



TUGAS AKHIR - TE 141599

***AUTONOMOUS DOCKING SYSTEM UNTUK MOBILE
ROBOT BERBASIS CITRA PADA STASIUN PENGISIAN
DAYA NIRKABEL***

Ariestya Putra Susanto
NRP 071114 4000115

Dosen Pembimbing
Dr. Muhammad Rivai, S.T., M.T.
Ir. Tasripan, M.T.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - TE 141599

***AUTONOMOUS DOCKING SYSTEM UNTUK MOBILE ROBOT
BERBASIS CITRA PADA STASIUN PENGISIAN DAYA
NIRKABEL***

Ariestya Putra Susanto
NRP 071114 4000115

Dosen Pembimbing
Dr. Muhammad Rivai, S.T., M.T.
Ir. Tasripan, M.T.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018

...



FINAL PROJECT - TE 141599

**AUTONOMOUS DOCKING SYSTEM FOR IMAGE BASED
MOBILE ROBOT ON WIRELESS CHARGING STATION**

Ariestya Putra Susanto
NRP 071114 4000115

Supervisor
Dr. Muhammad Rivai, S.T., M.T.
Ir. Tasripan, M.T.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "*Autonomous Docking System* untuk *Mobile Robot* Berbasis Citra pada Stasiun Pengisian Daya Nirkabel" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2018



Ariestya Putra Susanto
NRP. 07111440000115

**AUTONOMOUS DOCKING SYSTEM UNTUK MOBILE ROBOT
BERBASIS CITRA PADA STASIUN PENGISIAN DAYA
NIRKABEL**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Elektronika
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Dosen Pembimbing I



Dr. Muhammad Rivai, S.T., M.T.

NIP: 196904261994031003

Dosen Pembimbing II



Ir. Tasripan, M.T.

NIP: 196204181990031004



AUTONOMOUS DOCKING SYSTEM UNTUK MOBILE ROBOT BERBASIS CITRA PADA STASIUN PENGISIAN DAYA NIRKABEL

Nama : Ariestya Putra Susanto
Pembimbing 1 : Dr. Muhammad Rivai, ST., MT.
Pembimbing 2 : Ir. Tasripan, MT.

ABSTRAK

Teknologi pada robot mulai dikembangkan untuk dapat bekerja di luar ruangan selama 24 jam penuh. Permasalahan yang muncul adalah robot ini tidak memiliki daya baterai yang cukup agar dapat bekerja selama 24 jam penuh. Sebagai solusinya, maka dibangun stasiun pengisian daya di beberapa titik untuk mengisi daya pada baterai robot. Teknologi yang digunakan untuk mencari keberadaan stasiun pengisian daya adalah dengan menggunakan GPS. Akan tetapi GPS tidak selalu memberikan titik akurat dari keberadaan stasiun pengisian daya. Pada penelitian ini metode yang ditawarkan adalah dengan menggunakan kamera untuk mendeteksi dan menuju stasiun pengisian daya dengan asumsi robot telah menggunakan GPS untuk menemukan koordinatnya. Kamera akan mendeteksi objek dalam bentuk dua dimensi, kemudian citra yang ditangkap akan dikonversi ke bentuk *channel* HSV untuk mengetahui warna dari objek yang dijadikan target. Setelah itu, *contour* dari objek akan dicari untuk mendapatkan luas dan titik tengah objek. Dua parameter tersebut akan digunakan untuk menjalankan robot dengan motor dc yang akan dikontrol dengan kontrol proporsional untuk mendapatkan hasil yang lebih presisi. Kontrol proporsional menggunakan luas dan titik tengah sebagai *set point*. Hasil yang didapat dari uji coba pada *mobile robot* adalah bahwa titik tengah objek lebih efektif untuk dijadikan *set point* dengan tingkat keberhasilan yang didapat pada pengujian akhir sebesar 75%.

Kata kunci: *Autodocking*, citra, deteksi objek, kamera, *mobile robot*.

.....*Halaman ini sengaja dikosongkan*.....

AUTONOMOUS DOCKING SYSTEM FOR IMAGE BASED MOBILE ROBOT ON WIRELESS CHARGING STATION

Name : Ariestya Putra Susanto
1st Advisor : Dr. Muhammad Rivai, ST., MT.
2nd Advisor : Ir. Tasripan, MT.

ABSTRACT

Technology on robot is starting to get some improvement to be able to work for 24 hours a day. The problem is that the robot doesn't have enough battery capacity to work for 24 hours. The solution for this problem is to build some charging station on many location so that the robot can recharge its battery. The technology that is being used to find the location of the charging station is GPS. But, GPS weak point is the accuracy to decide the exact location of the charging station. For this problem, we offer the solution to use the camera to detect the charging station by capturing its image and tracking it. Camera will detect the object in the form of two dimensional image. After that, the image is converted into HSV image to find the color of the object. After the object color was found, the image is being processed to find the contour of the object. From the contour we can calculate the area and central point of the object. We can use both area and central point parameters to command the robot to approach the charging station by using dc motors and proportional control to get more precise result. Proportional control uses area and center point of the object as a set point. The result from the experiment on the mobile robot is that the central point of the object is more effective to be used as set point for mobile robot with the success rate about 75% from the result of the final experiment.

Key Word: Autodocking, camera, image, mobile robot, object detection.

.....*Halaman ini sengaja dikosongkan*.....

KATA PENGANTAR

Puji syukur atas rahmat yang telah diberikan oleh Allah SWT. Karena berkat rahmat dan hidayah-Nya penulis dapat menyelesaikan Tugas Akhir ini. Selama proses penelitian Tugas Akhir ini, penulis mendapatkan bantuan dari berbagai pihak baik dukungan secara moril maupun materiil. Oleh karena itu penulis ingin mengucapkan Terima kasih yang sebesar-besarnya kepada berbagai pihak yang mendukung dan membantu dalam tugas akhir ini, diantaranya :

1. Dr. Muhammad Rivai, ST., MT., dan Ir. Tasripan, MT. selaku dosen pembimbing, atas bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian tugas akhir ini.
2. Dr. Ir. Hendra Kusuma, M.Eng.Sc., Ir. Totok Mujiono, M.IKom.,Dr.Eng., Dr. Astria Nur Irfansyah, ST., M.Eng., Muhammad Attamimi, B.Eng, M.Eng, PhD. selaku dosen penguji, yang memberikan masukan dan pengarahan yang berguna bagi penulis dalam menyelesaikan buku tugas akhir ini dengan baik.
3. Kedua orang tua saya yang selalu memberi dukungan yang tidak ada hentinya dalam segala hal agar saya selalu mendapat semangat untuk mengerjakan tugas akhir ini.
4. Teman – teman Laboratorium Elektronika Dasar B202 dan Laboratorium Elektronika Industri B402 yang memberikan bantuan dan saran dalam pengerjaan tugas akhir ini.

Penulis sadar bahwa Tugas Akhir ini masih belum sempurna dan masih banyak hal yang perlu diperbaiki. Saran, kritik dan masukan baik dari semua pihak sangat membantu penulis terutama untuk berbagai kemungkinan pengembangan lebih lanjut.

Surabaya, Juli 2018

Ariestya Putra Susanto
NRP. 07111440000115

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	iii
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	1
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	2
1.5 Metodologi Penelitian.....	2
1.6 Sistematika Penulisan.....	3
1.7 Relevansi.....	4
BAB II TEORI PENUNJANG.....	5
2.1 <i>OpenCV</i>	5
2.2 <i>Image Processing</i>	5
2.2.1 <i>Color Spaces</i>	5
2.2.2 <i>Thresholding</i>	6
2.2.3 <i>Smoothing</i>	6
2.2.4 Morfologi Citra.....	7
2.2.5 Transformasi Citra.....	8
2.2.6 <i>Contour</i>	9
2.3 Motor DC.....	9
2.4 Driver Motor (L298).....	10
2.5 Raspberry Pi.....	12
2.6 Kamera Web.....	12
2.7 Pengendali Proposional Integral Derivatif (PID).....	13
BAB III PERANCANGAN SISTEM.....	15
3.1 Diagram Blok Sistem.....	15
3.2 Perancangan Perangkat Keras.....	17
3.2.1 <i>Buck Converter</i>	17
3.2.2 <i>Webcam</i>	17
3.2.3 Raspberry Pi.....	19
3.2.4 <i>Driver Motor L298</i>	19
3.3 Perancangan Mekanik <i>Mobile Robot</i>	20
3.3.1 Desain Badan <i>Mobile Robot</i>	20
3.4 Perangkat lunak.....	20

3.4.1 Perangkat Lunak Deteksi Objek	21
3.4.2 Perangkat Lunak Kontrol Kecepatan Motor	23
BAB IV PENGUJIAN DAN ANALISIS SISTEM	25
4.1 Realisasi Desain <i>Mobile Robot</i>	25
4.2 Pengujian <i>Software</i>	26
4.2.1 Pengujian Deteksi Objek	26
4.2.2 Pengujian Target pada Berbagai Macam Kondisi Pencahayaan	30
4.2.3 Pengujian Berdasarkan Letak Sumber Cahaya	32
4.2.4 Pengujian Jarak terhadap Target dengan Regresi	37
4.2.5 Pengujian <i>Mobile Robot</i> dengan <i>Set Point</i> Luas Objek	38
4.2.6 Pengujian <i>Mobile Robot</i> dengan <i>Set Point</i> Titik Tengah Objek	40
4.2.7 Pengujian dengan Posisi Awal yang Berbeda - beda	41
4.2.8 Pengujian Akhir <i>Mobile Robot</i> Terhadap Stasiun Pengisian Daya.....	43
BAB V PENUTUP	45
5.1 Kesimpulan	45
5.2 Saran	45
DAFTAR PUSTAKA	47
LAMPIRAN	49
BIODATA PENULIS	53

DAFTAR GAMBAR

Gambar 2. 1 Contoh hasil filter gauss	6
Gambar 2. 2 Morfologi dilasi.....	7
Gambar 2. 3 Morfologi erosi.....	8
Gambar 2. 4 Hasil canny edge detection.....	9
Gambar 2. 5 Diagram skematik motor dc	10
Gambar 2. 6 Modul <i>motor driver</i> l298n.....	10
Gambar 2. 7 Rangkaian modul <i>motor driver</i> l298n	11
Gambar 2. 8 Raspberry pi 3	12
Gambar 2. 9 Blok diagram kontrol pid	14
Gambar 3. 1 Blok diagram sistem Mobile Robot.....	16
Gambar 3. 2 Buck Converter dengan Output USB	17
Gambar 3. 3 Webcam Logitech C270	18
Gambar 3. 4 Desain <i>mobile robot</i> pada <i>sketch up</i>	20
Gambar 3. 5 <i>Flowchart</i> Deteksi Objek	22
Gambar 3. 6 <i>Flowchart</i> Pengatur Kecepatan Motor	23
Gambar 4. 1 Hasil Realisasi <i>Mobile robot</i>	25
Gambar 4. 2 Tiga Persegi Panjang Berwarna Biru	27
Gambar 4. 3 Grafik Perbandingan Jarak Asli dengan Luas <i>Target I</i>	27
Gambar 4. 4 Grafik Perbandingan Jarak Asli dengan Luas <i>Target II</i> ..	28
Gambar 4. 5 Hasil <i>Target I</i> pada Jarak 100 cm.	29
Gambar 4. 6 Hasil <i>Target II</i> pada Jarak 100 cm.....	29
Gambar 4. 7 Grafik Perbandingan Luas dan Intensitas Cahaya pada Jarak 50 cm.....	31
Gambar 4. 8 Grafik Perbandingan Luas dan Intensitas Cahaya pada Jarak 100 cm.....	32
Gambar 4. 9 Sumber Cahaya pada Posisi 1	33
Gambar 4.10 Sumber Cahaya pada Posisi 2	33
Gambar 4.11 Sumber Cahaya pada Posisi 3	34
Gambar 4.12 Grafik Perbandingan Luas.....	36
Gambar 4.13 Grafik Perbandingan Jarak dan Luas.....	37
Gambar 4.14 Posisi Awal 1	41
Gambar 4.15 Posisi Awal 2.....	41
Gambar 4.16 Posisi Awal 3.....	42
Gambar 4.17 Posisi Awal 4.....	42

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR TABEL

Tabel 3.1	Penggunaan Pin Raspberry Pi	19
Tabel 3.2	Penggunaan Pin <i>Motor Driver</i>	19
Tabel 4.1	Hasil pengujian deteksi warna target I	26
Tabel 4.2	Hasil pengujian deteksi warna target II.....	28
Tabel 4.3	Hasil Pengujian Target pada Jarak 50 cm	30
Tabel 4.4	Hasil Pengujian Target pada Jarak 100 cm	31
Tabel 4.5	Hasil Pengujian Target pada Posisi 1	34
Tabel 4.6	Hasil Pengujian Target pada Posisi 2.....	35
Tabel 4.7	Hasil Pengujian Target pada Posisi 3	35
Tabel 4.8	Data Perbandingan Jarak asli dengan Jarak Hasil Regresi	38
Tabel 4.9	Hasil pengujian <i>mobile robot</i> dengan <i>set point</i> luas objek.....	39
Tabel 4.10	Hasil pengujian <i>mobile robot</i> dengan <i>set point</i> titik tengah objek.....	40
Tabel 4.11	Hasil pengujian <i>mobile robot</i> dengan posisi awal berbeda.....	42
Tabel 4.12	Hasil pengujian akhir pada <i>mobile robot</i>	44

.....*Halaman ini sengaja dikosongkan*.....

