

**RANCANG BANGUN *MOBILE* ROBOT PENGIKUT OBJEK
BERDASARKAN WARNA DAN BENTUK MENGGUNAKAN *TEMPLATE*
MATCHING BERBASIS MINI PC**

LAPORAN TUGAS AKHIR

**RENDY DHARTA NUGRAHA
1010453004**



**JURUSAN SISTEM KOMPUTER
FAKULTAS TEKNOLOGI INFORMASI**

UNIVERSITAS ANDALAS

PADANG

2016

**RANCANG BANGUN *MOBILE* ROBOT PENGIKUT OBJEK
BERDASARKAN WARNA DAN BENTUK MENGGUNAKAN *TEMPLATE*
MATCHING BERBASIS MINI PC**

LAPORAN TUGAS AKHIR

*Sebagai Salah Satu Syarat Untuk Menyelesaikan Program Sarjana
Pada Jurusan Sistem Komputer Universitas Andalas*

RENDY DHARTA NUGRAHA

1010453004



**JURUSAN SISTEM KOMPUTER
FAKULTAS TEKNOLOGI INFORMASI**

UNIVERSITAS ANDALAS

PADANG

2016

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Saya mahasiswa/dosen/tenaga kependidikan* Universitas Andalas yang bertanda tangan di bawah ini:

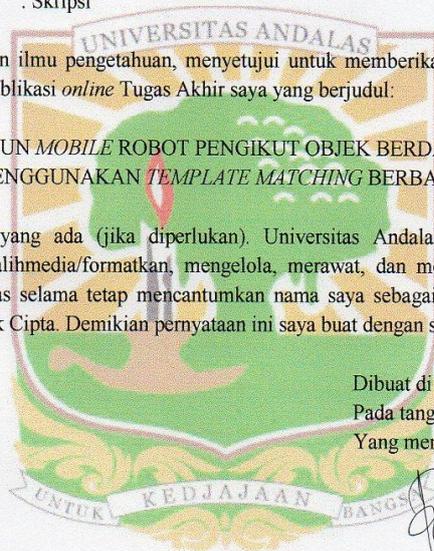
Nama lengkap : RENDY DHARTA NUGRAHA
No. BP/NIM/NIDN : 1010453004
Program Studi : SISTEM KOMPUTER
Fakultas : TEKNOLOGI INFORMASI
Jenis Tugas Akhir : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Andalas hak atas publikasi *online* Tugas Akhir saya yang berjudul:

RANCANG BANGUN *MOBILE* ROBOT PENGIKUT OBJEK BERDASARKAN WARNA DAN BENTUK MENGGUNAKAN *TEMPLATE MATCHING* BERBASIS MINI PC.

berserta perangkat yang ada (jika diperlukan). Universitas Andalas juga berhak untuk menyimpan, mengalihmedia/formatkan, mengelola, merawat, dan mempublikasikan karya saya tersebut di atas selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Padang
Pada tanggal 21 juni 2016
Yang menyatakan,



(Rendy Dharta Nugraha)

* pilih sesuai kondisi

** termasuk laporan penelitian, laporan pengabdian masyarakat, laporan magang, dll

LEMBARAN PERSETUJUAN TUGAS AKHIR

Dengan ini dinyatakan bahwa :

Nama : Rendy Dharta Nugraha

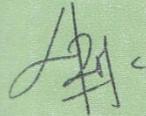
No.BP : 1010453004

Judul Tugas Akhir : **Rancang Bangun *Mobile Robot Pengikut Objek Berdasarkan Warna Menggunakan *Template Matching* Berbasis Mini PC***

Telah diujikan dan telah disetujui Seminar Hasil Tugas Akhirnya sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom) melalui ujian komprehensif yang diadakan pada tanggal 29 Maret 2016 berdasarkan ketentuan yang berlaku.

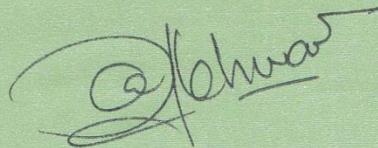
Padang, 29 Maret 2016

Penguji 1



Lathifah Arief, MT
NIP. 198109122014042001

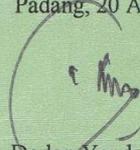
Penguji 2



Dody Ichwana Putra, MT
NIP.198611072015041001

Mengetahui :

Ketua Jurusan Sistem Komputer
Fakultas Teknologi Informasi
Universitas Andalas
Padang, 20 April 2016



Dodon Yendri, M.Kom
NIP. 196603091986031001

HALAMAN PENGESAHAN

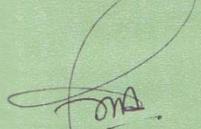
Nama : RENDY DHARTA NUGRAHA
No.BP : 1010453004
Judul Tugas Akhir : Rancang Bangun *Mobile* Robot Pengikut Objek Berdasarkan
Warna Menggunakan *Template Matching* Berbasis Mini PC

Tugas Akhir ini disetujui oleh Dosen Pembimbing dan disahkan oleh Ketua
Jurusan Sistem Komputer, Fakultas Teknologi Informasi Universitas Andalas.

Demikianlah lembaran pengesahan ini dibuat untuk diketahui bersama.

Padang, 20 April 2016

Pembimbing 1



FIRDAUS M.T

NIP.19770522208121001

Pembimbing 2

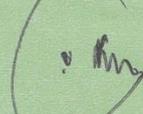


DERISMA M.T

NIP.19820419201012201

Mengetahui :

Ketua Jurusan Sistem Komputer
Fakultas Teknologi Informasi
Universitas Andalas
Padang, 20 April 2016



Dodon Yendri, M.Kom

NIP. 196603091986031001

HALAMAN PENGESAHAN

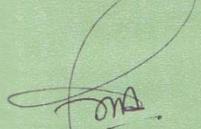
Nama : RENDY DHARTA NUGRAHA
No.BP : 1010453004
Judul Tugas Akhir : Rancang Bangun *Mobile* Robot Pengikut Objek Berdasarkan
Warna Menggunakan *Template Matching* Berbasis Mini PC

Tugas Akhir ini disetujui oleh Dosen Pembimbing dan disahkan oleh Ketua
Jurusan Sistem Komputer, Fakultas Teknologi Informasi Universitas Andalas.

Demikianlah lembaran pengesahan ini dibuat untuk diketahui bersama.

Padang, 20 April 2016

Pembimbing 1



FIRDAUS M.T

NIP.19770522208121001

Pembimbing 2

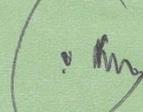


DERISMA M.T

NIP.19820419201012201

Mengetahui :

Ketua Jurusan Sistem Komputer
Fakultas Teknologi Informasi
Universitas Andalas
Padang, 20 April 2016



Dodon Yendri, M.Kom

NIP. 196603091986031001

LEMBARAN PERSETUJUAN TUGAS AKHIR

Tim Penguji Tugas Akhir Mahasiswa Jurusan Sistem Komputer, menyatakan bahwa :

Nama : Rendy Dharta Nugraha

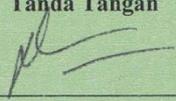
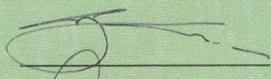
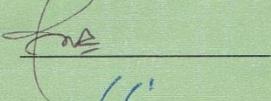
No.BP : 1010453004

Judul Tugas Akhir : Rancang Bangun *Mobile Robot* Pengikut Objek Berdasarkan
Warna Menggunakan *Template Matching* Berbasis Mini PC

Telah diujikan dan telah disetujui Seminar Hasil Tugas Akhirnya pada seminar hasil
untuk dilanjutkan pada tahap berikutnya.

Demikianlah lembaran pengesahan ini dibuat untuk diketahui bersama.

Padang, 20 April 2016

	Nama	Tanda Tangan
Penguji I	: Ratna Aisuwarya , M.Eng	
Penguji II	: Ir. Werman Kasoep , M.Kom	
Pembimbing I	: Firdaus , M.T	
Pembimbing II	: Derisma , M.T	

Mengetahui :
Ketua Program Studi Sistem Komputer


Dodon Yendri, M.Kom
NIP. 196603091986031001

PERNYATAAN

Dengan ini menyatakan bahwa :

1. Tugas Akhir dengan judul “Rancang Bangun *Mobile Robot* Pengikut Objek Berdasarkan Warna Menggunakan *Template Matching* Berbasis Mini PC” adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana, Magister, dan Doktor), baik di Universitas Andalas maupun di perguruan tinggi lainnya.
2. Tugas Akhir ini murni gagasan dan rancangan saya sendiri, tanpa bantuan tidak sah dari pihak lain, kecuali bantuan dan arahan dari tim pembimbing.
3. Tugas Akhir ini tidak terdapat hasil karya atau pendapat yang ditulis atau dipublikasikan oleh orang lain, kecuali dikutip secara tertulis dengan jelas dan dicantumkan sebagai acuan dalam tulisan saya dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya, dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik serta sanksi lainnya sesuai dengan norma dan ketentuan lain yang berlaku.

Demikianlah surat ini dibuat, untuk dapat dipergunakan sebagaimana mestinya.

Padang, 20 April 2016

Yang membuat pernyataan,



Rendy Dharta Nugraha
No.BP 1010453004

SALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

"Dia memberikan hikmah (ilmu yang berguna) kepada siapa yang dikehendaki-Nya. Barang siapa yang mendapat hikmah itu, sesungguhnya ia telah mendapat kebajikan yang banyak. Dan tiadalah yang menerima peringatan melainkan orang-orang yang berakal".

(Q.S. Al-Baqarah: 269)

Dan bahwasanya setiap manusia itu tiada akan memperoleh hasil selain apa yang telah diusahannya

(Q.S. An-Najm:39)

"Karena sesungguhnya sesudah kesulitan itu ada kemudahan"

(Q.S. Ash-Sharh: 5-6)

Niscaya Allah akan mengangkat (derajat) orang-orang yang beriman diantaramu dan orang-orang yang diberi ilmu beberapa derajat

(QS : Al-Mujadilah 11)

"Maka Nikmat Tuhanmu yang manakah yang kamu dustakan"

(Q.S: Ar-Rahman: 13)

"

"

Alhamdulillahirabbil alamin... Alhamdulillahirabbil alamin...

Alhamdulillahirabbil alamin...

Sembah sujud serta puji dan syukurku pada-MU Allah SWT. Dengan

cinta, kasih sayang, rahmat dan hidayah-MU telah memberikan ku

kekuatan, kesehatan, semangat pantang menyerah dengan ilmu pengetahuan.

Atas karunia serta kemudahan yang Engkau berikan akhirnya tugas akhir

ini dapat terselesaikan. Shalawat dan salam selalu ku limpahkan kepada

Nabi Muhammad SAW.

Ku persembahkan tugas akhir ini untuk orang-orang tercinta dan tersayang

atas kasihnya yang berlimpah Ayahanda Darman dan Ibunda Zuryetti

A.md, kep

Tiada cinta yang paling suci selain kasih sayang ayahanda dan ibundaku

tercinta. Setulus hatimu bunda, searif arahanmu ayah. Doamu hadirkan

keridhaan untukku, Petuahmu tuntunkan jalanku, Pelukmu berkahi hidupku,

diantara perjuangan dan tetesan doa malammu. Dan sebaith doa telah

merangkul diriku, Menuju hari depan yang cerah.

My beloved brother Drg. Gerry Dartha Anggriawan, Ryan Dartha

Adlar, S.si, dan Randa Dartha Nugraha (alm)

Terima kasih tiada tara atas segala support, motivasi dan doanya yang

selalu mengiringiku dalam menyelesaikan tugas akhir ini. Ku berdoa agar

kita bisa menjadi anak yang sukses yang bisa membanggakan papa dan mama

tercinta. Aminnnnnnn.....



Dosen pembimbing Bapak Firdaus, MT dan Ibu Derisma, MT
Terima kasih kepada bapak Firdaus, MT atas tenaga, pikiran, bimbingan,
nasihat serta selalu memberikan semangat dan support tiada tara dan
meluangkan banyak waktu untuk membimbing sehingga skripsi ini selesai,
dan Terima kasih juga kepada ibu Derisma, MT atas saran serta
masukan dan juga bimbingannya dalam menjalani skripsi ini. Tanpa bantuan
dan bimbingan Bapak dan Ibu mungkin skripsi ini tidak bisa diselesaikan.

Terima kasih banyak kepada Bapak Firdaus, MT dan Derisma, MT

Teman teman seperjuangan Sistem Komputer 010

"Tanpamu teman aku tak pernah berarti, tanpamu teman aku bukan siapa-
siapa yang takkan jadi apa-apa", buat saudara sekaligus sahabatku selama
masa kuliah ini.

