c=0;d=0;};
if (y[0]>81){c=0;d=1;};
if (c==1&&d==0){pwm1=pwm1+25;};
if (c==0&&d==1){pwm1=pwm1-25;};
if (c==0&&d==0){pwm1=pwm1+0;};

if (x[0]>1&&x[0]<44){a=1;b=0;j=1;};
if (x[0]>43&&x[0]<45){a=0;b=0;};
if (x[0]>44){a=0;b=1;j=1;};
if (a==1&&b==0){g=1;};
if (a==0&&b==1){g=2;};
if (a==0&&b==0){j=1;g=3;};

if (j==1&&pwm1>1160){maju();};
if (j==1&&pwm1<1161){tegak();delay_ms(1250);f=16;};

}

else if(f==16){

if (g==1){tendang_kiri();k=0;};
if (g==2){tendang_kanan();k=0;};
if (g==3){
if
(k==1){tendang_kanan();if(PINB.1==0&&PINB.2==1){tegak();delay_ms(1000);k=2;};}
;
if (k==2){tendang_kiri();k=0;};
};

if (k==0&&PINB.1==0&&PINB.2==1){f=17;};
if (k==0&&PINB.1==1&&PINB.2==0){f=17;};
}
////////////////////////////////////
else if(f==17){pwm1=1600;tegak();delay_ms(1250);g=0;j=0;k=0;f=1;}
};
}

```