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada area yang luas seperti area industri migas memiliki banyak pipa yang kebocorannya sulit dideteksi karena cakupan wilayah yang luas dan dibutuhkan sensor untuk mendeteksi adanya kebocoran gas. Untuk itu diperlukan sebuah alat yang mampu bergerak berupa robot untuk melakukan deteksi terhadap kebocoran gas. Dengan berkembangnya teknologi saat ini maka berpengaruh terhadap perkembangan sistem yang ada pada robot. Saat ini teknologi pada robot mulai dikembangkan untuk melakukan pekerjaan di luar ruangan. Permasalahan yang muncul adalah baterai pada robot sangat terbatas untuk melakukan pekerjaan di luar ruangan sehingga diperlukan cara untuk menjaga agar robot tersebut dapat bekerja selama 24 jam tanpa harus dipengaruhi oleh permasalahan yang timbul oleh terbatasnya daya pada baterai robot. Alternatif yang dapat digunakan adalah dengan membangun stasiun pengisian daya pada baterai mobile robot. Dengan adanya stasiun pengisian daya pada baterai robot, diperlukan suatu sistem pada robot agar dapat melakukan docking pada tempat pengisian daya dengan posisi yang tepat agar daya yang terisi pada robot menjadi optimal. Sistem yang akan dibuat untuk melakukan autodocking terdiri dari sensor – sensor yang nantinya diharapkan dapat menemukan lokasi dari stasiun pengisian daya dan membuat robot bergerak menuju stasiun pengisian daya. Saat ini telah dibuat sistem pada mobile robot dengan menggunakan sensor ultrasonik untuk mendeteksi jarak terhadap target. Perangkat yang dapat digunakan dapat berupa kamera yang akan menangkap citra dan dapat menentukan target berupa stasiun pengisian daya. Karena saat *mobile robot* hanya mengandalkan perangkat pelacak posisi seperti GPS hasilnya belum tentu akurat sehingga dibutuhkan perangkat tambahan untuk memastikan bahwa *mobile robot* dapat bergerak mencapai target dan melakukan docking pada stasiun pengisian daya.

1.2 Perumusan Masalah

Permasalahan pada tugas akhir ini adalah sebagai berikut :

1. Bagaimana sistem mampu melakukan deteksi pada stasiun pengisian daya.

2. Cara untuk menggerakkan *mobile robot* menuju stasiun pengisian daya untuk melakukan *docking*.
3. Mikrokontroler yang digunakan untuk menjalankan sistem *docking* otomatis.

1.3 Tujuan Penelitian

Penelitian ini memiliki beberapa tujuan sebagai berikut :

1. Memastikan sistem pada *mobile robot* agar dapat mendeteksi objek berupa stasiun pengisian daya.
2. Menggunakan motor dc dengan mengimplementasikan control PID sebagai metode untuk mengontrol kecepatan motor.
3. Menggunakan mikrokontroler Raspberry Pi.

1.4 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. *Mobile robot* hanya berhenti pada jarak yang sudah ditentukan. Jarak tersebut disesuaikan dengan jarak optimal untuk melakukan pengisian daya nirkabel
2. *Mobile robot* tidak melakukan pengisian daya setelah mencapai stasiun pengisian daya.

1.5 Metodologi Penelitian

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi literatur

Tahap ini meliputi pengumpulan dasar teori yang dapat dijadikan acuan tugas akhir. Dasar teori akan diambil dari buku – buku, jurnal – jurnal yang telah dipublikasi dan proceeding artikel dari internet mengenai metode untuk mengenali dan melacak objek menggunakan kamera.

2. Perancangan Platform Robot

Pada tahap ini akan dilakukan perncangan hardware yang akan digunakan pada auto-docking system berupa kamera sebagai sensor untuk menentukan jarak dengan stasiun pengisian daya dan menghubungkan ke motor pada mobile robot agar bergerak menuju stasiun pengisian daya. Perancangan platform robot meliputi perancangan desain kerangka robot dan pemasangan kamera pada robot.

3. Perancangan *Hardware* Elektronik

Pada tahap ini dilakukan perancangan sistem elektronika berupa kamera, raspberry pi, dan driver motor yang digabung menjadi sebuah sistem yang terintegrasi.

4. Perancangan *Software* Identifikasi Objek

Pada tahap ini akan dilakukan perancangan *software* berupa program untuk melakukan identifikasi dan *tracking* objek berupa stasiun pengisian daya menggunakan kamera agar dapat melakukan *docking* otomatis.

5. Perancangan Sistem

Setelah dilakukan studi literatur dari berbagai referensi pada pengerjaan tugas akhir ini, langkah berikutnya yang akan diambil adalah melakukan perancangan sistem berupa implementasi dari *hardware* dan *software* yang diperlukan dalam membuat *mobile robot* agar dapat melakukan tujuannya yaitu *docking* otomatis terhadap stasiun pengisian daya nirkabel.

6. Pengujian Sistem

Proses pengujian sistem dilakukan dengan menghubungkan semua perangkat keras berupa perancangan *mobile robot* dan perangkat lunak berupa proses pengolahan citra yang kemudian digabungkan ke dalam satu sistem yang dapat memungkinkan *mobile robot* untuk melakukan *autodocking*. Pada tahap ini dilakukan pengambilan data berupa pengujian sistem *autodocking* dengan menggunakan kamera sebagai perangkat untuk mendeteksi stasiun pengisian daya. Hasil *tracking* dari kamera kemudian akan menentukan pergerakan dari *mobile robot* dengan menggunakan motor dc sebagai aktuatornya. Kecepatan dari motor dc akan diatur menggunakan sistem control PID.

7. Penulisan Laporan Tugas Akhir

Setelah dilakukan pengujian dan pengambilan data, akan dilakukan penulisan Laporan Tugas Akhir yang bersifat final.

1.6 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari lima bab dengan sistematika penulisan sebagai berikut:

- Bab 1 : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.

- Bab 2 : Dasar Teori

Bab ini menjelaskan tentang dasar-dasar teori yang dibutuhkan dalam pengerjaan tugas akhir ini, yang meliputi teori dasar pengolahan citra, *PID controller*, dan Raspberry Pi.

- Bab 3: Perancangan Sistem

Bab ini menjelaskan tentang perancangan keseluruhan sistem yang terdiri dari perangkat keras (*hardware*) dan mekanik, serta perangkat lunak (*software*).

- Bab 4 : Pengujian dan Pembahasan Sistem

Bab ini menjelaskan tentang hasil yang didapat dari pengujian tiap Blok sistem secara keseluruhan

- Bab 5 : Penutup

Bab ini menjelaskan tentang kesimpulan meliputi kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

1.7 Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Dapat menggerakkan *mobile robot* menuju stasiun pengisian daya pada posisi yang tepat dengan menggunakan kamera.
2. *Mobile robot* dapat mencapai target berupa stasiun pengisian daya dengan jarak dan posisi yang tepat.

BAB II

TEORI PENUNJANG

2.1 OpenCV

OpenCV adalah sebuah *Application Programming Interface (API) library* yang digunakan pada proses pengolahan citra yang berbasis pada computer. *Computer vision* merupakan suatu ilmu yang mempelajari tentang pengolahan citra yang bertujuan agar computer dapat memiliki sistem penglihatan seperti manusia.

OpenCV berfungsi sebagai *open source library* yang digunakan pada *computer vision* untuk mengimplementasikan berbagai hal seperti deteksi wajah, deteksi objek, pengenalan wajah, dan lain – lain menggunakan berbagai macam bahasa pemrograman diantaranya adalah C/C++, Python, Java, dan Matlab. *OpenCV* dirancang untuk menjalankan aplikasi yang bersifat *real-time* terhadap fungsi – fungsi akuisisi citra dan video.[2]

2.2 Image Processing

Pengolahan citra adalah sebuah operasi yang dilakukan untuk mengolah data berupa data *digital* dalam bentuk cuplikan gambar maupun hasil rekaman video yang diambil dari perangkat berupa kamera. Hasil dari pengolahan citra nantinya akan disegmentasi dan dianalisis untuk dilakukan sebuah pengenalan pola atau *pattern*.

2.2.1 Color Spaces

Color spaces adalah tipe – tipe berbeda dari mode warna yang digunakan pada pemrosesan gambar, sinyal dan, sistem untuk berbagai tujuan. *Color space* diantaranya adalah RGB, HSV, dan CMYK. RGB adalah salah satu jenis *color space* yang memiliki tiga buah kanal yang digunakan untuk nilai warna dimana masing – masing kanal memiliki satu nilai warna. Tiga nilai warna tersebut adalah *red*, *green*, dan *blue*. Tiap – tiap kanal memiliki nilai sebesar 8 bit mulai dari 0 hingga 255. Jika semua kanal pada *color space* RGB bernilai nol maka *output* dari gambar adalah warna hitam, sedangkan bila semua kanal bernilai 255 maka akan menghasilkan warna putih pada gambar *output*.

HSV adalah salah satu jenis *color space* yang memiliki tiga buah kanal yaitu *hue*, *saturation*, dan *value*. *Hue* adalah kanal yang berfungsi sebagai tempat untuk membagi nilai dari warna ke dalam bentuk dimensi anguler mulai dari 0° hingga 360°. *Saturation* adalah kanal yang

berfungsi untuk menentukan seberapa kuat warna dari nilai warna yang ditentukan dari kanal *hue*. *Value* adalah kanal yang menjadi indikator dari intensitas gelap terang dari sebuah warna.

2.2.2 Thresholding

Thresholding adalah operasi non-linear yang melakukan konversi gambar *grayscale* menjadi bentuk gambar biner dimana dilakukan pemisahan pixel menjadi dua level yaitu pixel yang berada di bawah level *threshold* dan di atas level *threshold*. Pada *threshold* bila pixel input di bawah nilai *threshold* maka nilai output menjadi 0 atau hitam, bila di atas nilai *threshold* maka *output pixel* menjadi 255 atau putih.