Terima kasih kepada

Teman-teman SK010 atas kebersamaannya selama ini, canda, tawa,
bahagia, sedih, dimarahin, dapat nilai jelek, saling memberi dan saling

berbagi. Semua nya akan kita kenang sampai kelak...

Terima kasih kepada Hasbi, Surya, Ane, Armen, Taufik, Al, Ebing,

Arif, Rizky, Momoy, Rio, Ojid, Fikral Ef, Afwan, Rendy P

, Deby, Ridwan, Rahma, Dila, Shinta, Nisak, Sindi, DJ, Dede, Deni,

Gilang, dan semua nya yang tidak dapat disebutkan satu per satu.....

Untuk teman-teman yang lagi nyusun skripsinya Semoga diberikan kemudahan dan dilancarkan semuanya, sehingga sk010 cepat sarjana semuanya.. Aminnnnnnnn..

Terima kasihh karena sudah menjadi bagian dari sk010 yang super gokil hahahaha... semoga kelak dapat berkumpul lagi...

dan terkhusus Terima kasih kepada MY Dear



Makasih ya syg, semua berkat dorongan dan bantuan kamu sehingga aku bisa sampai mendapatkan gelar ini. Dengan segala pengorbanan kamu yang tak akan kulupakan. Yang telah memberikan solusi dan saran ketika aku jatuh. Selalu mendengarkan keluh kesah aku disaatku sedih. Menemani hari-hariku dengan perhatian dan kasih sayang yang begitu besar kepadaku terima kasih untuk segalanya . terima kasih atas canda dan tawamu bagiku semua tak bernilai dan tak tergantikan begitu indah untuk dikenang begitu sempurna untuk diabadikan cinta,kasih sayang dan pengorbananmu, dan limpahkan doa-doa yang tak henti terucap semua karna karena sayang,cinta dan keihlasanmu. Semoga kamu Cepat mendapatkan gelar ini yah ,doa rndy selalu ada untukmu ..

Untuk Pejoang Siskom 11 dan lainnya terutama Amid sama kakak , tetap semangat bikin T A nya, jangan menyerah. Jika kita selalu berusaha dan diiringi dengan doa, Insya Allah akan membuahkan hasil. Terimakasih juga buat pihak-pihak yang terlibat dalam memebantu penyelesaian Tugas Akhir ini, makasih, reslabers dll. Untuk warga SM&L, teruslah berkarya hingga kita dikenal dunia.



KATA PENGANTAR

Segala puji dan syukur penulis ucapkan atas kehadiran Allah SWT yang telah memberikan rahmat dan karunia-Nya serta shalawat dan salam atas junjungan Nabi Muhammad SAW, sehingga memberikan kemudahan kepada penulis untuk dapat menyelesaikan Tugas Akhir ini yang berjudul “Robot *Mobile* Pengikut Manusia Berbasis Mini PC ”.

Selama penulisan Tugas Akhir ini, penulis banyak mendapat bantuan dan dukungan dari berbagai pihak, baik berupa bantuan moril maupun materil. Untuk itu, pada kesempatan kali ini penulis ingin mengucapkan terima kasih sedalam-dalamnya kepada :

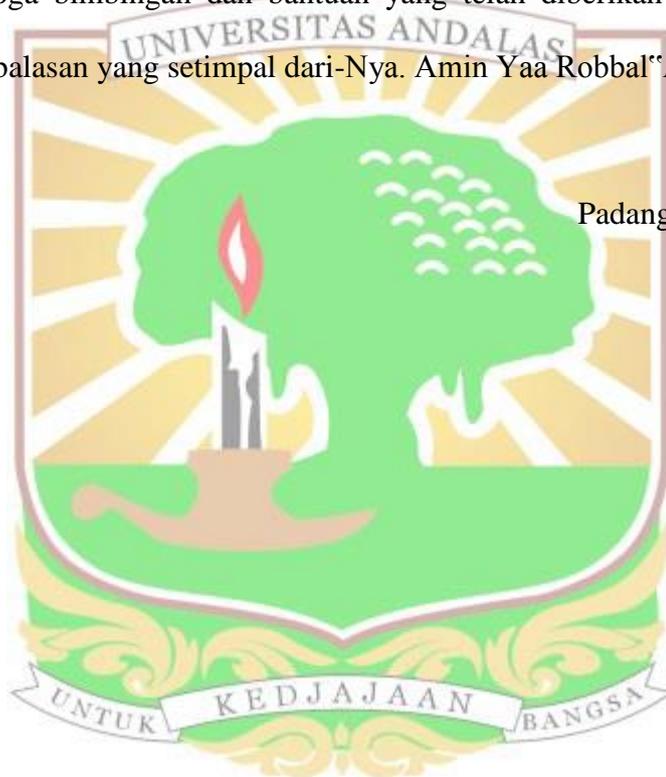
1. Bapak Dodon Yendri, M.Kom selaku Ketua Jurusan Sistem Komputer Universitas Andalas.
2. Bapak Firdaus, M.T selaku Pembimbing I yang telah membimbing penulis dalam pengerjaan Tugas Akhir ini dan banyak memberikan masukan serta saran kepada penulis.
3. Ibu Derisma, M.T selaku Pembimbing II yang telah banyak memberikan masukan dan saran kepada penulis dalam pengerjaan Tugas Akhir ini.
4. Kedua orang tua penulis yang telah memberikan dukungan dan semangat kepada penulis sehingga dapat menyelesaikan Tugas Akhir ini.
5. Kepada teman-teman yang telah membantu dan memberikan semangat dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa dalam penulisan Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu penulis memohon maaf apabila ada kekurangan maupun kesalahan dalam penyajiannya, serta mengharapkan kritikan dan saran yang bermanfaat untuk dapat mengembangkan dan meningkatkan mutu Tugas Akhir ini.

Akhir kata, penulis hanya dapat berserah diri kepada Allah SWT dan berharap semoga bimbingan dan bantuan yang telah diberikan kepada penulis mendapatkan balasan yang setimpal dari-Nya. Amin Yaa Robbal'Alamin.

Padang, Januari 2016

Penulis



**RANCANG BANGUN *MOBILE* ROBOT PENGIKUT OBJEK
BERDASARKAN WARNA DAN BENTUK
MENGUNAKAN METODE *TEMPLATE MATCHING* BERBASIS MINI
PC**

Rendy dartha Nugraha¹, Firdaus, MT², Derisma, MT³

¹ Mahasiswa Sistem Komputer Fakultas Teknologi Informasi Universitas Andalas

² Dosen Teknik Telekomunikasi Politeknik Negeri Padang

³ Dosen Sistem Komputer Fakultas Teknologi Informasi Universitas Andalas

ABSTRAK

Penerapan teknologi di bidang robotika telah banyak dimanfaatkan oleh kalangan industri di dunia kerja dalam rangka meningkatkan efisiensi dan efektivitas berbagai aktivitas kerja manusia. Sebuah *mobile* robot dibuat agar dapat mengikuti pergerakan objek berupa manusia berdasarkan warna menggunakan *webcam*. *Mobile* robot yang dibuat dapat melakukan proses pengolahan citra dari warna objek yang ditangkap oleh *webcam*, sehingga dapat mengenali warna objek yang diikuti. Proses pengolahan citra akan di proses di dalam Raspberry Pi sebagai pusat kontrol, yang akan menggerakkan motor pada *mobile* robot. Ada beberapa proses pengolahan citra untuk mengikuti pergerakan objek, yaitu: mencari nilai HSV masing-masing warna pada kondisi pencahayaan yang berbeda, menganalisa warna objek yang akan diikuti dengan membandingkan nilai HSV yang didapat dengan nilai HSV pada proses sebelumnya, dan mendeteksi warna lingkaran pada objek yang tujuan dengan menggunakan metode *template matching*. *Mobile* robot akan mengikuti pergerakan objek yang telah ditangkap oleh *webcam*, Dari penelitian yang dilakukan, didapatkan nilai HSV yang tepat untuk semua keadaan pencahayaan untuk masing-masing warna yaitu: nilai H (*Hue*) untuk warna, merah: 0-66, nilai S (*Saturation*): 64-169, dan nilai V (*Value*): 75-124. Dari 30 kali pengujian sistem yang dilakukan pada semua kondisi pencahayaan, diperoleh 7 kali kegagalan dengan persentase kegagalan sebesar 28,6% dan 29 kali keberhasilan dengan persentase keberhasilan sebesar 71,4 %.

Kata kunci: HSV, Robot, Raspberry, Webcam, Template Matching

THE DESIGN OF MOBILE ROBOT FOLLOWER OBJECTS BASED ON COLOR AND SHAPE USING TEMPLATE MATCHING METHOD BASED MINI PC

Rendy Dharta Nugraha¹, Firdaus, MT², Derisma, MT³

*1 Undergraduate Student, Computer System Major, Information Technology
Faculty, Andalas University*

2 Lecturer, Electrical Engineering, Padang State Polytechnic

*3 Lecturer, Computer System, Information Technology Faculty, Andalas
University*

ABSTRACT

Application of robotics has been used extensively by the industry in the world of work in order to improve the efficiency and effectiveness of the various activities of human labor. A mobile robot is set up to follow the movement of the object in the form of humans by color using a webcam. Mobile robot is made to perform image processing on the color of the object captured by the webcam, so it can recognize the color of objects that followed. Image processing will be processed in the Raspberry Pi as the control center, which will drive the motor in a mobile robot. There are some image processing to follow the movement of objects, namely: seeking the HSV values of each color in different lighting conditions, analyzing the color of objects that will be followed by comparing the HSV values obtained with the HSV values in the previous process, and detect the color of the circle on the object the destination by using template matching. Mobile robot will follow the movement of objects that have been captured by a webcam, From the research conducted, it was found that exact HSV values for all state lighting for each color, namely: the value of H (Hue) for color, red: 0-66, the value of S (Saturation): 64-169, and the value of V (value): 75-124. Of the 30 times the system testing performed on all lighting conditions, obtained 7 times failure with failure percentage was 28.6%, and 29 times the success with a success percentage of 71.4%.

Key Word: *HSV, Robot, Raspberry, Webcam, Template Matching*

DAFTAR ISI

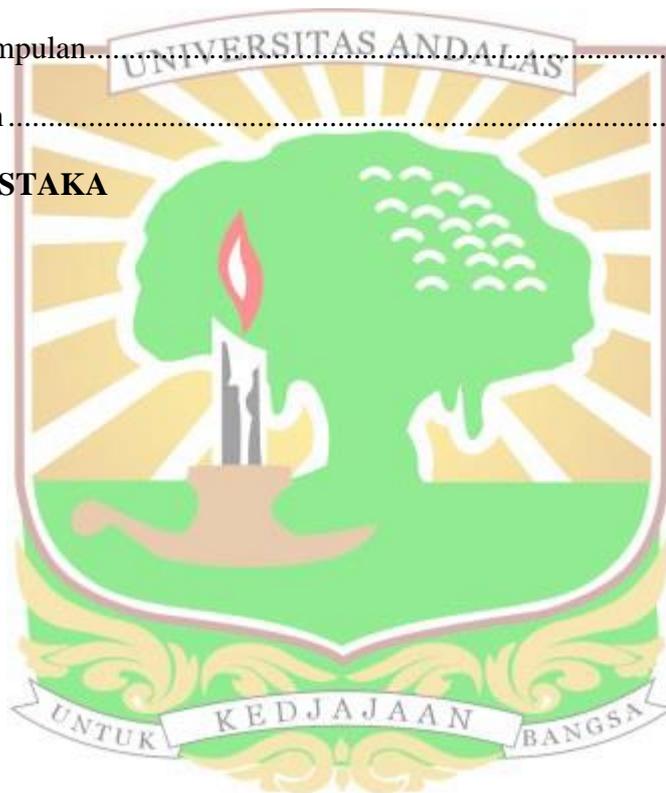
KATA PENGANTAR	iv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xx
DAFTAR TABEL	xxii
DAFTAR LAMPIRAN	xxv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	5
2.1 <i>Web Camera</i>	5
2.1.1 Definisi <i>Webcam</i>	5
2.1.2 Bagian dan kelengkapan <i>Webcam</i>	7
2.2 Robot	9
2.2.1 <i>Robot line follower</i>	13
2.2.2 Robot Pendeteksi.....	14
2.2.3 Komponen Robot	15
2.3 Motor DC	16
2.4 Raspberry Pi	19
2.5 <i>Python</i>	22
2.6 Pengolahan citra	24
2.6.1 Definisi Citra Digital.....	24

2.6.2	Tipe-tipe citra	25
2.7	Warna	27
2.8	Model warna HSV	29
2.9	Metode <i>Template Matching</i>	31
BAB III METODOLOGI PENELITIAN		33
3.1	Metode Penelitian	33
3.2	Rancangan Penelitian	34
3.3	Perancangan Sistem	37
3.3.1	Blok Diagram Sistem	37
3.3.2	Skema rangkaian	39
3.3.3	Perancangan <i>Hardware mobile robot</i>	40
3.3.4	Perancangan <i>Software</i>	41
3.4	Alat Dan Bahan Penelitian	46
3.4.1	Alat Penelitian	46
3.4.2	Bahan Penelitian	47
BAB IV HASIL DAN ANALISA		48
4.1	Proses	48
4.1.1	Implementasi <i>Hardware</i>	48
4.1.2	Implementasi <i>Software</i>	49
4.2	Implementasi Pencarian <i>Template</i>	49
4.2.1	Pemotongan citra (<i>cropping</i>)	49
4.2.2	Proses Konversi ke <i>grayscale</i>	50
4.2.3	Proses <i>Thresholding</i>	51
4.2.4	Proses Pendeteksian warna	51
4.3	Pengujian <i>Hardware</i>	52

4.4	Pengujian dan Analisa Sistem	53
4.4.1	Pengujian <i>webcam</i> dalam mendeteksi objek.....	53
4.4.2	Pengujian Motor <i>Shield L298</i> terhadap kecepatan motor	55
4.5	Implementasi Alat	63
4.5.1	Pengujian dan Analisa Sistem.....	63
4.5.2	Pengujian mencari nilai HSV.....	64
BAB V KESIMPULAN DAN SARAN		71
5.1	Kesimpulan.....	71
5.2	Saran.....	72

DAFTAR PUSTAKA

LAMPIRAN



DAFTAR GAMBAR

Gambar 2. 1 Contoh Webcam ^[4]	6
Gambar 2. 2 Elemen Utama Kamera Digital ^[4]	6
Gambar 2. 3 Kabel Data ^[5]	7
Gambar 2. 4 Tombol Snapshot ^[5]	8
Gambar 2. 5 Karakteristik Robot ^[7]	12
Gambar 2. 6 Robot Line Follower	13
Gambar 2. 7 Robot pendeteksi objek	15
Gambar 2. 8 Motor DC biasa beserta komponennya ^[8]	16
Gambar 2. 9 IC Driver Motor L298 ^[9]	17
Gambar 2. 10 Raspberry Pi ^[10]	20
Gambar 2. 11 Pengaksesan GPIO dengan Python ^[10]	21
Gambar 2. 12 BCM dengan board pada Raspberry ^[10]	22
Gambar 2. 13 Tampilan IDLE ^[11]	23
Gambar 2. 14 Tipe citra biner ^[12]	25
Gambar 2. 15 Tipe citra grayscale ^[12]	26
Gambar 2. 16 Citra warna indeks ^[12]	26
Gambar 2. 17 Spektrum Cahaya ^[13]	27
Gambar 2. 18 Pengelompokan Warna ^[13]	29
Gambar 2. 19 Skema Deteksi Model Warna HSV ^[14]	30
Gambar 2. 20 Model Warna HSV ^[13]	31
Gambar 3. 1 Rancangan Penelitian	34
Gambar 3. 2 Proses pengolahan citra	36
Gambar 3. 3 Blok Proses	38
Gambar 3. 4 Perancangan Rangkaian Komponen	40
Gambar 3. 5 Perancang Hardware (a) bagian atas, (b) bagian bawah	40
Gambar 3. 6 Flowchart Perancangan Software Secara Keseluruhan	Error!
Bookmark not defined.	
Gambar 3. 7 Flowchart mencari nilai HSV	41
Gambar 3. 8 Flowchart Analisa Warna Baju	44

Gambar 3. 9 Flowchart Identifikasi Warna.....	Error! Bookmark not defined.
Gambar 3. 10 Flowchart Robot Mobil Pengikut Manusia	45
Gambar 4. 1 Komponen setelah dirangkai	48
Gambar 4. 2 Cropping Citra.....	50
Gambar 4. 3 Citra grayscale.....	51
Gambar 4. 4 Citra Threshold.....	51
Gambar 4. 5 Area pendeteksian warna	52
Gambar 4. 6 Mobile robot.....	63
Gambar 4. 7 Mobile Robot	64
Gambar 4. 8 Mencari nilai HSV	65



DAFTAR TABEL

Tabel 2. 1 Fungsi Pin-Pin Pada IC L298 ^[9]	18
Tabel 4. 1 Pengujian sudut webcam.....	53
Tabel 4. 2 Pengujian Motor Shield L298P.....	55
Tabel 4. 3 Pengujian Kontrol Maju.....	56
Tabel 4. 4 Pengujian Kontrol Belok Kanan	57
Tabel 4. 5 Pengujian Kontrol Belok Kiri	57
Tabel 4. 6 Pengujian Kontrol Gerak Maju.....	59
Tabel 4. 7 Pengujian Kontrol Gerak Belok Kanan	60
Tabel 4. 8 Pengujian Kontrol Gerak Belok Kiri	62
Tabel 4. 9 Pengujian Respon webcam pada ruangan intensitas cahaya terang	66
Tabel 4. 10 Pengujian Respon webcam pada ruangan dengan intensitas cahaya Redup	69



DAFTAR ISTILAH

<i>Vision System</i>	: Sistem Visual Buatan
<i>Digital Image Processing</i>	: Pengolahan Citra Digital
<i>Perangkat Output</i>	: Perangkat yang berfungsi menyampaikan informasi
<i>Browsing internet</i>	: Mencari informasi yang sering digunakan di internet
<i>Array</i>	: Variable yang memiliki tipe data yang sama
<i>Monocrom</i>	: Citra biner
<i>Grayscale</i>	: Citra keabuan
<i>Thresholding citra</i>	: Proses mengubah citra derajat keabuan menjadi hitam putih atau biner
<i>Interfaces sistem</i>	: Mekanisme komunikasi antara manusia dengan sistem
<i>Realtime</i>	: Waktu pengoperasian perangkat
<i>Software</i>	: Perangkat lunak
<i>Hardware</i>	: Perangkat Keras
<i>image processing</i>	: Pengolahan Citra
HSV	: <i>Hue, Saturation, and Value</i>
CMOS	: Complementary Metal Oxide Semiconductor
CCD	: Charge Coupled Device
SBC	: Single Board Circuit
PC	: Personal Computer
SoC	: System On Chip
RAM	: Random Access Memory



SDHC : Secure Digital High Capacity

RCA : Radio Corporation of America

HDMI : High Definition Multimedia Interface

USB : Universal Serial Bus

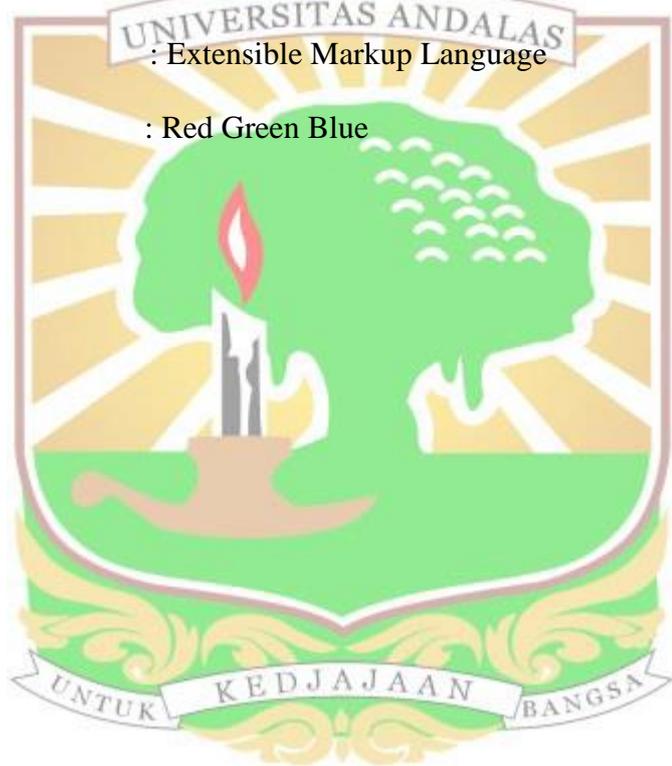
GPIO : General Purpose Input/Output

OpenCV : Open Computer Vision

API : Application Programming Interface

XML : Extensible Markup Language

RGB : Red Green Blue



DAFTAR LAMPIRAN

Lampiran 1 : *Source Code* Program



BAB I

PENDAHULUAN

1.1 Latar Belakang

Mata merupakan indera yang sangat penting dalam kehidupan sehari-hari. Dengan mata makhluk hidup dapat mengetahui benda apa yang ada di depannya berdasarkan warna dan bentuk benda tersebut. Dengan mata makhluk hidup dapat memperkirakan jaraknya ke suatu benda tersebut. Sistem visual buatan atau *vision system (computer vision)* adalah suatu sistem yang mempunyai kemampuan untuk menganalisis objek secara visual, setelah data objek yang bersangkutan dimasukkan dalam bentuk citra (*image*)^[1]. Disiplin ilmu yang melahirkan teknik-teknik untuk pengolahan citra ini dinamakan pengolahan citra digital (*digital image processing*)^[2].

Salah satu jenis robot yang banyak dikembangkan pada saat ini adalah *mobile robot*. Robot ini memiliki aktuator/penggerak berupa roda, sehingga robot dapat melakukan perpindahan posisi dari satu titik ke titik yang lain dengan menggunakan sensor sebagai indera penglihatannya.

Robot Pengikut manusia merupakan salah satu implementasi dari pengolahan citra digital robot harus dapat mengenali objek. Cara yang paling sederhana untuk mengenali objek adalah dengan menentukan warna objek . Permasalahan dalam robot pengikut manusia sangatlah kompleks, meliputi *human detection, tracking*, navigasi, pengendalian pergerakan, dan pengendalian posisi^[3]. Beberapa penelitian tentang permasalahan robot pengikut manusia telah

dilakukan sebelumnya. Dalam penelitian yang dilakukan S.Thrun yang berjudul “Implementasi Sistem Pendeteksian Target Pada Robot Pengikut Manusia“ Robot disini menggunakan metode data *map environment* untuk melakukan *tracking* dengan *webcam* sebagai input. Metode yang digunakan kurang cocok untuk kasus robot yang beroperasi pada tempat yang sering berubah-ubah, sehingga terlihat kelemahan dalam proses pencocokan *template* dikarenakan sering melakukan perubahan tempat.

Berdasarkan latar belakang tersebut, maka penulis ingin membuat sebuah rancangan *mobile* robot dengan judul “**Rancang Bangun Mobile Robot Pengikut Manusia Berdasarkan Warna Dan Objek Berbasis Mini PC**”. Dalam tugas akhir ini, Robot yang dirancang sebuah *mobile* robot yang terdiri dari penggabungan dari robot *line follower* dan robot pendeteksi, Sehingga proses pencocokan *template* yang dilakukan akan lebih akurat dengan objek yang diikuti beroperasi pada tempat yang sering berubah-ubah.

1.2 Rumusan Masalah

Adapun permasalahan yang dibahas dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Bagaimana merancang sebuah *mobile* robot yang dapat bergerak secara otomatis mengikuti pergerakan manusia berdasarkan warna dan objek.
2. Bagaimana membuat *mobile* robot agar dapat menganalisa warna baju dan bentuk objek dari pergerakan manusia sampai ke tujuan dengan memanfaatkan *webcam* sebagai inputnya.