2.2.3 Smoothing

Smoothing atau disebut juga sebagai *Blurring* adalah salah satu operasi dalam pengolahan citra yang berfungsi untuk menghilangkan *noise* yang ada pada citra yang ditampilkan. Filter yang digunakan pada tugas akhir ini adalah *Gaussian Filter*. *Gaussian Filter* merupakan jenis filter yang cara kerjanya adalah dengan melakukan konvolusi pada tiap *point* pada *input array* menggunakan *Gaussian Kernel* kemudian dijumlahkan untuk menghasilkan *output array*. *Gaussian filter* dapat digunakan pada citra dengan satu kanal maupun tiga kanal.[6]

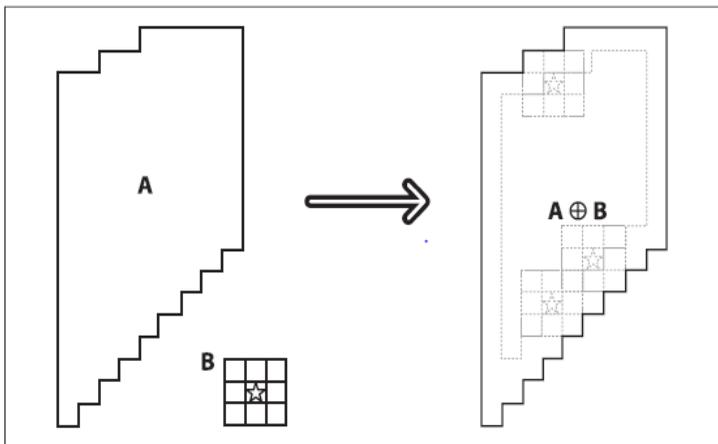


Gambar2. 1 Contoh Hasil Filter Gauss [6]

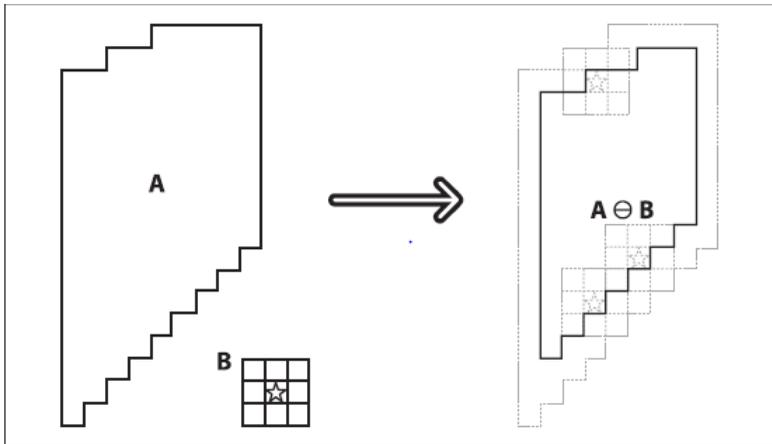
2.2.4 Morfologi Citra

Transformasi morfologi paling sederhana pada pengolahan citra adalah dilasi dan erosi. Dilasi adalah sebuah konvolusi dari sebuah wilayah di suatu gambar yang dimisalkan dengan nilai A dan *kernel* yang dimisalkan dengan nilai B, dimana kernel dapat memiliki berbagai macam bentuk dan ukuran yang memiliki sebuah *anchor point* yang biasanya terletak pada pusat dari *kernel*. Dilasi menyebabkan nilai A yang di *scanning* oleh kernel B dan posisi dari *pixel*-nya berada di bawah *anchor point* berubah nilainya menjadi maksimum. Hal ini menyebabkan wilayah terang dari citra semakin melebar. Oleh karena itu dilasi berfungsi sebagai *local maximum operator*. [6]

Erosi adalah operasi yang sifatnya berlawanan dengan dilasi. Pada erosi, gambar A yang di *scanning* oleh *kernel* B dan posisi pixel gambar A yang berada di bawah *anchor point* nilainya berubah menjadi minimum. Sehingga erosi merupakan *local minimum operator*. Operasi yang sering dijalankan pada morfologi citra adalah *opening* dan *closing*. Pada *opening*, operasi yang dijalankan terhadap gambar adalah erosi kemudian dilasi, sedangkan pada *closing* operasi yang dijalankan adalah dilasi kemudian erosi. [6]



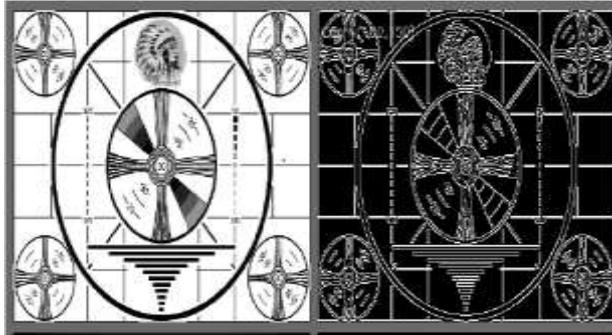
Gambar2. 2 Morfologi Dilasi [6]



Gambar2. 3 Morfologi Erosi[6]

2.2.5 Transformasi Citra

Transformasi citra adalah sebuah metode untuk mengubah sebuah citra menjadi sebuah representasi alternatif dari data secara keseluruhan. Dalam tugas akhir ini jenis transformasi yang digunakan adalah *Canny* dan transformasi *Hough*. *Canny* adalah sebuah metode untuk mencari ujung dari sebuah citra. Hasil dari citra yang diolah dengan algoritma *canny* adalah dengan menyusun *individual pixel* yang menjadi kandidat untuk dijadikan ujung gambar menjadi *contour*. *Contour* ini nantinya akan terbentuk dengan menggunakan *upper* dan *lower threshold* untuk menentukan *pixel* mana yang akan dijadikan ujung dari sebuah gambar. Transformasi *hough* secara umum adalah metode untuk mencari garis, lingkaran, dan bentuk sederhana lain dari sebuah citra. Bentuk dasar dari *hough transform* adalah transformasi garis yang tujuannya adalah untuk mencari garis lurus pada sebuah citra yang bersifat biner.[6]



Gambar2. 4 Contoh hasil *Canny Edge Detection*[6]

2.2.6 Contour

Contour adalah susunan dari titik yang merepresentasikan kurva atau lengkungan dari sebuah citra. Pada *OpenCV*, *contour* direpresentasikan oleh sekuen dimana setiap *entry* dari sekuen menuliskan informasi mengenai lokasi dari tiap titik pada kurva. *Contour* dari sebuah citra dapat ditemukan dengan menghitung hasil dari citra yang telah diproses melalui *canny edge detection* atau melalui *threshold*. Setelah ditemukan, *contour* akan digambar lagi sebagai *output* pada layar.

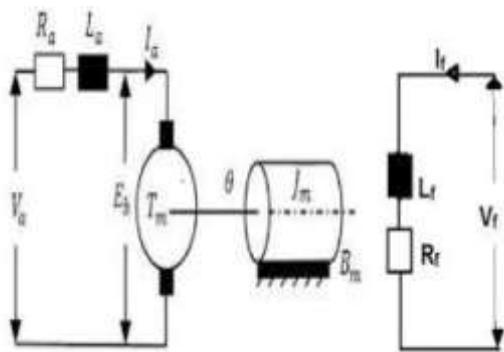
Pada sebuah citra berupa gambar atau video, dapat ditemukan lebih dari satu *contour*. Untuk menemukan *contour* yang cocok maka harus dilakukan komparasi terhadap dua *contour* yang berbeda. Untuk melakukan komparasi tersebut, terdapat sebuah metode bernama *contour moments*. *Moments* adalah sebuah karakteristik mencolok dari *contour* yang dihitung dengan cara mengintegrasikan atau menjumlahkan semua pixel yang ada di dalam *contour*. [6]

2.3 Motor DC

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Bagian stator terbagi menjadi sikat, rangka dan kumparan medan. Sedangkan rotor terdiri dari komutator dan kumparan jangkar.

Cara kerja motor dc adalah saat motor diberi sumber arus dc, arus akan melewati bagian sikat dari motor. Sikat motor akan mengarahkan

arus menuju komutator, dimana komutator akan mengubah arah dari arus listrik di kumparan. Arus listrik akan melewati konduktor sehingga konduktor akan menghasilkan medan magnet yang arahnya ditentukan oleh arah datangnya arus. Stator memiliki dua buah kutub magnet permanen yaitu kutub utara dan selatan. Kedua kutub magnet inilah yang nantinya akan memutar kumparan yang arah putarannya ditentukan oleh aturan *Flamming* menggunakan tangan kiri.



Gambar2. 5 Diagram Skematik Motor DC [3]

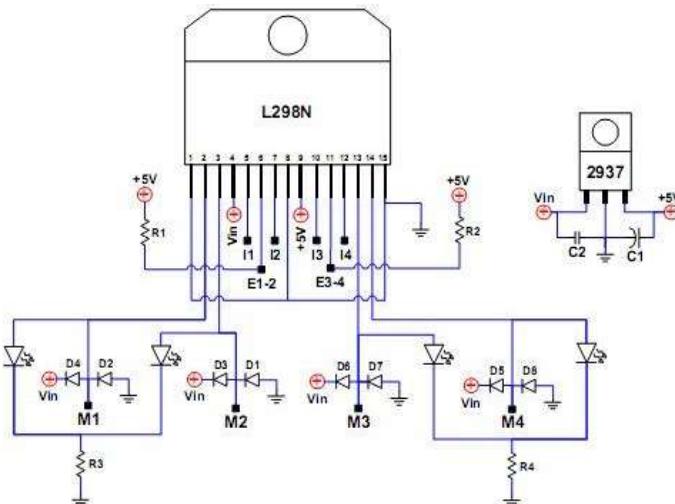
2.4 Driver Motor (L298)



Gambar2. 6 Modul *Motor Driver* L298N

Gambar 2.6 merupakan contoh modul *driver* motor yang digunakan pada tugas akhir ini. *Driver* motor merupakan komponen penting yang berfungsi untuk mengatur arah (*Counter Clock Wise* atau *Counter Wise*) dan kecepatan putar pada motor. Pengaturan arah putar dan kecepatan motor yang dihasilkan berasal dari perintah program yang ada pada Raspberry Pi.

Pada Gambar 2.7 di bawah merupakan konfigurasi dari rangkaian modul *driver* motor tipe L298N secara umum. Prinsip kerja dari *motor driver* yaitu memberikan sumber tegangan 12 volt sesuai dengan tegangan pada motor yang digunakan, kemudian tegangan tersebut digunakan sebagai sumber tegangan motor dan regulator. Regulator berfungsi untuk menurunkan tegangan dari 12 volt menjadi 5 volt, yang berfungsi sebagai sumber tegangan pada IC yang digunakan pada *driver* motor L298N. Dalam menentukan arah putar motor, dapat menggunakan Raspberry Pi yang terhubung dengan pin IN1, IN2, IN3, dan IN4 pada motor driver, sedangkan untuk mengatur kecepatan putar pada motor, dapat menggunakan PWM (*Pulse Width Modulation*) yang terhubung dengan pin ENA dan ENB. [5]



Gambar2. 7 Rangkaian Modul *Motor Driver* L298N [5]

2.5 Raspberry Pi



Gambar2. 8 Raspberry Pi 3

A Raspberry Pi adalah sebuah papan tunggal (SBC) berukuran kartu kredit yang dihubungkan ke TV (via HDMI) dan keyboard. Sebagai IoT (Internet of Things), seperti layaknya sebuah desktop, PC kecil ini mampu digunakan untuk menjalankan spreadsheet, pengolahan kata dan permainan, terutama untuk memainkan video definisi tinggi. Desain Raspberry Pi didasarkan seputar SoC (System-on-a-chip) Broadcom BCM2835, yang telah menanamkan prosesor ARM1176JZF-S dengan 700 MHz, VideoCore IV GPU, dan 256 Megabyte RAM (model B). Penyimpanan data didesain tidak untuk menggunakan hard disk atau solid-state drive, melainkan mengandalkan kartu SD (SD memory card) untuk booting dan penyimpanan jangka panjang. Tugas utama Raspberry Pi menjalankan sistem operasi berbasis kernel Linux.

2.6 Kamera Web

Kamera web atau *Webcam* adalah perangkat penangkap citra berupa foto atau video yang bekerja secara *real-time* dan dihubungkan ke komputer melalui USB. Kamera web terdiri dari lensa standar dan papan sirkuit untuk dapat melakukan penangkapan sinyal gambar dan *casing* untuk sebagai badan kamera dan pelindung lensa kamera. Kamera web memiliki perangkat lunak untuk mengambil gambar dari kamera digital secara terus menerus atau dalam interval waktu tertentu. Salah satu fitur kamera web yang digunakan dalam pengerjaan tugas akhir ini adalah *custom coding*, yaitu mengimpor kode komputer untuk memberi perintah terhadap kamera.