3. Bagaimana melakukan pengolahan citra agar dapat robot mengikuti pergerakan manusia berdasarkan warna dan objek .

1.3 Batasan Masalah

Agar perancangan yang dibahas dalam Tugas Akhir ini tidak terlalu luas dan menyimpang, maka dibuat batasan-batasan sebagai berikut:

1. *Mobile Robot* ini hanya dapat mengikuti pergerakan manusia dengan objek warna merah.
2. Pengujian dilakukan pada tempat dengan kondisi pencahayaan yang berbeda, yaitu kondisi cahaya terang dan redup.
3. Implementasi dilakukan bebas dari halangan
4. Sistem yang dibuat *prototype*

1.4 Tujuan Penelitian

1. *Mobile Robot* dapat bergerak secara otomatis mengikuti pergerakan manusia.
2. *Mobile Robot* dapat menganalisa objek warna merah dari pergerakan manusia sampai tujuan dengan memanfaatkan *webcam* sebagai inputnya.
3. Dapat melakukan pengolahan citra, Sehingga robot dapat mengikuti pergerakan manusia yang diikuti.
4. Mempelajari prinsip kerja robot pengikut manusia



1.5 Sistematika Penulisan

Secara garis besar penulisan tugas akhir ini akan dibagi menjadi beberapa bab sebagai berikut:

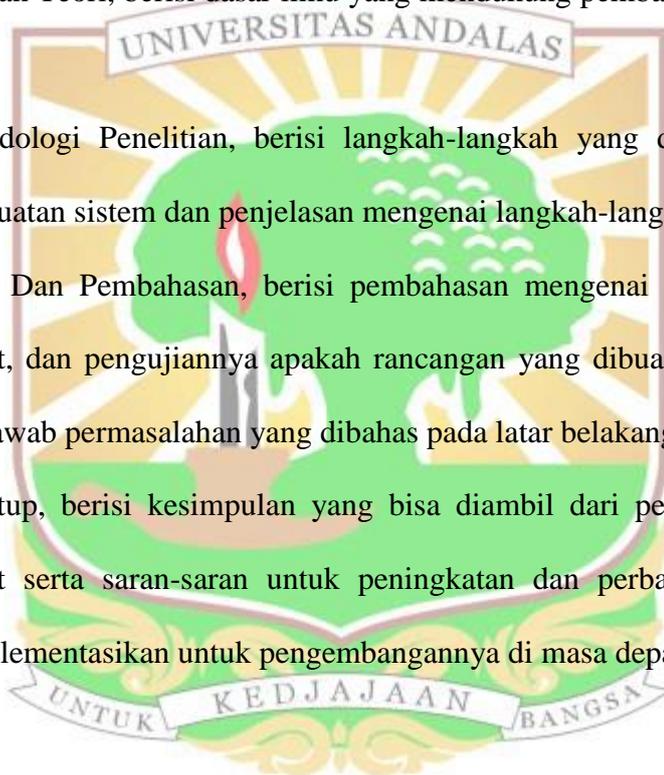
Bab I Pendahuluan, berisi permasalahan yang menjadi latar belakang penulisan tugas akhir ini, rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penulisan.

Bab II Landasan Teori, berisi dasar ilmu yang mendukung pembahasan penelitian ini.

Bab III Metodologi Penelitian, berisi langkah-langkah yang ditempuh dalam pembuatan sistem dan penjelasan mengenai langkah-langkah tersebut.

Bab IV Hasil Dan Pembahasan, berisi pembahasan mengenai rancangan yang dibuat, dan pengujiannya apakah rancangan yang dibuat sudah berhasil menjawab permasalahan yang dibahas pada latar belakang.

Bab V Penutup, berisi kesimpulan yang bisa diambil dari perancangan yang dibuat serta saran-saran untuk peningkatan dan perbaikan yang bisa diimplementasikan untuk pengembangannya di masa depan.



BAB II

LANDASAN TEORI

2.1 *Web Camera*

2.1.1 *Definisi Webcam*

Webcam^[4] adalah kamera video sederhana berukuran relatif kecil yang dihubungkan ke komputer melalui port USB. Sering digunakan untuk konferensi video jarak jauh atau sebagai kamera pemantau. Definisi yang lain tentang *webcam* adalah sebuah periferan berupa kamera sebagai pengambil citra atau gambar dan mikropon (optional) sebagai pengambil suara atau audio yang dikendalikan oleh sebuah komputer atau oleh jaringan komputer. Gambar yang diambil oleh *Webcam* ditampilkan ke layar monitor, karena dikendalikan oleh komputer maka ada *interface* atau *port* yang digunakan untuk menghubungkan *webcam* dengan komputer atau jaringan.

Sebuah *webcam* sederhana yang terlihat pada Gambar 2.1 terhubung dengan komputer dengan media kabel *USB* atau *parallel port*. Agar *webcam* dapat digunakan oleh komputer, dibutuhkan suatu perangkat lunak. Perangkat lunak itu akan menangkap tiap *frame* pada waktu tertentu, biasanya disebut dengan *frame rate*. *File* yang disimpan biasanya berformat *avi* (video), tapi dapat diubah dalam bentuk *jpeg* (*image*) atau yang lainnya agar tidak terlalu banyak memakan tempat pada media penyimpanan.

Berdasarkan sifatnya, *webcam* dapat dibagi menjadi beberapa jenis, yaitu:

- a. *Webcam Statis*

Webcam statis adalah *webcam* yang mengamati objek pada posisi tetap.

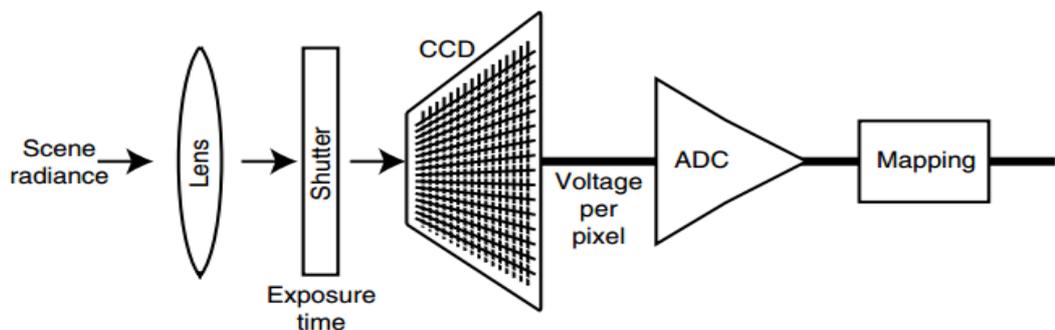
b. *Webcam* Dinamis

Webcam dinamis adalah *webcam* yang mampu mengamati objek yang bergerak dan bisa dikendalikan pergerakannya dari jarak jauh.



Gambar 2. 1 Contoh Webcam^[4]

Secara umum terlihat pada Gambar 2.2 cara kerja dari kamera digital.



Gambar 2. 2 Elemen Utama Kamera Digital^[4]

Lensa menangkap gambar objek, lalu diteruskan ke bagian panel penangkap gambar yang disebut dengan sensor CCD, pada CCD ini terdapat jutaan titik sensor yang dikenal dengan piksel. Tegangan tiap piksel yang berupa sinyal analog akan diukur dan diubah ke bentuk digital oleh ADC. Gambar yang ditangkap tadi akan diproses menjadi data digital berupa file format gambar, serta melakukan proses kompresi sesuai format gambar yang dipilih. Blok *mapping* berfungsi untuk memperbesar skala CCD dan untuk meningkatkan detail dari citra.

2.1.2 Bagian dan kelengkapan Webcam

Webcam terdiri dari sebuah lensa, sensor gambar (*image sensor*) dan sirkuit elektronik pendukung. Sensor gambar dapat berupa CMOS dan CCD. *Webcam* dilengkapi dengan peralatan tambahan^[5], yaitu :

1. Kabel data, merupakan kabel penghubung antara webcam dengan komputer atau pheripheral lainnya.



Gambar 2.3 Kabel Data^[5]

2. Tombol *Snapshot*, adalah tombol untuk pengambilan gambar bergerak dan menyimpan dalam bentuk foto.



Gambar 2. 4 Tombol Snapshot^[5]

3. Pengatur Focus, digunakan untuk mengatur lensa agar gambar yang ditampilkan jelas.
4. Lensa kamera, digunakan untuk mengambil gambar atau video.
5. *Motion sensing* – *web camera* akan mengambil gambar ketika kamera mendeteksi gerakan.
6. *Image archiving* – pengguna dapat membuat sebuah *archive* yang menyimpan semua gambar dari *webcam* atau hanya gambar-gambar tertentu saat interval pre-set.
7. *Video messaging* – beberapa program messaging mendukung fitur ini.
8. *Advanced connections* – menyambungkan perangkat *home theater* ke kamera Web dengan kabel maupun nirkabel.
9. *Automotion* – kamera robotik yang memungkinkan pengambilan gambar secara pan atau tilt dan setting program pengambilan frame berdasarkan posisi kamera.
10. *Streaming media* – aplikasi profesional, setup web camera dapat menggunakan kompresi MPEG4 untuk mendapatkan streaming audio dan video yang sesungguhnya.

11. *Custom coding* – mengimport kode komputer pengguna untuk memberitahu *web camera* apa yang harus dilakukan (misalnya *automatically refresh*).
12. *AutoCam* – memungkinkan pengguna membuat web page untuk kameranya secara gratis di server perusahaan pembuat Kamera Web.

2.2 Robot

Kata robot diambil dari kata yang berasal dari kata robota, yang mempunyai arti pekerja. Robot pertama kali dikembangkan oleh *Computer Aided Manufacturing-International (CAM-1)*, “ Robot adalah peralatan yang mampu melakukan fungsi-fungsi yang biasa dilakukan oleh manusia, atau peralatan yang mampu bekerja dengan intelegensi yang mirip dengan manusia”^[6].

Definisi kedua, dikembangkan oleh *Robotics Institute of America (RIA)*, perkumpulan pembuat robot yang lebih menitikberatkan terhadap kemampuan nyata yang dimiliki oleh robot terhadap kemiripannya dengan manusia.

Menurut definisi dari kamus Meriam-Webster, robot adalah mesin yang terlihat seperti manusia dan melakukan berbagai tindakan yang kompleks dari manusia seperti berjalan atau berbicara, atau suatu peralatan yang bekerja secara otomatis. Robot biasanya diprogram untuk melakukan pekerjaan berulang kali dan memiliki mekanisme yang dipandu oleh kontrol otomatis. Sedangkan robotika adalah cabang teknologi yang berkaitan dengan desain, konstruksi, operasi, dan aplikasi dari robot.

Robotika merupakan cabang ilmu pengetahuan yang mempelajari tentang robot. Cabang ilmu tersebut mencakup desain mesin robot, elektronika, pengontrolan, pemrograman komputer, kecerdasan buatan, dan lain sebagainya.^[6] Terdapat pendapat lain mengenai definisi robot, seperti diutarakan oleh Robotic Institute of America sebuah institusi robot yang terdapat 19 di Universitas *Carnegie Mellon* di kota Pittsburgh, negara bagian Pensilvania, Amerika Serikat, pada tahun 1979 mendefinisikan secara formal yang dimaksud dengan robot adalah "*manipulator* multi fungsi dan dapat diprogram ulang yang dirancang untuk menggerakkan material, alat, atau perangkat khusus melalui sejumlah gerakanterprogram untuk melakukan aktifitas tertentu".

Meskipun terdapat berbagai definisi robot, seperti telah disebutkan sebelumnya, tidak ditemukan definisi standar yang menjelaskan sebuah robot. Namun demikian, terdapat empat karakteristik dasar yang harus dimiliki oleh setiap robot modern.

Karakteristik dasar tersebut adalah sebagai berikut. ^[7]

1. Memiliki sensor.

Sensor merupakan peralatan yang berguna untuk mengukur ataupun merasakan sesuatu pada lingkungan di luar robot, layaknya indera pada makhluk hidup, dan member laporan hasilnya kepada robot. Dengan adanya sensor, robot bisa memiliki suatu pertimbangan dalam mengambil keputusan. Contoh dari sensor adalah sensor cahaya untuk mendeteksi adanya cahaya dan sensor temperatur untuk mengukur suhu.

2. Memiliki sistem kecerdasan (Kontrol).

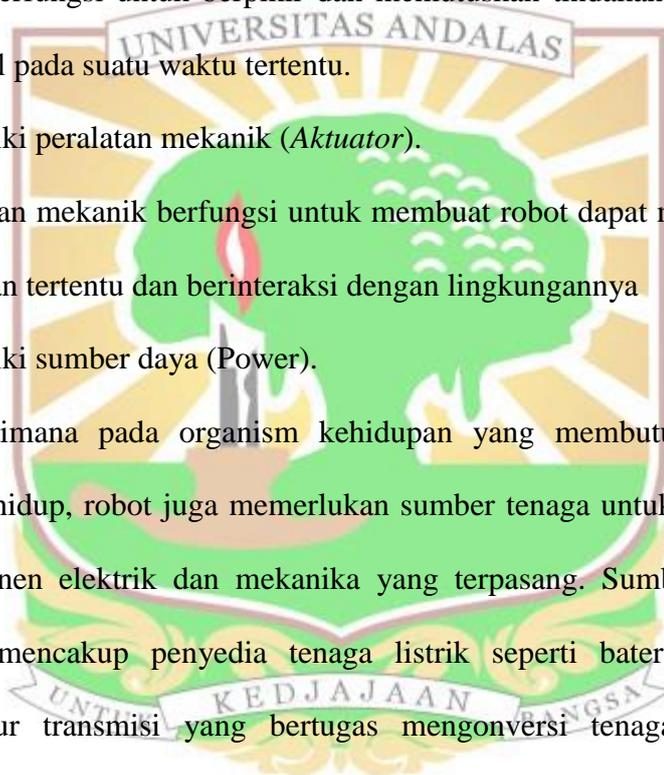
Sistem kecerdasan bekerja dengan memproses data masukan berupa keadaan ataupun kejadian yang sedang terjadi dari luar lingkungan. Selanjutnya sistem menghasilkan keluaran berupa instruksi ataupun keputusan pada robot untuk melakukan suatu tindakan tertentu. Sistem ini secara umum memiliki prinsip kerja seperti otak pada makhluk hidup, yang berfungsi untuk berpikir dan memutuskan tindakan apa yang perlu diambil pada suatu waktu tertentu.

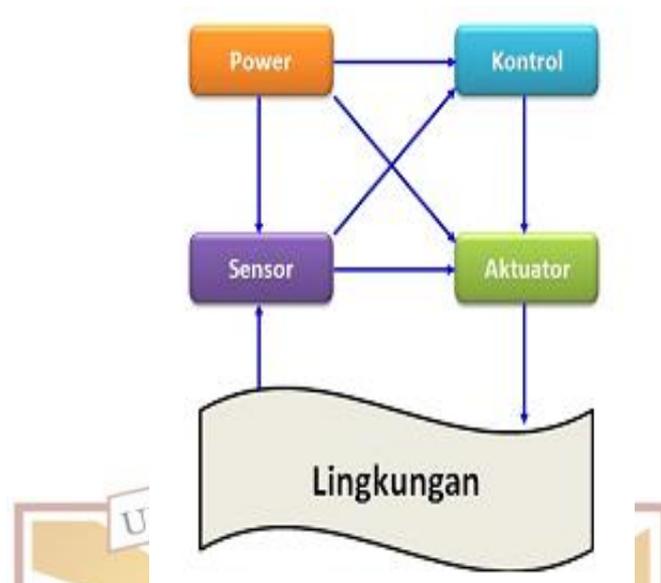
3. Memiliki peralatan mekanik (*Aktuator*).

Peralatan mekanik berfungsi untuk membuat robot dapat melakukan suatu tindakan tertentu dan berinteraksi dengan lingkungannya

4. Memiliki sumber daya (Power).

Sebagaimana pada organism kehidupan yang membutuhkan makanan untuk hidup, robot juga memerlukan sumber tenaga untuk menggerakkan komponen elektrik dan mekanika yang terpasang. Sumber energi pada robot mencakup penyedia tenaga listrik seperti baterai, dan system pengatur transmisi yang bertugas mengonversi tenaga listrik sesuai kebutuhan setiap komponen.





Gambar 2.5 Karakteristik Robot^[7]

Robot terdiri dari berbagai macam jenis, seperti ^[6]:

1. *Mobile robot*

Mobile Robot adalah konstruksi robot yang ciri khasnya mempunyai aktuator berupa roda untuk menggerakkan keseluruhan badan robot tersebut, sehingga robot tersebut dapat melakukan perpindahan posisi dari satu titik ke titik yang lain dengan bantuan navigasi dari sebuah sensor.

2. *Robot Manipulator*

Robot manipulator disebut juga dengan robot tangan. Robot ini hanya memiliki satu tangan, seperti tangan manusia yang fungsinya untuk memegang atau memindahkan barang.

3. *Robot Humanoid*

Robot humanoid merupakan robot yang memiliki kemampuan menyerupai manusia, baik fungsi maupun cara bertindak. Contoh robot ini adalah Ashimo yang dikembangkan oleh Honda.

4. Robot Berkaki

Robot ini memiliki kaki seperti hewan atau manusia, yang mampu melangkah, seperti robot serangga, robot laba-laba, dll.

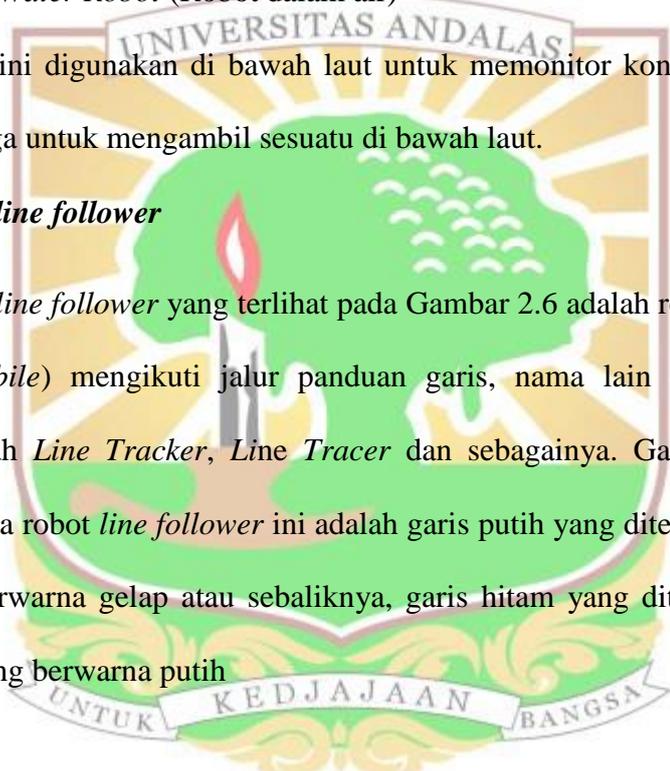
5. *Flying robot* merupakan robot yang mampu terbang, robot ini menyerupai pesawat model yang diprogram khusus untuk memonitor keadaan di tanah dari atas, dan juga untuk meneruskan komunikasi.

6. *Under Water Robot* (Robot dalam air)

Robot ini digunakan di bawah laut untuk memonitor kondisi bawah laut dan juga untuk mengambil sesuatu di bawah laut.

2.2.1 *Robot line follower*

Robot *line follower* yang terlihat pada Gambar 2.6 adalah robot yang dapat bergerak (*mobile*) mengikuti jalur panduan garis, nama lain dari robot *line follower* adalah *Line Tracker*, *Line Tracer* dan sebagainya. Garis pandu yang digunakan pada robot *line follower* ini adalah garis putih yang ditempatkan di atas permukaan berwarna gelap atau sebaliknya, garis hitam yang ditempatkan pada permukaan yang berwarna putih



Gambar 2. 6 Robot Line Follower^[6]

Dalam perancangan robot *line follower* diperlukan sebuah sensor garis yang terdiri dari pasangan LED . Prinsip kerja robot *line follower* mengikuti garis

adalah sensor *photodiode* akan mendeteksi adanya garis pada permukaan lintasan dengan membandingkan kondisi saat terkena cahaya dan tidak terkena cahaya. Informasi yang diterima sensor akan diteruskan ke prosesor untuk diolah dan akhirnya hasil olahan tersebut akan diteruskan ke penggerak atau motor, agar motor dapat menyesuaikan gerak tubuh robot sesuai garis yang dideteksinya^[2].

Pada robot *line follower* terdapat dua buah motor DC yang dapat digerakkan maju dan mundur dengan menggunakan driver motor, saat sensor mendeteksi adanya garis hitam di tengah dari wilayah sensor maka kedua motor akan berjalan searah jarum jam sehingga robot maju, dan ketika sensor mendeteksi adanya garis hitam dipinggir wilayah sensor maka salah satu motor akan berputar searah jarum jam dan yang satu berlawanan arah jarum jam sehingga robot akan bergerak ke arah kanan atau kiri.

2.2.2 Robot Pendeteksi

Robot pendeteksi yang terlihat pada Gambar 2.7 merupakan robot yang dapat mendeteksi suatu warna, garis, benda dan objek lainnya dengan menggunakan sebuah indera penglihat berupa kamera. Suatu objek berupa wajah manusia akan dideteksi oleh sebuah sensor yaitu kamera, sehingga data yang didapatkan tersebut di jadikan sebagai sebuah inputan untuk diproses, sehingga menghasilkan sebuah output yang dapat menggerakkan robot untuk melakukan sebuah tindakan khusus sesuai dengan yang telah diprogram sebelumnya oleh manusia^[2].



Gambar 2. 7 Robot pendeteksi objek^[7]

2.2.3 Komponen Robot

Komponen utama robot terdiri dari empat bagian, yaitu ^[7]:

1. *Manipulator*

Manipulator adalah bagian mekanik yang dapat difungsikan untuk memindah, mengangkat dan memanipulasi benda kerja.

2. *Sensor*

Sensor adalah komponen berbasis instrumentasi (pengukuran) yang berfungsi sebagai pemberi informasi tentang berbagai keadaan atau kedudukan dari bagian-bagian *manipulator*.

3. *Aktuator*

Aktuator adalah komponen mekanik yang digunakan untuk menghasilkan gerakan pada robot. Sumber gerakan utama pada robot untuk menggerakkan bagian-bagian tubuh dari robot berupa motor. *Aktuator* bisa menggunakan elektrik, hidrolis ataupun *pneumatik*.

3. Kontroler

Kontroler adalah rangkaian elektronik berbasis mikroprosesor yang berfungsi sebagai pengatur seluruh komponen dalam membentuk fungsi kerja.

2.3 Motor DC

Pada prinsipnya motor DC yang terlihat pada Gambar 2.8 memiliki dua bagian dasar :

1. Bagian yang tetap / stasioner disebut *stator*, *stator* ini menggunakan medan magnet, baik yang dibangkitkan dari sebuah koil (elektromagnet) ataupun magnet permanen.
2. Bagian yang berputar disebut rotor atau armature. *Rotor* ini berupa sebuah koil dimana arus listrik mengalir.

Jenis motor dibedakan berdasarkan pengaturan listrik dan konstruksi



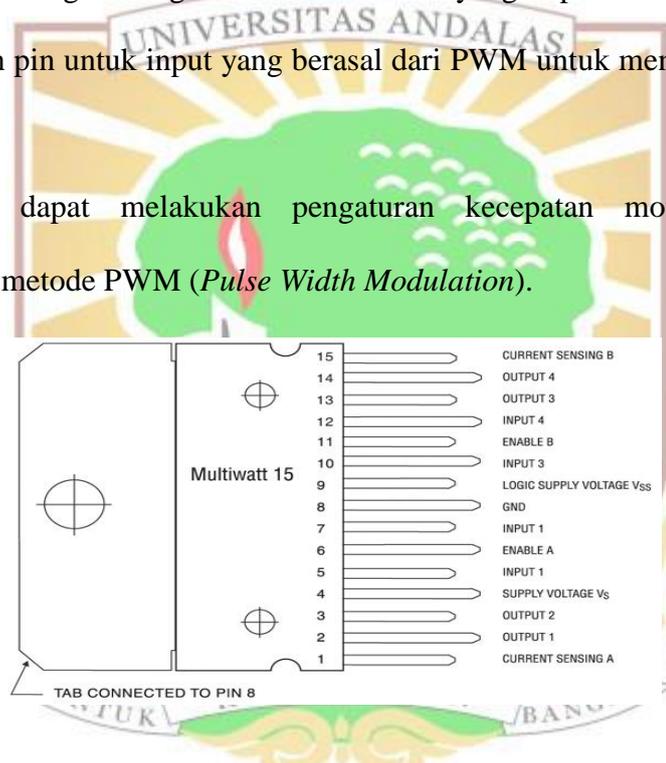
Gambar 2. 8 Motor DC biasa beserta komponennya^[8]

Motor DC bekerja bila pada kedua kaki stator diberikan tegangan atau beda potensial sehingga pada *rotor* terjadi perubahan energi listrik menjadi energi mekanik yang mengakibatkan terjadinya tolak menolak antara *rotor* dan *stator*, karena kedua bagian ini dipasang secara berdampingan dengan kutub yang berbeda. Dari tolak menolak yang terjadi pada kedua magnet yang ditimbulkan

oleh *stator* dan *rotor* maka terjadi suatu pergerakan yang mengakibatkan *rotor* akan berputar sesuai dengan pemberian tegangan pada kaki *stator* [8]..