Pada tugas akhir ini, kamera web yang digunakan adalah *Logitech C270*. Kamera ini memiliki resolusi untuk menangkap citra berupa video dengan nilai resolusi sebesar 1280 x 720 *pixels* dan foto sebesar 3 *megapixels*. [4]

2.7 Pengendali Proporsional Integral Derivatif (PID)

PID merupakan sebuah kontroler yang berfungsi mengatur kecepatan motor dc dari *mobile robot* agar dapat bergerak dengan baik menuju stasiun pengisian daya nirkabel. Pengendali PID terbagi menjadi tiga macam respon, yaitu :

- Respon Proporsional
Komponen proporsional hanya bergantung pada perbedaan antara set point dan variable proses. Perbedaan ini dihitung sebagai error. Proportional Gain (K_p) menentukan rasio dari respon output terhadap sinyal error. Meningkatkan nilai K_p akan meningkatkan kecepatan respon sistem control. Bila K_p terlalu besar maka variable proses mulai berosilasi. Bila osilasi semakin besar maka sistem menjadi tidak stabil.
- Respon Integral
Komponen integral menjumlah error terhadap waktu. Pertambahan error yang kecil pun akan menyebabkan komponen integral meningkat secara lambat. Respon integral akan meningkat terhadap waktu walaupun error bernilai nol sehingga efeknya yaitu menggerakkan *steady state error* menuju nol. *Steady state error* adalah selisih final antara variable proses dengan set point. Kejadian bernama *integral wind up* akan menghasilkan output ketika integral menghasilkan saturasi pada kontroler tanpa kontroler menggerakkan sinyal *error* menuju nol.
- Respon *Derivative*
Komponen *Derivative* menyebabkan *output* menjadi berkurang jika variabel proses bertambah dengan cepat. Respon *derivative* bersifat proporsional terhadap jangkauan perubahan dari variabel proses. Meningkatkan parameter waktu *derivative* akan menyebabkan sistem kontrol bereaksi lebih kuat pada perubahan *error* dan meningkatkan kecepatan respon sistem kontrol secara keseluruhan. Pada aplikasinya , kebanyakan sistem kontrol hanya menggunakan waktu *derivative* (T_d) yang kecil, karena respon *derivative* sangat sensitif terhadap *noise* pada sinyal proses. Jika sinyal feedback sensor memiliki *noise* tinggi atau *rate* dari *control loop* terlalu pelan, respon *derivative* dapat membuat sistem kontrol menjadi tidak stabil.

Dari gambar 2.9 fungsi transfer untuk pengendalian PID adalah :

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(t) + K_d \frac{de(t)}{dt} \quad (2.2)$$

Keterangan :

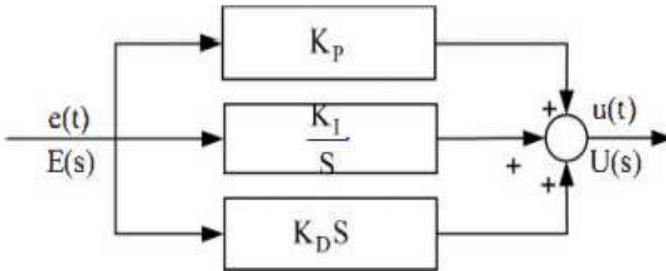
u = *Output*

e = sinyal *error*

K_p = Konstanta Proporsional

K_i = Konstanta Integral

K_d = Konstanta *Derivative*



Gambar2. 9 Blok Diagram Kontrol PID [1].

BAB III

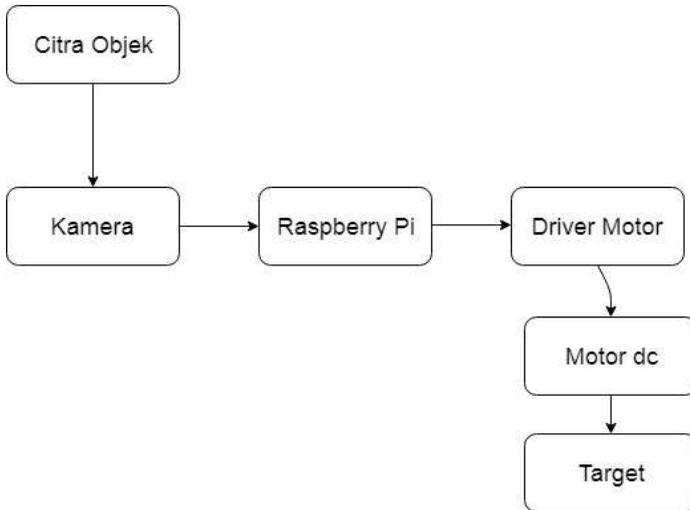
PERANCANGAN SISTEM

Pada bab ini dijelaskan mengenai perancangan sistem secara keseluruhan yang meliputi perancangan perangkat keras, perangkat lunak dan perancangan mekanik dari *mobile robot*. Perangkat mekanik merupakan badan dari robot yang dirangkai dengan bahan akrilik dan disusun menggunakan dua roda belakang yang digerakkan oleh motor dc dan satu roda depan yang berupa roda bebas. Sedangkan perangkat elektrik yang akan digunakan yaitu Raspberry Pi, *webcam*, dan rangkaian *buck converter*. Perangkat lunak terdiri dari program pengolahan citra yang didapat dari kamera untuk melakukan deteksi target berupa stasiun pengisian daya dan memberi perintah untuk mendekati target.

3.1 Diagram Blok Sistem

Diagram blok sistem menjelaskan rangkaian kerja secara keseluruhan dari *auto docking* yang akan dilakukan oleh *mobile robot* terhadap stasiun pengisian daya. Sistem kerja dari *mobile robot* yaitu dengan menggunakan kamera untuk melakukan deteksi terhadap target dan menggunakan motor dc agar robot dapat bergerak menuju target.

Gambar 3.1 menunjukkan blok diagram sistem secara keseluruhan pada *mobile robot* untuk melakukan *docking* otomatis terhadap stasiun pengisian daya nirkabel. Komponen yang akan dijadikan *set point* pada sistem ini adalah titik tengah dari target yang akan dituju. Kamera akan digunakan sebagai perangkat untuk menangkap citra dan diolah untuk mendapatkan posisi target. Pengendali PID akan memberi kontrol terhadap driver motor untuk mengatur sinyal PWM yang akan digunakan untuk mengatur kecepatan motor dc dalam melakukan pergerakan menuju target.



Gambar3. 1 Blok diagram sistem Mobile Robot

Perangkat keras pada sistem *mobile robot* terdiri dari :

- *Raspberry Pi*
Raspberry Pi digunakan untuk mengolah data berupa citra untuk menentukan pergerakan dari *mobile robot*.
- *Driver Motor* dan *Motor*
Driver motor digunakan untuk mengatur sinyal PWM menjadi besaran tegangan yang berfungsi sebagai sumber dari motor pada *mobile robot*.
- *Webcam*
Webcam adalah kamera yang digunakan untuk mengambil data masukan berupa citra yang kemudian akan diolah oleh *raspberry pi*.
- *Buck Converter*
Buck converter adalah rangkaian elektronika yang digunakan untuk menurunkan tegangan agar menjadi 5 volt untuk digunakan pada *raspberry pi*.



Gambar3. 2 Buck Converter dengan Output USB

Perangkat lunak untuk sistem *autodocking* adalah :

- Perangkat lunak
Perangkat lunak yang dirancang adalah untuk melakukan pengolahan citra agar dapat mendeteksi objek yang dijadikan target dan memberi perintah pada *mobile robot* untuk bergerak menuju target.

Berdasarkan gambar 3.1, ketika kamera telah menangkap citra dari objek maka citra tersebut akan diproses ke dalam *raspberry pi* untuk menentukan objek tersebut adalah target yang harus dituju oleh *mobile robot* atau bukan merupakan target. Apabila objek adalah target maka *raspberry pi* akan memberi perintah kepada *motor driver* untuk menggerakkan motor dc menuju target berupa stasiun pengisian daya nirkabel. Kecepatan dan arah motor dc diatur oleh pengendali PID.

3.2 Perancangan Perangkat Keras

Perangkat keras pada *mobile robot* meliputi komponen *buck converter*, *webcam*, *motor driver*, dan *Raspberrry Pi*.

3.2.1 Buck Converter

Buck converter dengan output USB *mini* seperti pada gambar 3.2 berfungsi mengubah tegangan dari sumber dari sumber berupa baterai 18650 sebesar 7,4 volt menjadi tegangan 5 volt yang dijadikan sebagai sumber tegangan pada komponen – komponen dari *mobile robot* diantaranya kamera berupa *webcam* dan mini PC *Raspberrry Pi 3*.

3.2.2 Webcam

Webcam merupakan sebuah kamera yang nantinya akan digunakan untuk menangkap citra dan hasilnya akan diolah untuk mendapatkan data berupa citra dari objek yang akan dituju oleh *mobile robot* sehingga

dapat dilakukan *docking* otomatis terhadap objek tersebut. Objek yang dituju adalah stasiun pengisian daya nirkabel.

Kamera tidak hanya melakukan deteksi objek tetapi juga melakukan proses perhitungan jarak terhadap objek tersebut. Proses perhitungan dilakukan dengan melakukan kalibrasi pada dua buah kamera yang hasilnya dapat digunakan sebagai variabel dalam menghitung jarak terhadap objek. Proses kalibrasi dilakukan dengan meletakkan dua kamera pada posisi sejajar dan menghadap ke arah yang sama. Objek yang akan digunakan untuk melakukan kalibrasi adalah gambar papan catur. Kamera yang digunakan adalah *webcam Logitech c270* seperti yang terlihat pada gambar 3.3

Raspberry Pi digunakan untuk mengolah data berupa citra yang ditangkap oleh kamera dan dilakukan proses deteksi objek. Setelah titik tengah dari objek didapatkan, maka raspberry memberi perintah kepada *motor driver* untuk menjalankan motor dc melalui dua buah *pinout* agar dapat menjalankan motor kanan dan kiri serta menggunakan PWM untuk mengubah kecepatan motor dc.



Gambar3. 3 Webcam Logitech C270 [4]

3.2.3 Raspberry Pi

Tabel 3.1 Penggunaan pin Raspberry Pi

No.	Pin Raspberry	Komponen	Keterangan
1	USB <i>port</i>	Kamera	Menangkap citra objek
2	GPIO23	IN1 <i>motor driver</i>	Menggerakkan dan mengatur kecepatan motor dc
3	GPIO26	IN2 <i>motor driver</i>	Menggerakkan dan mengatur kecepatan motor dc
4	<i>Mini USB port</i>	Kabel <i>mini USB</i>	Sumber tegangan 5 V
5	GND	-	Ground

3.2.4 Driver Motor L298

Tabel 3.2 Penggunaan pin *driver motor*

No.	Pin L298 <i>Motor Driver</i>	Keterangan
1	12 V	Sumber tegangan 12 V
2	5 V	Sumber tegangan 5 V
3	GND	Ground
4	IN1	Input dari raspberry
5	IN2	Input dari raspberry
6	IN3	Input dari raspberry
7	IN4	Input dari raspberry
8	OUT1	Output ke motor dc 1 (+)
9	OUT2	Output ke motor dc 1 (-)
10	OUT3	Output ke motor dc 2 (+)
11	OUT4	Output ke motor dc 2 (-)
12	ENA	Pin enable motor dc 1
13	ENB	Pin enable motor dc 2

Driver motor L298 merupakan komponen yang berfungsi sebagai penggerak dan pengatur kecepatan motor dc. Pin input berfungsi menerima data dari raspberry yang nantinya akan diolah untuk dikeluarkan melalui pin output yang akan menggerakkan motor dc. Pin *enable* berfungsi sebagai pengolah sinyal PWM untuk mengatur

kecepatan motor dc. Pin 12 V, 5 V , dan pin *ground* digunakan sebagai sumber tegangan untuk *motor driver* agar dapat menjalankan motor dc.

3.3 Perancangan Mekanik *Mobile Robot*

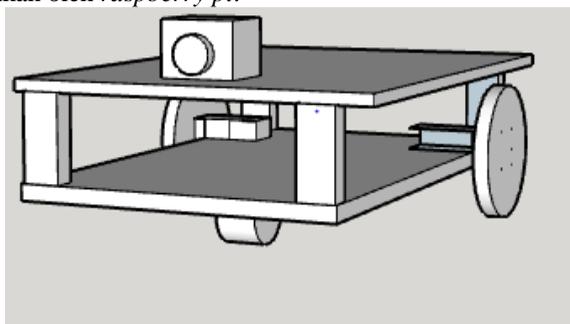
Perancangan mekanik pada *mobile robot* menggunakan rancangan yang didesain pada *sketch up* dengan memperhitungkan luas dan tinggi dari rancangan tersebut agar dapat menampung semua perangkat keras yang akan menjalankan sistem *docking* otomatis oleh *mobile robot*.

3.3.1 Desain Badan *Mobile Robot*

Desain dari robot yang akan digunakan berupa robot dengan menggunakan bentuk yang mirip dengan *line tracer* yaitu dengan menggunakan satu roda bebas di bagian depan dan dua roda di bagian belakang yang digerakkan oleh satu motor dc untuk setiap roda.