Motor DC tidak dapat dikendalikan langsung oleh mikrokontroler, karena kebutuhan arus yang besar sedangkan keluaran arus dari mikrokontroler sangat kecil, maka digunakan sebuah *driver motor*. *Driver motor* merupakan alternatif yang dapat digunakan untuk menggerakkan motor DC. IC L298 yang terlihat pada Gambar 2.9 berfungsi sebagai *driver motor DC* yang dapat mengatur arah putar dan disediakan pin untuk input yang berasal dari PWM untuk mengatur kecepatan motor DC. [8]

Untuk dapat melakukan pengaturan kecepatan motor DC dapat menggunakan metode PWM (*Pulse Width Modulation*).



Gambar 2. 9 IC Driver Motor L298[9]

Metode PWM dapat digunakan untuk mengatur kecepatan motor dan untuk menghindari rangkaian mengkonsumsi daya berlebih. PWM dapat mengatur kecepatan motor karena tegangan yang diberikan dalam selang waktu tertentu saja. PWM ini dapat dibangkitkan melalui software. Lebar pulsa PWM

dinyatakan dalam *Duty Cycle*, misalnya duty cycle 10%, berarti lebar pulsa adalah 1/10 bagian dari satu perioda penuh. ^[9].

Driver motor IC L298 merupakan driver motor berbasis *H-Bridge*, driver motor ini bertegangan tinggi dan didesain untuk dapat menerima logic level TTL standar dan dapat mengendalikan output yang bersifat induktif contohnya: *relay*, solenoida, motor DC dan motor *stepper*. Driver motor mempunyai 2 input *enable* yang mendukung untuk mengaktifkan atau menonaktifkan sinyal input secara bebas. *Emitter* dari *transistor* untuk setiap brigdenya dihubungkan secara bersama-sama. Fungsi-fungsi pin IC L298 seperti yang terlihat pada Tabel 2.1 ^[9].

Tabel 2.1 Fungsi Pin-Pin Pada IC L298^[9]

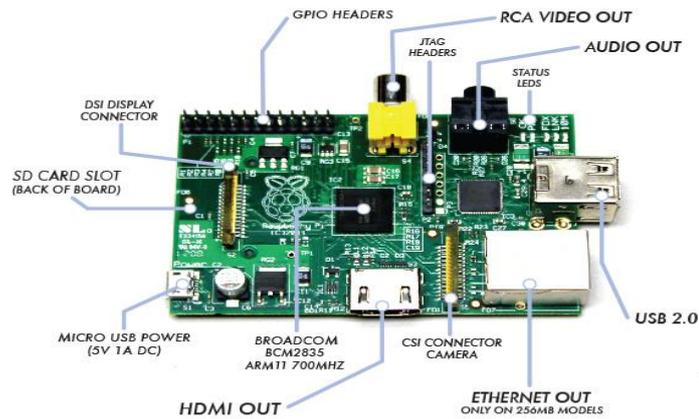
Nama Pin	Fungsi
<i>Sense A</i> dan <i>Sense B</i>	Diantara pin ini dengan <i>ground</i> terhubung resistor <i>sense</i> yang berfungsi untuk mengontrol arus
<i>Output 1</i> dan <i>output 2</i>	<i>Output</i> ke <i>bridge A</i> . Arus yang mengalir melalui beban dihubungkan antara dua pin ini dipantau di pin 1 (pin <i>Sense A</i>)
<i>Vs</i>	Tegangan <i>supply</i> . Sebuah kapasitor sebesar 100 nF dihubungkan ke pin ini yang kemudian dihubungkan ke <i>ground</i>
<i>Input 1</i> dan <i>input 2</i>	<i>Input</i> TTL untuk <i>bridge A</i>
<i>Enable A</i> dan <i>enable B</i>	<i>Input</i> TTL yang kompetibel. Jika diberi logika <i>low</i> maka akan menonaktifkan <i>bridge A</i> (<i>enable A</i>) dan

	bridge B (<i>enable B</i>)
GND	<i>Ground</i>
VSS	Tegangan <i>supply</i> untuk blok <i>logic</i> . Sebuah kapasitor 100 nF harus dihubungkan ke pin ini dengan <i>ground</i> .
<i>Input 3 dan input 4</i>	<i>Input TTL</i> untuk <i>bridge B</i>
<i>Output 3 dan output 4</i>	<i>Output</i> ke <i>bridge B</i> . Arus yang mengalir melalui beban dihubungkan antara dua pin ini dipantau di pin 15 (pin <i>Sense B</i>)

2.4 Raspberry Pi

Raspberry Pi adalah komputer papan tunggal *Single Board Circuit* (SBC) dimana ada penghubung komponen dan fungsi komputer serta elektronika ke dalam satu chip atau bisa disebut dengan *embeded system* yang terlihat pada Gambar 2.10^[10]. Raspberry Pi menyediakan socket RJ45 untuk terkoneksi ke internet dengan koneksi kabel, apabila berminat untuk menggunakan *WiFi*, dapat menggunakan *WiFi dogle* yang kompatibel dengan Rapsberry.

Raspberry Pi bisa digunakan untuk berbagai keperluan, seperti spreadsheet, game, bahkan bisa digunakan sebagai media player karena kemampuannya dalam memutar video *high definition*. Raspberry Pi menggunakan *System On a Chip* (SoC) dari *Broadcom BCM2835*, juga sudah termasuk prosesor ARM1176JZF-S 700 MHz, GPU VideoCore IV dan RAM sebesar 256 MB (untuk Rev.B). Tidak menggunakan *hard disk*, namun menggunakan *SD Card* untuk proses *booting* dan penyimpanan data jangka-panjang^[7].

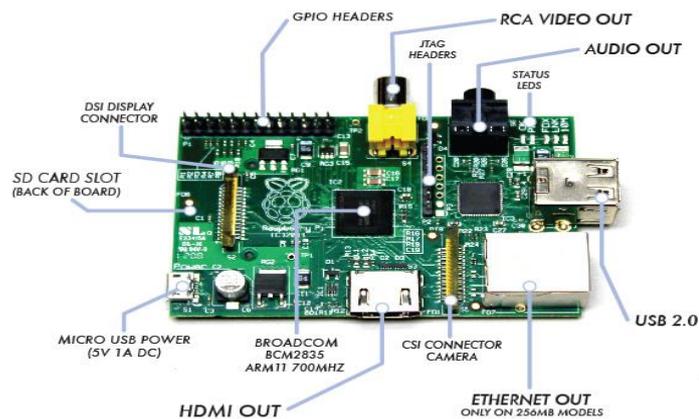


Gambar 2. 10 Raspberry Pi^[10]

Spesifikasi Raspberry PI ^[10]. :

- a. Prosesor ARM 11 (700 MHz ARM11 ARM1176JZF-S core).
- b. Sumber daya / power bisa menggunakan *port* mini usb dengan besar daya maximal 5 V / 1A
- c. *Storage* SDHC dan MicroSD + Adaptor minimal 2G class 4
- d. Satu *port* RCA untuk dihubungkan ke TV
- e. Satu *port* HDMI
- f. Dua *port* USB
- g. Satu *port* RJ45
- h. Mendukung General Purpose Input/Output (GPIO) pins , *Universal asynchronous receiver/transmitter* (UART)
- i. Satu *port* audio TRS connector | 3.5 mm
- j. Mendukung operation sistem :
- k. Raspberry “Wheezy memiliki bagian terdiri dari:

1. *Arch Linux ARM*
2. RISC OS
3. *Fedora remix*
4. FirefoxOS
5. Android Distro-distro linux for raspberry pi



Gambar 2.11 Pengaksesan GPIO dengan Python^[10]

Jumlah pin GPIO yang tersedia pada Raspberry Pi yaitu 26 buah. Untuk membaca atau menulis nilai dari perangkat eksternal yang terhubung dilakukan dengan menuliskan pin yang digunakan. Ada dua versi penamaan dari pin GPIO, yang pertama adalah BCM dan yang kedua BOARD. Sistem penamaan BCM berdasarkan pada channel yang digunakan oleh masing – masing pin sedangkan penamaan pada sistem BOARD mengacu pada penomoran secara fisik.

Untuk lebih jelas mengenai perbedaan antara penamaan BCM dengan board dapat dilihat pada Gambar 2.12

3.3V	1	2	5V
I2CO SDA	3	4	DNC
I2CO SCL	5	6	GROUND
GPIO4	7	8	UART TXD
DNC	9	10	UART RXD
GPIO 17	11	12	GPIO 18
GPIO 21	13	14	DNC
GPIO 22	15	16	GPIO 23
DNC	17	18	GPIO 24
SP10 MOSI	19	20	DNC
SP10 MISO	21	22	GPIO 25
SP10 SCLK	23	24	SP10 CE0 N
DNC	25	26	SP10 CE1 N

Gambar 2. 12 BCM dengan board pada Raspberry^[10]

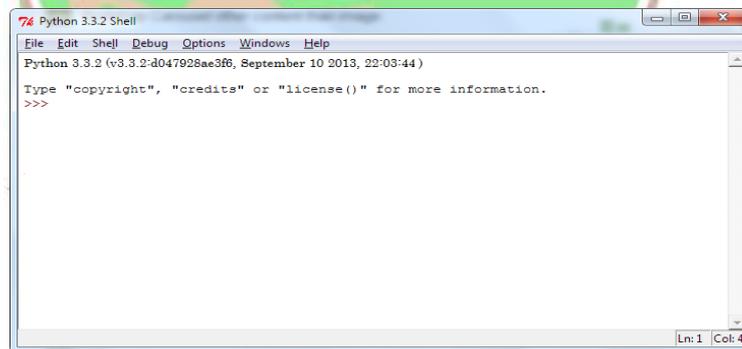
2.5 Python

Python merupakan bahasa pemrograman tingkat tinggi yang sangat fleksibel. *Python* sangat cocok digunakan untuk pembuatan program yang membutuhkan waktu singkat, karena bahasa pemrograman *Python* dilengkapi dengan modul dan paket yang mempermudah dalam pembuatan *software*. *Python* dikembangkan oleh *Guido Van Rossum* pada akhir 1989 dan dipublikasikan pada tahun 1991^[11].

Python dapat digunakan untuk berbagai keperluan pengembangan *software* dan dapat berjalan di berbagai macam sistem operasi karena sifatnya yang multiplatform. Saat ini kode *Python* dapat dijalankan pada sistem berbasis: Linux/Unix, Windows, Mac OS X, OS/2, Amiga, Palm, *Symbian* (untuk produk-produk Nokia). Beberapa fitur yang dimiliki *Python* adalah:

- a. Memiliki kepastakaan yang luas; dalam distribusi *Python* telah disediakan modul-modul siap pakai untuk berbagai keperluan.

- b. Memiliki tata bahasa yang jernih dan mudah dipelajari.
- c. Memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber.
- d. Berorientasi objek.
- e. Memiliki sistem pengelolaan memori otomatis (*garbage collection*, seperti java)
- f. Modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa *Python* maupun C/C++.
- g. Memiliki fasilitas pengumpulan sampah otomatis, seperti halnya pada bahasa pemrograman *Java*, *Python* memiliki fasilitas pengaturan penggunaan ingatan komputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatan komputer secara langsung



Gambar 2.13 Tampilan IDLE^[11]

Untuk menulis *script* program *Python* dapat menggunakan *word processing* seperti notepad pada windows atau vim pada linux dengan ketentuan

ekstensi file yang dibuat adalah “.py”. Namun pada dasarnya *Python* telah dilengkapi dengan IDE yang digunakan untuk penulisan script program IDLE.

2.6 Pengolahan citra

2.6.1 Definisi Citra Digital

Pengolahan citra atau (*image processing*) merupakan proses untuk memperbaiki kualitas citra, khususnya menggunakan komputer, menjadi citra yang kualitasnya lebih baik agar lebih mudah untuk diinterpretasikan oleh manusia ataupun mesin.

Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu^[12]

Umumnya citra digital berbentuk persegi panjang atau bujur sangkar (pada beberapa sistem pencitraan ada pula yang berbentuk segi enam) yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau piksel sehingga ukuran citra selalu bernilai bulat. Setiap titik memiliki koordinat sesuai posisinya dalam citra. Koordinat ini biasanya dinyatakan dalam bilangan bulat positif, yang dapat dimulai dari 0 atau 1 tergantung pada sistem

yang digunakan. Setiap titik juga memiliki nilai berupa angka digital yang merepresentasikan informasi yang diwakili oleh titik tersebut.

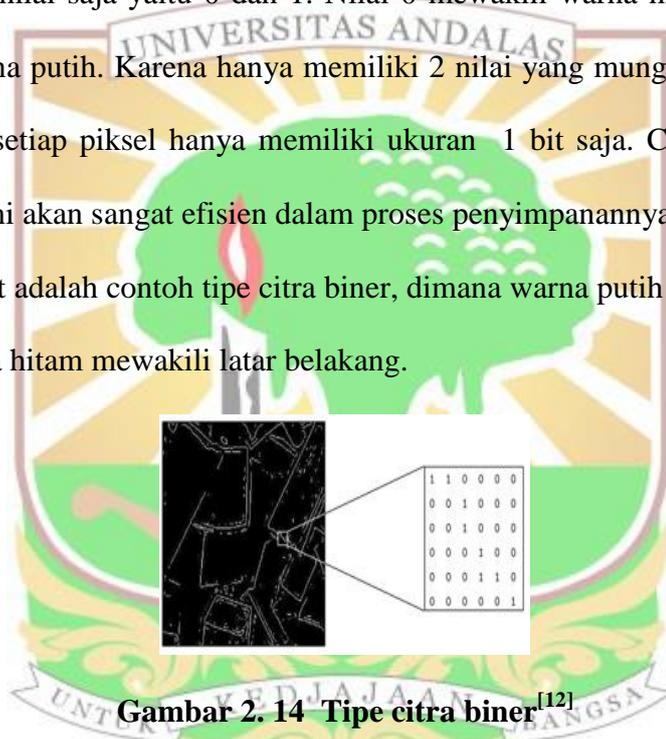
2.6.2 Tipe-tipe citra

Terdapat empat tipe dasar citra digital sebagai berikut^[12] :

1. Citra Digital Bertipe Biner

Pada citra digital dengan tipe biner, setiap piksel pada citra hanya memiliki dua nilai saja yaitu 0 dan 1. Nilai 0 mewakili warna hitam dan nilai 1 mewakili warna putih. Karena hanya memiliki 2 nilai yang mungkin untuk setiap piksel, maka setiap piksel hanya memiliki ukuran 1 bit saja. Citra dengan tipe biner seperti ini akan sangat efisien dalam proses penyimpanannya.

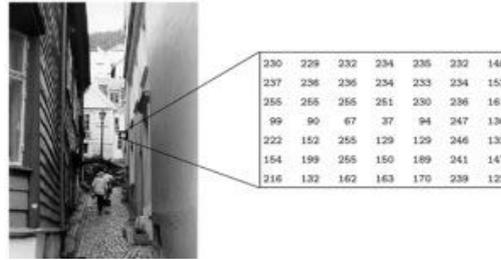
Berikut adalah contoh tipe citra biner, dimana warna putih mewakili piksel tepi dan warna hitam mewakili latar belakang.



Gambar 2. 14 Tipe citra biner^[12]

2. Citra Digital Bertipe *Grayscale*

Pada citra dengan tipe grayscale, setiap piksel mewakili derajat keabuan dengan nilai antara 0 (hitam) sampai 255 (putih). Pada jangkauan nilai 0 sampai 255 , ini berarti bahwa setiap piksel memiliki ukuran 8 bit atau 1 byte. Berikut adalah contoh citra bertipe grayscale.^[12]

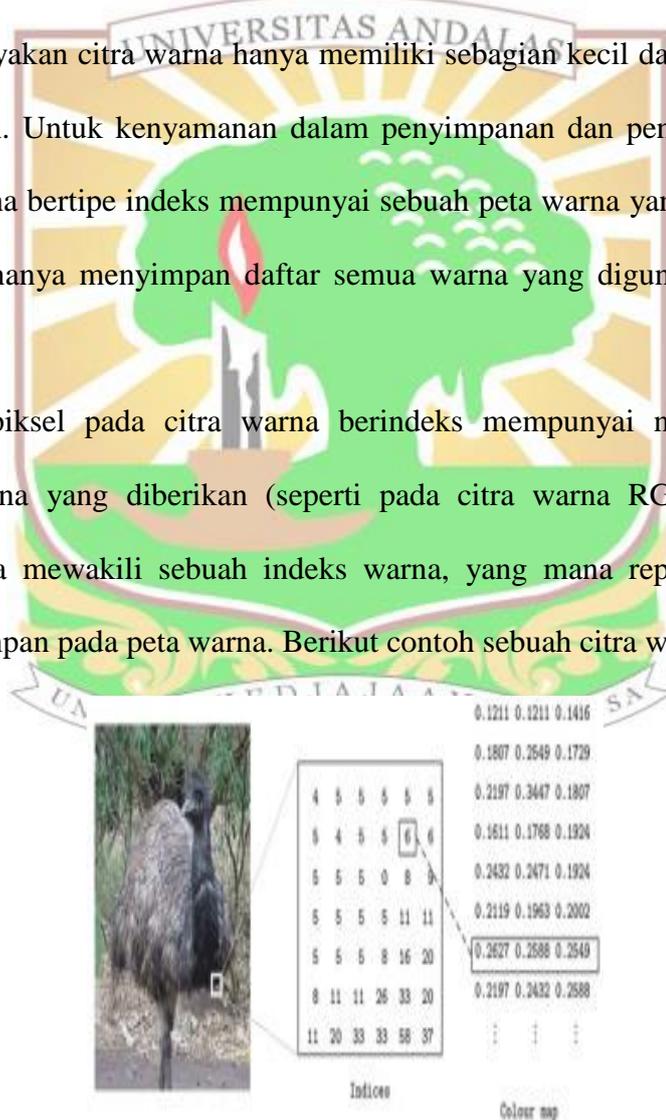


Gambar 2. 15 Tipe citra grayscale^[12]

3. Citra Digital Bertipe Warna Berindeks

Kebanyakan citra warna hanya memiliki sebagian kecil dari 16 juta warna yang mungkin. Untuk kenyamanan dalam penyimpanan dan penanganan berkas file, citra warna bertipe indeks mempunyai sebuah peta warna yang terkait indeks warna, yang hanya menyimpan daftar semua warna yang digunakan pada citra tersebut.

Setiap piksel pada citra warna berindeks mempunyai nilai yang tidak mewakili warna yang diberikan (seperti pada citra warna RGB), tetapi nilai tersebut hanya mewakili sebuah indeks warna, yang mana representasi warna tersebut tersimpan pada peta warna. Berikut contoh sebuah citra warna berindeks.

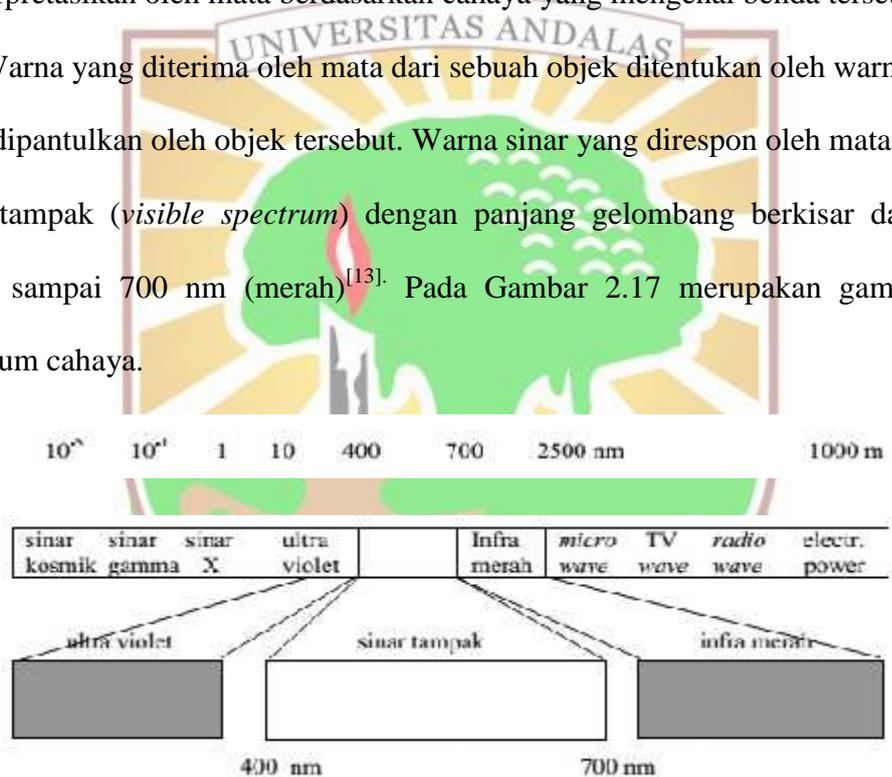


Gambar 2. 16 Citra warna indeks^[12]

2.7 Warna

Menurut Endang Widjajanti Laksono warna merupakan bagian dari cahaya yang diteruskan atau dipantulkan. Terdapat tiga unsur penting dari pengertian warna yaitu benda, mata dan unsur cahaya. Secara umum, warna didefinisikan sebagai unsur cahaya yang dipantulkan oleh sebuah benda dan selanjutnya diinterpretasikan oleh mata berdasarkan cahaya yang mengenai benda tersebut^[13].

Warna yang diterima oleh mata dari sebuah objek ditentukan oleh warna sinar yang dipantulkan oleh objek tersebut. Warna sinar yang direspon oleh mata adalah sinar tampak (*visible spectrum*) dengan panjang gelombang berkisar dari 400 (biru) sampai 700 nm (merah)^[13]. Pada Gambar 2.17 merupakan gambar 27 spectrum cahaya.



Gambar 2. 17 Spektrum Cahaya^[13]

Warna-warna yang diterima oleh mata manusia merupakan hasil kombinasi cahaya dengan panjang gelombang berbeda. Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah *red*, *green* dan *blue*^[13].

Pada tahun 1831, Brewster mengemukakan teori tentang pengelompokan warna. Teori Brewster membagi warna–warna yang ada di alam menjadi empat kelompok warna, yaitu^[13]:

1. Warna *Primer*

Warna *primer* adalah warna dasar yang tidak berasal dari campuran dari warna–warna lain, yang dikenal dengan RGB (*Red, Green, Blue*). Warna *primer* tersusun atas warna merah (*Red*), hijau (*Green*), dan biru (*Blue*).

2. Warna *Sekunder*

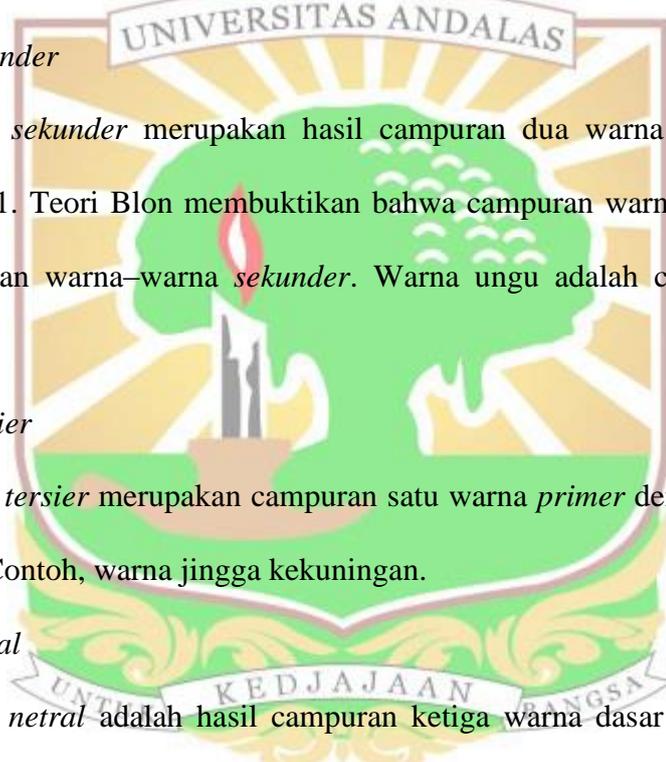
Warna *sekunder* merupakan hasil campuran dua warna *primer* dengan proporsi 1:1. Teori Blon membuktikan bahwa campuran warna–warna *primer* menghasilkan warna–warna *sekunder*. Warna ungu adalah campuran merah dan biru.

3. Warna *Tersier*

Warna *tersier* merupakan campuran satu warna *primer* dengan satu warna *sekunder*. Contoh, warna jingga kekuningan.

4. Warna *Netral*

Warna *netral* adalah hasil campuran ketiga warna dasar dalam proporsi 1:1:1. Pada Gambar 2.10 berikut adalah gambar pengelompokan warna secara umum.