3.4 Perangkat lunak

Pada bab ini, perancangan perangkat lunak berupa proses pengolahan citra yang ditujukan untuk mendeteksi objek yang akan dijadikan target berupa stasiun pengisian daya nirkabel dan memberi perintah pada *mobile robot* untuk bergerak menuju ke target. Proses ini akan dijalankan oleh *raspberry pi*.

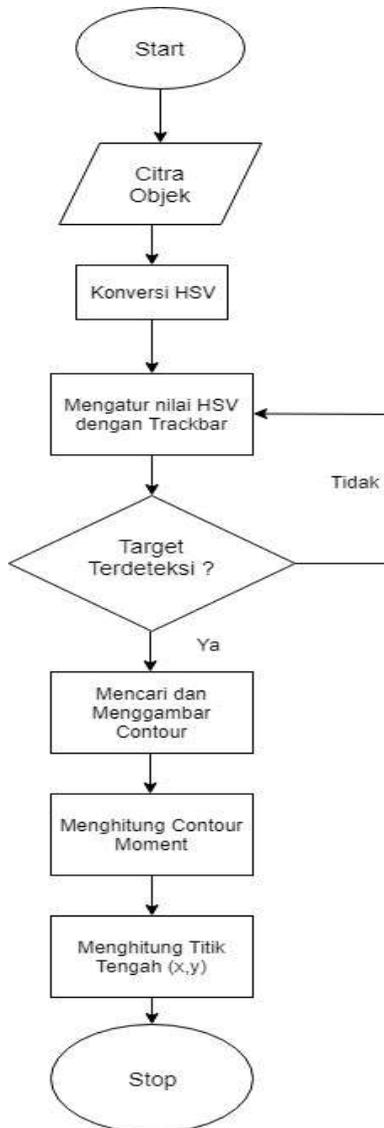


Gambar3. 4 Desain *mobile robot* pada *sketch up*

3.4.1 Perangkat Lunak Deteksi Objek

Pada perangkat lunak untuk melakukan deteksi objek seperti pada gambar 3.9 berupa flowchart yang dimulai dari pengambilan citra oleh kamera yang dikonversi ke dalam *color space* HSV. Kemudian dilakukan *smoothing* pada gambar hasil konversi. Nilai dari masing – masing *channel* pada HSV kemudian diubah menggunakan *interface* yang berasal dari fungsi *Trackbar*. Pengaturan nilai HSV bertujuan agar warna objek yang dikehendaki saja yang muncul pada layar.

Setelah mendapat citra dari objek, maka nilai *contour* dari objek akan dicari dan digambar. Setelah *contour* dari objek telah didapat, langkah berikutnya adalah mencari luas permukaan objek yang terlihat menggunakan *contour moment*. Dari nilai *moment* dapat dihitung titik tengah objek dengan membagi luas dengan *moment10* untuk sumbu x dan *moment01* untuk sumbu y.



Gambar3. 5 Flowchart Deteksi Objek

3.4.2 Perangkat Lunak Kontrol Kecepatan Motor

Perancangan perangkat lunak untuk mengatur kecepatan motor adalah menggunakan pengendali proporsional. Pengendali proporsional dirancang dengan melakukan *setting* pada nilai K_p (*gain*) dan menghitung nilai *error* yang didapatkan dengan mencari selisih antara nilai *set point* dengan nilai yang muncul dari output. Nilai *set point* yang digunakan adalah nilai dari titik tengah dari *frame* yaitu 319 pixel di sumbu x dan 239 di sumbu y. Output dalam hal ini berupa titik tengah objek yang telah dideteksi menggunakan kamera. Nilai yang didapat dari pengendali proporsional digunakan untuk mengatur sinyal PWM pada *motor driver* yang berfungsi mengatur kecepatan dan arah motor dc dalam bergerak mendekati stasiun pengisian daya.



Gambar3. 6 Flowchart Kontrol Kecepatan Motor

.....*Halaman ini sengaja dikosongkan*.....

BAB IV

PENGUJIAN DAN ANALISIS SISTEM

Bab ini membahas tentang pengujian dan analisis sistem dari *mobile robot* yang telah dirancang pada bab sebelumnya. Gambar 4.1 merupakan *mobile robot* yang digunakan dalam penelitian tugas akhir ini. Pengujian dan analisis bab ini dibagi menjadi berbagai tahap bagian, dengan membahas sub bagian dari perancangan sistem secara keseluruhan disertai tabel dan gambar yang mendukung pengujian dan analisis sistem.

4.1 Realisasi Desain *Mobile Robot*

Hasil, rancangan dari badan robot yang telah dibuat kemudian direalisasikan dengan membuat badan robot dengan panjang 310 mm dan lebar 210 mm dan membuat roda dari robot dengan diameter 90 mm pada *corel draw* dan dicetak dengan bahan akrilik untuk kemudian disusun seperti pada gambar 4.1. Pada gambar 4.1 terdapat bidang datar berbentuk segi delapan yang diasumsikan sebagai antena penerima saat melakukan pengisian baterai robot secara nirkabel. Ketika *mobile robot* telah sampai pada target maka antena penerima akan berada di bawah posisi dari antena pemancar dari stasiun pengisian daya dengan bentuk antena yang sama. Kemudian, di bagian paling depan dari badan robot diletakkan kamera berupa *webcam* yang digunakan untuk mendeteksi target yang akan dituju.



Gambar 4. 1 Hasil Realisasi *Mobile robot*

4.2 Pengujian Software

Pada pengujian *software* ini dilakukan dengan cara menguji perangkat lunak untuk melakukan deteksi terhadap objek, dan perangkat lunak untuk mengatur kecepatan motor dc.

4.2.1 Pengujian Deteksi Objek

Pada pengujian deteksi objek dilakukan proses deteksi terhadap warna dan bentuk objek yang menjadi target untuk didekati. Pengujian dilakukan dengan mengubah jarak asli dengan posisi kamera yang sama. Pada perangkat lunak untuk mendeteksi objek terdapat fungsi *trackbar* untuk mengubah nilai dari masing – masing *channel* HSV yaitu *hue*, *saturation*, dan *value*. Gambar yang digunakan adalah struktur berbentuk persegi panjang berjumlah tiga buah yang terdapat pada gambar 4.2. Tiga gambar tersebut memiliki warna biru yang berbeda – beda. Sebagai permisalan, gambar yang akan dijadikan target adalah persegi panjang di sisi paling kanan dan kiri dengan gambar di sisi kanan sebagai target I dan gambar di sisi kiri sebagai target II.

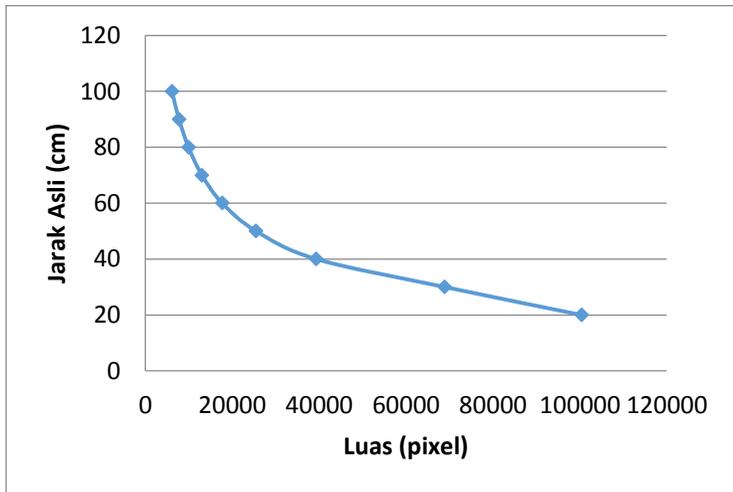
Nilai yang didapatkan adalah luas dari objek dan titik tengah yang didapatkan dari perhitungan menggunakan *contour moment*. Pengujian dilakukan sebanyak Sembilan kali dengan jarak asli mulai dari 20 cm hingga 100 cm dengan interval 10 cm. Hasil pengujian terdapat pada tabel 4.1 seperti di bawah ini.

Tabel 4.1 Hasil pengujian deteksi warna target I

No.	Jarak Asli(cm)	Hue	Saturation	Value	Luas (pixel)	Titik Tengah(x,y)
1	20	94	145	152	100411	145,183
2	30	96	140	185	68925	174,188
3	40	96	148	136	39324	199,233
4	50	97	154	143	25497	214,262
5	60	97	155	150	17703	229,282
6	70	97	159	150	13047	233,296
7	80	97	159	150	9962	237,307
8	90	97	159	152	7763	308,268
9	100	97	168	154	6155	247,322



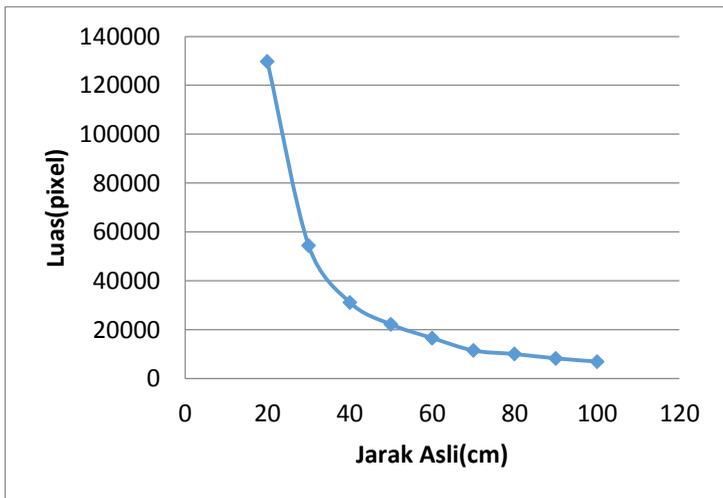
Gambar 4. 2 Tiga Persegi Panjang Berwarna Biru



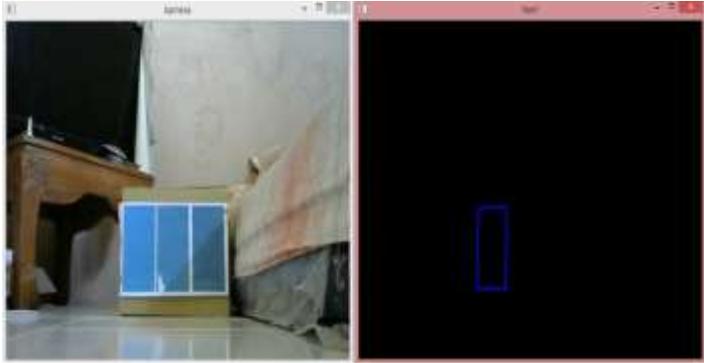
Gambar 4. 3 Grafik Perbandingan Jarak Asli dengan Luas *Target* I.

Tabel 4.2 Hasil pengujian deteksi warna target II

No.	Jarak Asli(cm)	Hue	Saturation	Value	Luas (pixel)	Titik Tengah(x,y)
1	20	84	93	100	129692	184,254
2	30	90	121	103	54432	198,326
3	40	94	100	101	31211	218,364
4	50	97	155	103	22167	222,381
5	60	97	150	103	16603	225,395
6	70	96	143	103	11516	230,404
7	80	96	154	105	10058	234,397
8	90	96	133	105	8282	236,318
9	100	96	136	108	6969	240,324



Gambar 4. 4 Grafik Perbandingan Jarak Asli dengan Luas Target II.



Gambar 4. 5 Hasil Target I pada Jarak 100 cm.



Gambar 4. 6 Hasil Target II pada Jarak 100 cm.

Dari hasil pada tabel 4.1 dapat dibuat sebuah grafik perbandingan antara jarak asli objek dengan kamera dan luas *contour* dari objek yaitu persegi panjang sisi paling kanan yang dapat dilihat pada gambar 4.2 dengan hasil berupa grafik seperti pada gambar 4.3

Untuk percobaan berikutnya, digunakan persegi panjang di sisi paling kiri yang terdapat pada gambar 4.2. Hasil percobaan dapat dilihat pada tabel 4.2. Perbandingan dari luas dan jarak asli pada percobaan kedua dapat dilihat pada gambar 4.4. Bila dilihat pada tabel 4.1 dan tabel 4.2, nilai *hue*, *saturation*, dan *value* pada tabel 4.1 cenderung lebih stabil pada jarak yang berbeda – beda. Dapat disimpulkan bahwa pada gambar

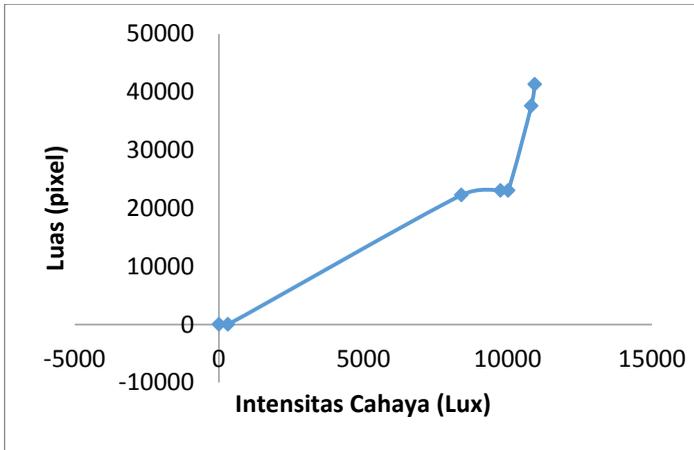
4.2, persegi panjang pada sisi paling kanan memiliki hasil yang paling baik untuk dijadikan target untuk melakukan *autodocking*. Kesimpulan ini didukung dengan hasil yang terdapat pada gambar 4.5 dan 4.6.

4.2.2 Pengujian Target pada Berbagai Macam Kondisi Pencahayaan

Pada bagian ini dilakukan pengujian target pada jarak 50 cm dan 100 cm dengan kondisi pencahayaan yang berbeda – beda. Target diberi penerangan berupa lampu bohlam yang tingkat pencahayaannya diatur oleh *dimmer*. Intensitas cahaya dari lampu akan diukur oleh lux meter. Kondisi pengujian dilakukan dengan nilai *hue*, *saturation*, dan *value* yang bersifat konstan atau tetap. Hasil dari pengujian dapat dilihat pada tabel 4.3 dan tabel 4.4 dimana tabel 4.3 adalah hasil pengujian dengan jarak asli sebesar 50 cm dan tabel 4.4 merupakan hasil pengujian pada jarak asli 100 cm.

Tabel 4.3 Hasil Pengujian Target pada Jarak 50 cm

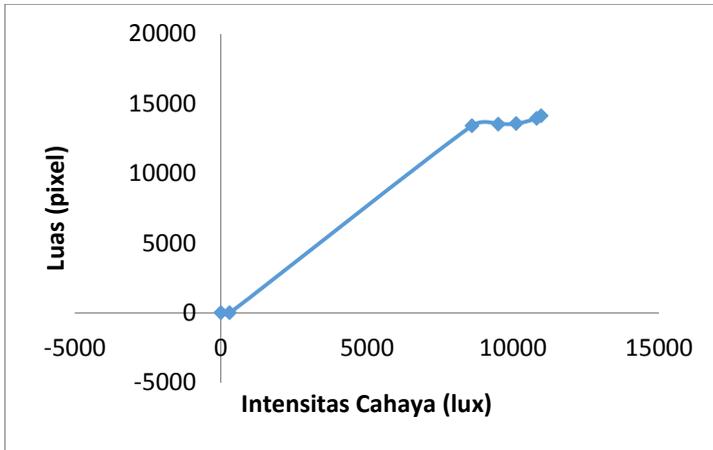
No.	Intensitas Cahaya (lux)	Luas (pixel)	Titik Tengah (x,y)
1	0	0	0
2	309	0	0
3	8400	22270	290,321
4	9760	23048	302,322
5	10020	23068	269,216
6	10820	37585	279,267
7	10940	41304	274,276



Gambar 4. 7 Grafik Perbandingan Luas dan Intensitas Cahaya pada Jarak 50 cm

Tabel 4.4 Hasil Pengujian Target pada Jarak 100 cm

No.	Intensitas Cahaya (lux)	Luas (pixel)	Titik Tengah (x,y)
1	0	0	0
2	304	0	0
3	8600	13414	247,339
4	9500	13532	246,338
5	10120	13561	246,338
6	10820	13931	246,337
7	10970	14128	245,336



Gambar 4. 8 Grafik Perbandingan Luas dan Intensitas Cahaya pada Jarak 100 cm

Hasil dari dua kali pengujian yaitu pada jarak 50 cm dan 100 cm menunjukkan hasil bahwa semakin terang cahaya yang diberikan, maka semakin jelas citra yang muncul dari target. Dapat dilihat pada gambar 4.7 dan 4.8 bahwa ketika intensitas cahaya semakin besar maka nilai luas dari target juga semakin besar. Pada nilai dari titik tengah target yang terdeteksi, tidak terjadi perubahan yang cukup signifikan atau cenderung tetap pada setiap percobaan yang dilakukan dengan intensitas cahaya yang berbeda – beda baik pada pengujian dengan jarak 50 cm atau 100 cm antara kamera dengan target.

4.2.3 Pengujian Berdasarkan Letak Sumber Cahaya

Pada bagian ini dilakukan pengujian target pada jarak 100 cm dengan kondisi pencahayaan dan letak sumber cahaya yang berbeda – beda. Sumber cahaya diletakkan pada tiga posisi yang berbeda. Posisi peletakkan yang pertama yaitu pada bagian atas *mobile robot*. Posisi kedua yaitu dari samping dengan jarak dari target maupun robot yaitu 50 cm. Posisi ketiga adalah pada bagian atas dari target. Hasil dari masing – masing pengujian dapat dilihat pada tabel 4.5, tabel 4.6, dan tabel 4.7. Untuk posisi dari peletakkan sumber cahaya terdapat pada gambar 4.9, gambar 4.10, dan gambar 4.11.



Gambar 4. 9 Sumber Cahaya pada Posisi 1



Gambar 4. 10 Sumber Cahaya pada Posisi 2



Gambar 4. 11 Sumber Cahaya pada Posisi 3

Tabel 4.5 Hasil Pengujian Target pada Posisi 1

No.	Intensitas Cahaya (lux)	Luas (pixel)	Titik Tengah (x,y)
1	0	0	0
2	300	0	0
3	8400	25367	334,251
4	9200	25288	334,251
5	10300	25299	334,251
6	11300	25274	334,251

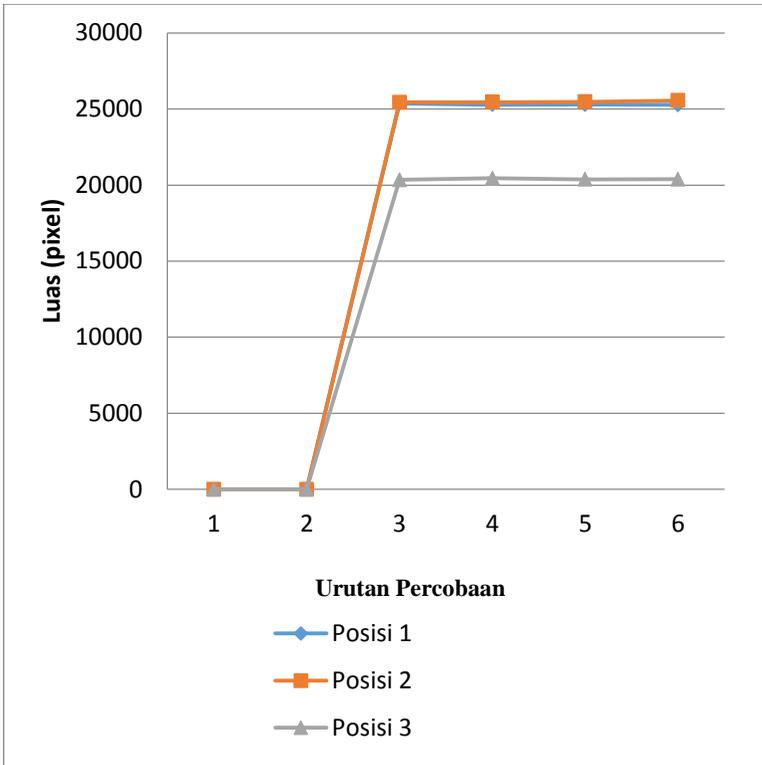
Tabel 4.6 Hasil Pengujian Target pada Posisi 2

No.	Intensitas Cahaya (lux)	Luas (pixel)	Titik Tengah (x,y)
1	0	0	0
2	200	0	0
3	8300	25448	330,392
4	9500	25451	333,393
5	10300	25472	340,250
6	11400	25556	340,250

Tabel 4.7 Hasil Pengujian Target pada Posisi 3

No.	Intensitas Cahaya (lux)	Luas (pixel)	Titik Tengah (x,y)
1	0	0	0
2	200	0	0
3	8400	20341	348,249
4	9500	20447	348,249
5	10500	20377	348,249
6	11200	20385	348,249

Pengujian dengan posisi sumber cahaya yang berbeda – beda dilakukan sebanyak enam kali dan hasilnya terdapat pada tabel 4.5 untuk posisi pertama, tabel 4.6 untuk posisi kedua, dan tabel 4.7 untuk posisi ketiga. Pengujian dengan tiga posisi yang berbeda menggunakan intensitas cahaya yang nilainya diubah pada setiap percobaan untuk mengetahui pengaruhnya terhadap kamera saat mendeteksi target. Perbedaan posisi dari sumber cahaya digunakan untuk mengetahui letak dan arah sumber cahaya mana yang dapat memberikan pengaruh yang signifikan terhadap kamera saat melakukan penangkapan citra dari target yang akan dituju oleh *mobile robot*.



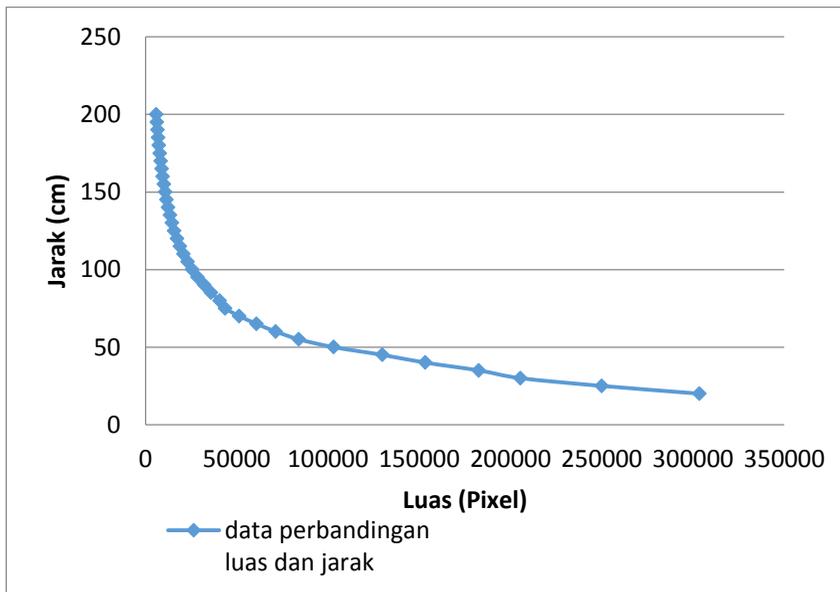
Gambar 4. 12 Grafik Perbandingan Luas

Hasil pengujian dari tiga posisi peletakkan sumber cahaya yang terdapat pada tabel 4.5, tabel 4.6, dan tabel 4.7 menunjukkan bahwa sumber cahaya pada posisi 1 dan 2 masih menunjukkan hasil yang hampir sama. Hasil pada tabel 4.7 dengan letak sumber cahaya pada posisi 3 menunjukkan perbedaan dari posisi yang lain dengan selisih nilai luas yang terdeteksi sekitar 5000 *pixel*. Perbedaan nilai luas ini menunjukkan bahwa pengaruh dari cahaya yang paling kuat adalah ketika sumber cahaya datang dari arah target. Perbedaan nilai luas target pada setiap posisi sumber cahaya dapat dilihat pada grafik di gambar 4.12.

4.2.4 Pengujian Jarak terhadap Target dengan Regresi

Pada bagian ini dilakukan pengujian untuk mendapatkan jarak antara *mobile robot* dengan target dengan mengambil data luas target terhadap target dengan menggunakan jarak asli sebesar 20 cm hingga 200 cm dengan interval 5 cm. Grafik dari data ini dapat dilihat pada gambar 4.13. Hasil dari pengambilan data ini akan dihitung menggunakan metode regresi yang didapat dari *trendline* data pada gambar 4.13. Hasil perhitungan dari regresi data perbandingan luas dan jarak terdapat pada tabel 4.8.

Data hasil pengujian pada tabel 4.8 berupa perbandingan antara jarak asli dengan jarak hasil perhitungan menggunakan metode regresi. Dari hasil pengujian jarak terhadap target menggunakan metode regresi memiliki hasil dengan *error* paling tinggi saat pengujian pada jarak 20 cm, dengan *error* sebesar 65%. Untuk pengujian lainnya, nilai *error* yang didapatkan tidak ada yang besarnya melebihi 5%. Tingkat keberhasilan rata – rata yang didapatkan yaitu 96,44%.



Gambar 4. 13 Grafik Perbandingan Jarak dan Luas

Tabel 4.8 Data Perbandingan Jarak asli dengan Jarak Hasil Regresi

No.	Jarak Asli (cm)	Jarak Hasil Regresi (cm)	Error (cm)	Tingkat Keberhasilan (%)
1	20	27	7	65
2	30	32	2	93,4
3	40	40	0	100
4	50	48	2	96
5	60	59	1	98,3
6	70	70	0	100
7	80	81	1	98,75
8	90	92	2	98,9
9	100	103	3	97
10	110	114	4	96,4
11	120	122	2	98,3
12	130	133	3	97,7
13	140	139	1	99,3
14	150	148	2	98,7
15	160	161	1	99,4
16	170	170	0	100
17	180	183	3	98,3
18	190	192	2	98,9
19	200	204	4	98

4.2.5 Pengujian *Mobile Robot* dengan *Set Point* Luas Objek

Pada bagian ini dilakukan pengujian terhadap pengendali proporsional yang digunakan untuk mengendalikan kecepatan motor. Input yang digunakan diantaranya nilai K_p atau *gain* yang digunakan untuk mendapatkan output dan nilai *set point* berupa luas dari *frame*. Nilai *error* didapatkan dengan menghitung selisih *set point* dengan luas objek yang muncul selama kamera mengambil citra. Luas *frame* yang digunakan sebagai *set point* adalah 303849 *pixel*. Percobaan dilakukan dengan menentukan bahwa robot berhenti saat berada pada jarak asli sebesar 20 cm dari target. Gambar objek yang digunakan yaitu gambar persegi panjang di sisi paling kanan dari gambar 4.2 karena warna dari persegi panjang tersebut dianggap memiliki hasil yang lebih baik

berdasarkan kesimpulan pada subbab sebelumnya. Pada pengujian dengan menggunakan luas sebagai *set point*, kontrol yang dapat dilakukan pada motor dc hanya berupa kontrol kecepatan saja tanpa mengatur arah kanan atau kiri pada pergerakan *mobile robot* sehingga percobaan hanya dilakukan dalam keadaan lurus, Hasil dari percobaan ini dapat dilihat pada tabel 4.9

Pada tabel 4.9 terlihat bahwa hasil dari pengujian dengan menggunakan *set point* berupa luas objek atau target memiliki tingkat keberhasilan yang cukup tinggi walaupun pengujian ini hanya dilakukan dalam keadaan jalur yang lurus. Karena saat menggunakan luas sebagai *set point* hanya dapat digunakan untuk menentukan kecepatan motor dc tanpa memperhatikan arah dari *mobile robot*.

Pengujian dengan tingkat keberhasilan paling tinggi terdapat pada percobaan dengan jarak 60 cm, 160 cm, 200 cm, dan 220 cm yaitu sebesar 97,5%. Sedangkan pengujian dengan tingkat keberhasilan paling rendah terdapat pada percobaan pada jarak 80 cm dengan persentase sebesar 65%. Pengujian dengan tingkat keberhasilan tertinggi memiliki *error* sebesar 0,5 cm dari target, sedangkan pengujian dengan tingkat keberhasilan paling rendah memiliki *error* sebesar 7 cm dari target. Nilai *error* yang tidak tetap pada setiap percobaan dipengaruhi oleh intensitas cahaya yang selalu berubah setiap waktu.

Tabel 4.9 Hasil pengujian *mobile robot* dengan *set point* luas objek

No.	Jarak Awal dari Target (cm)	Jarak Saat Berhenti (cm)	Error (cm)	Tingkat Keberhasilan (%)
1	60	19,5	0,5	97,5
2	80	13	7	65
3	100	18	2	90
4	120	21	1	95
5	140	24	4	80
6	160	19,5	0,5	97,5
7	180	26,5	6,5	67,5
8	200	19,5	0,5	97,5
9	220	19,5	0,5	97,5

4.2.6 Pengujian *Mobile Robot* dengan *Set Point* Titik Tengah Objek

Pada bagian ini dilakukan pengujian yang sama dengan pengujian sebelumnya, yaitu pengujian motor dc *mobile robot* dengan kontrol proporsional. Perbedaannya adalah *set point* yang digunakan dalam percobaan ini adalah titik tengah objek yang terdeteksi. Jarak asli yang digunakan saat robot berhenti tetap sama, yaitu 20 cm. Nilai titik tengah masing – masing sumbu (x,y) yang dijadikan *set point* adalah 319 dan 239 *pixel*. Pada pengujian menggunakan titik tengah sebagai *set point*, dapat dilakukan kontrol terhadap kecepatan motor dc dan arah dari *mobile robot* agar dapat berbelok ke kanan atau ke kiri sesuai letak target yang dituju. Hasil pengujian terdapat pada tabel 4.10.

Hasil dari pengujian gerak *mobile robot* menggunakan titik tengah target sebagai *set point* menghasilkan tingkat keberhasilan rata – rata sebesar 77,5%. Hasil ini memiliki tingkat keberhasilan yang lebih kecil daripada menggunakan luas dari target, tetapi pada pengujian ini *mobile robot* tidak hanya dapat mengatur kecepatan motornya saja tetapi juga dapat menentukan arah saat bergerak menuju ke target sehingga pergerakan robot dapat dilakukan tidak hanya dalam kondisi jalur yang lurus.

Tabel 4.10 Hasil pengujian *mobile robot* dengan *set point* titik tengah objek

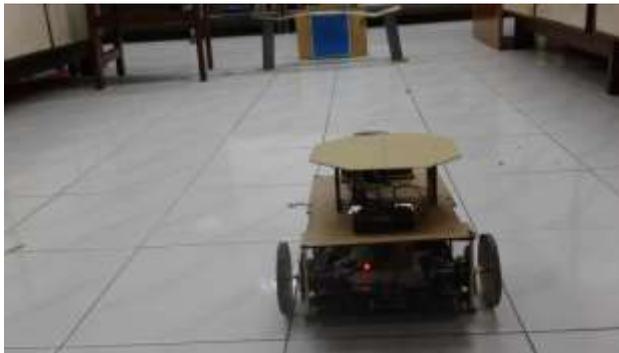
No.	Jarak Awal dari Target (cm)	Jarak Saat Berhenti (cm)	Error (cm)	Tingkat Keberhasilan (%)
1	60	17	3	85
2	80	15	5	75
3	100	13	7	65
4	120	15,5	4,5	77,5
5	140	14	6	70
6	160	17	3	85
7	180	16	4	80
8	200	14	6	70
9	220	18	2	90

4.2.7 Pengujian dengan Posisi Awal yang Berbeda - beda

Pada bagian ini dilakukan pengujian dari *mobile robot* menggunakan posisi awal yang berbeda – beda. Pengujian dilakukan dengan empat posisi awal berbeda seperti pada gambar 4.14, 4.15, 4.16, dan 4.17. Pengujian ini dilakukan untuk melihat perbandingan antara luas dengan titik tengah sebagai *set point*. Pengujian dilakukan pada jarak tetap yaitu 240 cm. Pengujian ini juga bertujuan untuk mengetahui pengaruh dari posisi robot terhadap proses deteksi objek yang dilakukan oleh kamera untuk mendapatkan citra dari target yang akan dituju. Hasil pengujian dapat dilihat pada tabel 4.10.



Gambar 4. 14 Posisi Awal 1



Gambar 4. 15 Posisi Awal 2



Gambar 4. 16 Posisi Awal 3



Gambar 4. 17 Posisi Awal 4

Tabel 4.11 Hasil pengujian *mobile robot* dengan posisi awal berbeda

No.	Posisi Awal	Hasil dengan Set Point Luas	Hasil dengan Set Point Titik Tengah
1	1	Gagal	Berhasil
2	2	Gagal	Berhasil
3	3	Berhasil	Berhasil
4	4	Gagal	Gagal

Pada bagian ini dilakukan pengujian untuk mengetahui bila posisi awal dari *mobile robot* berubah – ubah akan memiliki tingkat keberhasilan yang baik atau tidak. Untuk pengujian dengan menggunakan luas sebagai *set point* hanya berhasil sebanyak satu kali yaitu saat dilakukan pengujian pada posisi awal tiga. Pada pengujian dengan titik tengah sebagai *set point*, robot dapat berhasil mencapai target sebanyak tiga kali dan mengalami kegagalan sebanyak satu kali yaitu pada pengujian di posisi awal empat. Tingkat keberhasilan lebih tinggi terdapat pada pengujian dengan menggunakan titik tengah sebagai *set point* dengan persentase sebesar 75% karena dalam empat kali pengujian, *mobile robot* berhasil tiba di target sebanyak tiga kali.

Pengujian *mobile robot* dengan menggunakan titik tengah dari *frame* sebagai *set point* memberikan hasil yang lebih baik karena *error* yang didapat dari selisih antara titik tengah *frame* dengan titik tengah dari target yang terdeteksi oleh kamera dapat dijadikan sebagai input untuk kontrol proporsional dalam menentukan arah gerak dari *mobile robot*.

4.2.8 Pengujian Akhir *Mobile Robot* Terhadap Stasiun Pengisian Daya

Pada bagian ini dilakukan pengujian yang bersifat final dengan menggunakan objek yang disimulasikan sebagai stasiun pengisian daya dengan penanda menggunakan warna biru sesuai dengan hasil percobaan sebelumnya, hanya saja memiliki ukuran yang lebih besar agar dapat dilihat oleh *mobile robot* dari jarak yang cukup jauh. Penanda digunakan untuk mengindikasikan bahwa objek adalah target yang akan dituju oleh *mobile robot*. Hasil pengujian akhir ini dapat dilihat pada tabel 4.12 dengan melakukan percobaan sebanyak enam kali. Pengujian dilakukan dengan mengambil jarak yang cukup jauh yaitu mulai dari 250 cm hingga 500 cm dengan interval antar setiap percobaan yaitu 50 cm. Jarak yang cukup jauh ini digunakan untuk mengetahui seberapa baik kamera dalam melakukan deteksi terhadap target yang akan dituju oleh *mobile robot*.

Pada pengujian ini, jarak yang digunakan sebagai acuan saat *mobile robot* berhenti atau telah sampai pada target yang dituju adalah 20 cm. Dari hasil pada tabel 4.12 menunjukkan *error* yang terus berubah – ubah. Hal ini dapat disebabkan oleh kondisi pencahayaan yang tidak tetap. *Error* paling kecil terdapat pada pengujian dengan jarak terjauh yaitu 500 cm dengan *error* 0,5 cm dari hasil yang diharapkan.

Error ini muncul karena beberapa faktor diantaranya adalah pengaruh dari intensitas cahaya yang selalu berubah dari lingkungan sekitar dan kemampuan kamera dalam menangkap serta mengolah citra sehingga didapatkan citra dari target. Secara keseluruhan, tingkat keberhasilan secara rata – rata pada pengujian ini adalah 75 %

Error yang terjadi saat *mobile robot* mencapai target selalu berubah dikarenakan kondisi cahaya dari lingkungan sekitar yang intensitasnya dapat memberi pengaruh yang cukup signifikan pada kamera dalam melakukan penangkapan citra agar dapat mendeteksi target yang akan dituju.

Tabel 4.12 Hasil pengujian akhir pada *mobile robot*

No.	Jarak Awal dari Target (cm)	Jarak Saat Berhenti (cm)	Error (cm)	Tingkat Keberhasilan (%)
1	250	10,5	9,5	52,5
2	300	14,5	5,5	72,5
3	350	16	4	80
4	400	16	4	80
5	450	13,5	6,5	67,5
6	500	20,5	0,5	97,5

BAB V PENUTUP

5.1 Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah semakin dekat jarak kamera dengan target maka semakin kecil nilai luas objek yang dideteksi oleh kamera. Hal ini dapat dijadikan parameter untuk *mobile robot* dalam bergerak menuju stasiun pengisian daya. Dari luas objek juga didapatkan titik tengah dari objek yang dapat dijadikan sebagai *set point* dalam menentukan kerja dari pengendali proporsional untuk mengatur kecepatan motor dc dalam bergerak menuju stasiun pengisian daya sehingga *mobile robot* mampu mendapatkan posisi yang tepat dalam melakukan *docking* otomatis. Hasil yang didapat yaitu pengujian dengan menggunakan luas objek sebagai *set point* memiliki tingkat keberhasilan dengan rata – rata sebesar 87,5% dengan catatan pengujian ini hanya dapat dilakukan dalam kondisi lurus. Pada pengujian dengan menggunakan titik tengah sebagai *set point* memberi tingkat keberhasilan dengan rata – rata sebesar 77,5% dan mampu mengatur arah kanan dan kiri motor dc. Sehingga *set point* yang digunakan pada percobaan akhir adalah titik tengah dari objek. Pada pengujian dengan posisi awal dari robot yang berbeda – beda yaitu sebanyak empat posisi, pengujian dengan menggunakan titik tengah sebagai *set point* memiliki tingkat keberhasilan yang lebih banyak daripada menggunakan luas target yaitu sebesar 75%. Hasil pengujian akhir pada *mobile robot* dengan melakukan pengujian sebanyak enam kali memiliki tingkat keberhasilan sebesar 75%.

5.2 Saran

Beberapa saran untuk pengembangan tugas akhir ini adalah :

1. Metode untuk menghitung jarak dengan target dapat menggunakan *stereo imaging* untuk mempermudah dalam mendeteksi jarak antara robot dengan target.
2. Metode untuk mendeteksi target yang memiliki hasil lebih baik daripada dengan mendeteksi warna dapat menggunakan *template matching* untuk mencocokkan antara *template* target dengan hasil dari kamera.

.....*Halaman ini sengaja dikosongkan*.....

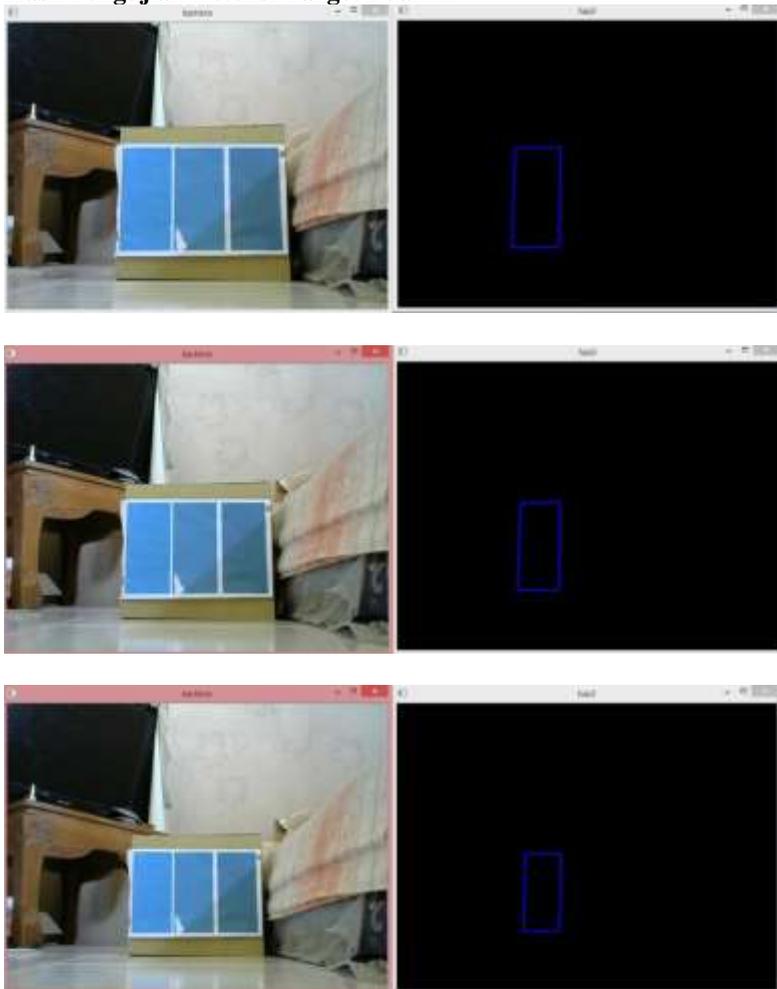
DAFTAR PUSTAKA

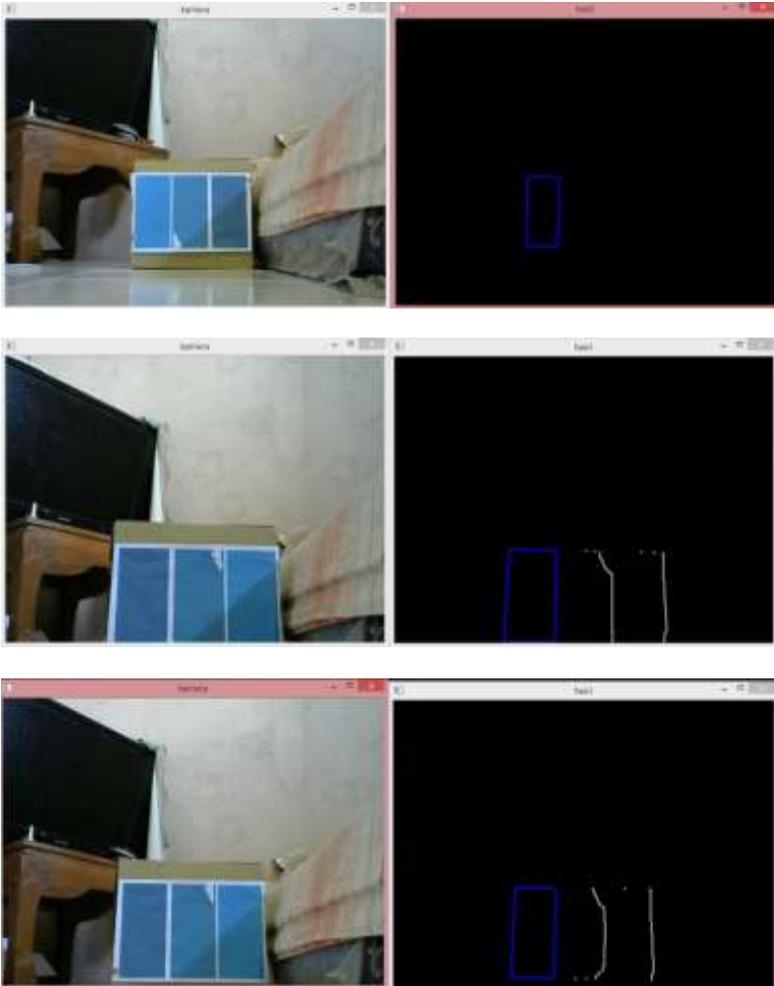
- [1] Meshram, P. M., & Kanojiya, R. G. "Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor." *IEEE-International Conference On Advances in Engineering, Science And Management (ICAESM -2012)*, 117 - 122.2012
- [2] D. Krisrenanto, M. Rivai, & F. Budiman. "Identifikasi Jumlah dan Tingkat Aktivitas Orang Berbasis Pengolahan Citra Berbasis Raspberry Pi." *JURNAL TEKNIK ITS Vol.6.No.1.(2017)*, 110-115.2017
- [3] Suman, S. K., & Giri, V. K. "Speed control of DC motor using optimization techniques based PID Controller." *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, 581-587.2016
- [4] <URL:<https://www.logitech.com/en-in/product/hd-webcam-c270h#specification-tabular.html>>
- [5] Tjahyadi, Christianto. <URL : <http://christianto.tjahyadi.com/belajar-robotik/driver-motor-l298.html>>. 27 Februari 2017.
- [6] G. R. Bradski dan A. Kaehler, *Learning OpenCV: computer vision with the OpenCV library*, 1. ed., [Nachdr.]. USA: O'Reilly, 2008.
- [7] A. A. Kurniawan, M. Rivai, & F. Budiman. Sistem Pemandu Pendaratan pada Balon Udara Berbasis Pengolahan Citra dan Kendali PID. *JURNAL TEKNIK ITS Vol 5.No.2*, 179-184.2016
- [8] Reis, M. R., Silva, F. S., Araujo, W. R., Calixto, W. P., Araujo, D. S., Mendes, L. N., et al. "Speed control for direct current motor using optimization tuning for PID controller." *2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, 1 - 4.2016

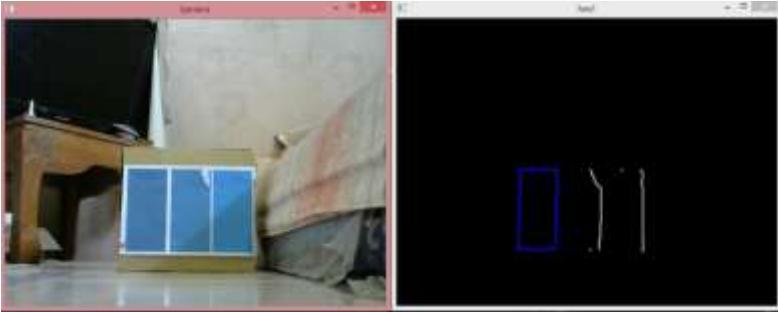
- [9] Meshram, P. M., & Kanojiya, R. G. "Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor." *IEEE-International Conference On Advances in Engineering, Science And Management (ICAESM -2012)*, 117 - 122.2012
- [10] Lang, H., Wang, Y., & Clarence, W. d. "Vision based object identification and tracking for mobile robot visual servo control." *IEEE ICCA 2010*, 92-96.2010
- [11] Luo, R. C., Liao, C. T., & Lin, S. C. "Multi-sensor fusion for reduced uncertainty in autonomous mobile robot docking and recharging." *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2203-2208.2009
- [12] Wang, W., Li, Z., Yu, W., & Zhang, J. "An autonomous docking method based on ultrasonic sensors for self-reconfigurable mobile robot." *2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 1744-1749.2009

LAMPIRAN

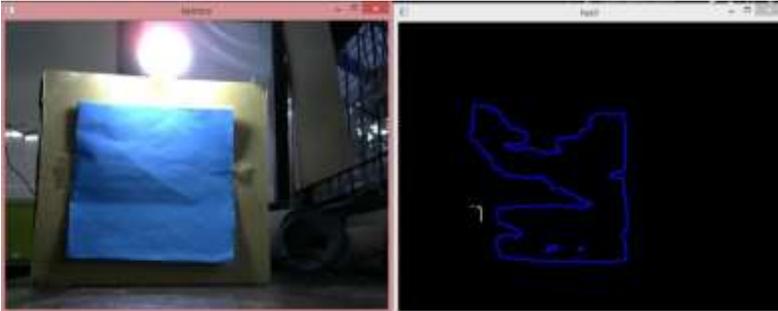
Hasil Pengujian Deteksi Target I dan II

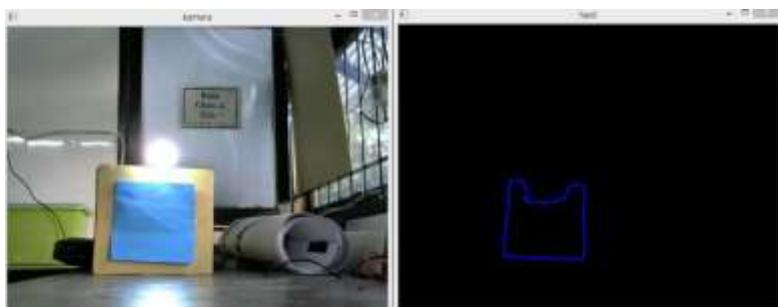






Hasil Pengujian Target Terhadap Tingkat Intensitas Cahaya





BIODATA PENULIS



Penulis bernama Ariestya Putra Suasnto, lahir di Surabaya pada tanggal 19 Juli 1996. Penulis merupakan anak pertama dari dua bersaudara. Pendidikan formal yang ditempuh mulai dari tingkat sekolah dasar adalah SDN Ketabang III Surabaya mulai dari tahun 2002 - 2008, pada jenjang SMP yaitu SMPN 37 Surabaya mulai dari tahun 2008 - 2011, dan pada jenjang SMA yaitu SMAN 5 Surabaya mulai tahun 2011 – 2014. Pendidikan tinggi yang ditempuh penulis saat ini adalah Teknik Elektro ITS yang dimulai tahun 2014 hingga sekarang. Di departemen teknik elektro, penulis mengambil peminatan bidang studi elektronika mulai tahun 2016.