Gambar 2. 18 Pengelompokan Warna^[13]

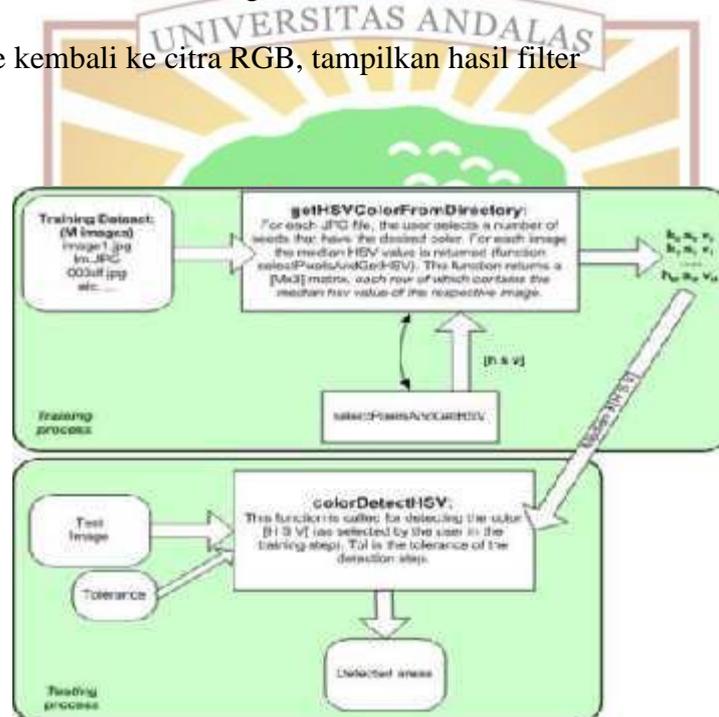
Pengolahan warna RGB pada posisi tertentu merupakan dasar dari pengolahan citra. RGB adalah singkatan dari *Red-Green-Blue*, yaitu tiga warna dasar merah (*Red*), hijau (*Green*), dan biru (*Blue*) yang dijadikan patokan warna secara universal (*primary colors*). Setiap komponen warna (*Red, Green, Blue*) memiliki matriks sendiri-sendiri. RGB (r, g, b) memiliki nilai masing-masing antara 0 hingga 255 yaitu pertama *Red*, kedua *Green* dan ketiga *Blue* dengan demikian masing-masing komponen ada 256 tingkat. Apabila dikombinasikan maka ada $256 \times 256 \times 256$ atau 16.777.216 kombinasi warna RGB yang dapat dibentuk.

2.8 Model warna HSV

Dalam proses pengenalan objek atau deteksi objek diperlukan suatu pemisahan bagian atau segmen tertentu dalam citra yang akurat. Proses pemisahan tersebut dikenal dengan proses segmentasi. Secara umum pendekatan segmentasi citra yang sering digunakan adalah melalui pendekatan intensitas, pendekatan warna dan pendekatan bentuk^[14].

Segmentasi warna objek merupakan proses segmentasi dengan pendekatan daerah yang bekerja dengan menganalisis warna dari tiap piksel pada citra dan membagi citra tersebut sesuai dengan fitur yang diinginkan. Secara garis besar, gambaran proses segmentasi dapat dilihat pada Gambar 2.19 dan berikut ini merupakan proses segmentasi menurut Giannakopoulos^[14]:

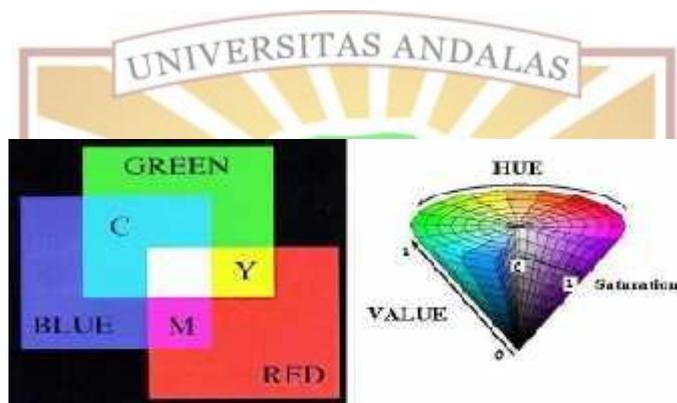
1. Tentukan citra RGB yang menjadi objek deteksi, nilai warna HSV yang menjadi acuan (hasil proses pelatihan data) dan nilai toleransi HSV yang digunakan.
2. Transpose citra RGB ke HSV
3. Lakukan filter warna pada citra berdasarkan nilai acuan (T) dan nilai toleransi (tol). Dengan x sebagai warna HSV pada piksel yang ada maka warna yang tidak termasuk dalam rentang $T - \text{tol} < x < T + \text{tol}$ diberi warna hitam.
4. Transpose kembali ke citra RGB, tampilkan hasil filter



Gambar 2. 19 Skema Deteksi Model Warna HSV^[14]

Model warna HSV merupakan model warna yang diturunkan dari model warna RGB, sehingga untuk mendapatkan warna HSV harus melakukan proses konversi warna dari RGB ke HSV. HSV merupakan salah satu cara untuk mendefinisikan warna yang didasarkan pada roda warna. *Hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning. *Hue* digunakan untuk membedakan

warna dan menentukan kemerahan (*redness*), kehijauan (*greenness*), kebiruan (*blueness*) dari cahaya. *Saturation* menyatakan tingkat kemurnian suatu warna yang menunjukkan pada radius roda warna sehingga menunjukkan proporsi antara gelap (pusat) untuk warna ke putih murni (di luar). *Value* menunjukkan nilai kecerahan. *Hue* memiliki nilai antara 0 hingga 360 (derajat), *Saturation* dan *Value* berkisar dari 0 hingga 100% [13]. Gambar model warna HSV, dapat dilihat pada Gambar 2.20



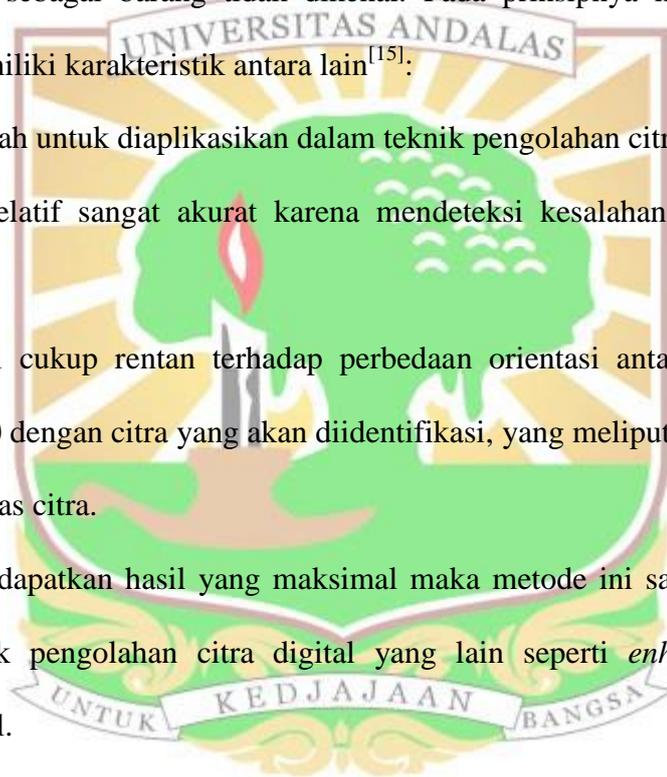
Gambar 2. 20 Model Warna HSV^[13]

2.9 Metode *Template Matching*

Template matching merupakan salah satu teknik dalam pengolahan citra digital yang berfungsi untuk mencocokkan tiap-tiap bagian dari suatu citra dengan citra yang menjadi *template* (acuan). Teknik ini banyak digunakan dalam bidang industri sebagai bagian dari *quality control*. Metode *template matching* termasuk salah satu metode terapan dari teknik konvolusi. Metode ini sering digunakan untuk mengidentifikasi citra karakter huruf, angka, sidik jari (*fingerprint*) dan aplikasi-aplikasi pencocokan citra lainnya^[14].

Prinsip metode ini adalah membandingkan antara citra *streaming* yang akan dikenali dengan citra sampel (*template*). Citra barang yang akan dikenali mempunyai tingkat kemiripan sendiri terhadap masing-masing citra sampel (*template*). Pengenalan dilakukan dengan melihat nilai tingkat kemiripan tertinggi dan nilai batas ambang pengenalan dari citra barang tersebut. Bila nilai tingkat kemiripan berada di bawah nilai batas ambang maka citra barang tersebut dikategorikan sebagai barang tidak dikenal. Pada prinsipnya metode *template matching* memiliki karakteristik antara lain^[15]:

1. Relatif mudah untuk diaplikasikan dalam teknik pengolahan citra digital.
2. Hasilnya relatif sangat akurat karena mendeteksi kesalahan hingga ukuran piksel.
3. Metode ini cukup rentan terhadap perbedaan orientasi antara citra sampel (*template*) dengan citra yang akan diidentifikasi, yang meliputi: ukuran, posisi dan kualitas citra.
4. Untuk mendapatkan hasil yang maksimal maka metode ini sangat tergantung pada teknik pengolahan citra digital yang lain seperti *enhancement*, *color filtering*, dll.



BAB III

METODOLOGI PENELITIAN

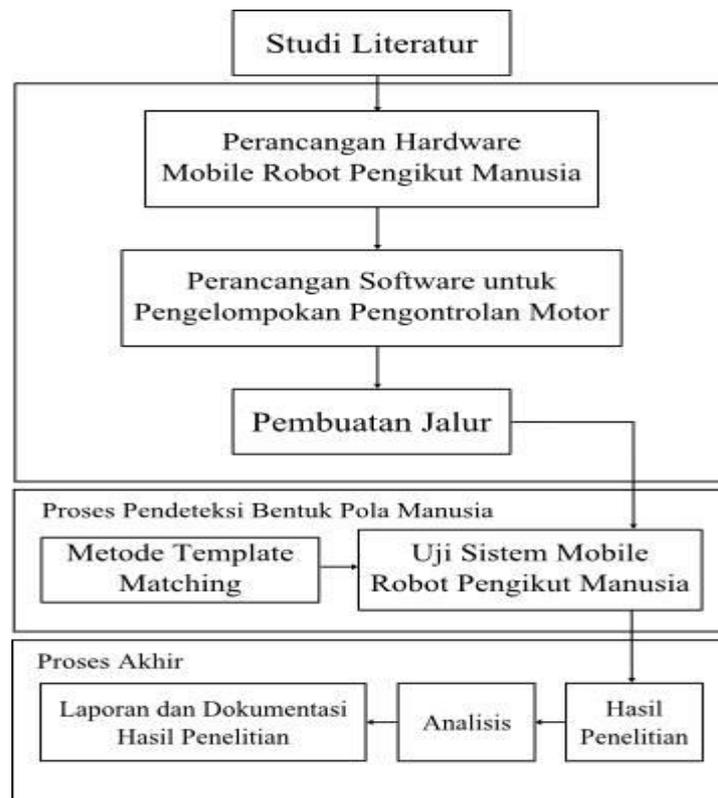
3.1 Metode Penelitian

Penelitian eksperimen (*experimental research*) adalah jenis penelitian yang digunakan untuk melihat hubungan sebab akibat. Penelitian eksperimental digunakan untuk mencari pengaruh perlakuan tertentu terhadap yang lain dalam kondisi yang terkendalikan. Penelitian eksperimental menggunakan suatu percobaan yang dirancang secara khusus guna membangkitkan data yang diperlukan untuk menjawab pertanyaan penelitian. Penelitian eksperimental dilakukan secara sistematis, logis, dan teliti didalam melakukan kontrol terhadap kondisi. Penelitian jenis ini bertujuan untuk mempelajari sesuatu dengan memvariasikan beberapa kondisi dan mengamati efek yang terjadi.

Penelitian ini ditunjang dengan studi literatur (*literatur research*) dengan membaca dan mempelajari literatur tentang perancangan pendeteksi mobil pada area parkir serta berbagai komponen yang dibutuhkan dalam perancangan untuk memperoleh informasi yang relevan dengan topik. Pada penelitian ini, mobile robot akan diuji dengan beberapa variabel yang mempengaruhi hasil mendeteksi pola bentuk objek yang ditangkap oleh *webcam*. Seperti: pengaruh pencahayaan.

3.2 Rancangan Penelitian

Dalam pembuatan tugas akhir ini, terdapat beberapa tahap penelitian seperti pada gambar 3.1



Gambar 3.1 Metode Penelitian

Berdasarkan Gambar 3.1, dapat dijelaskan tahapan-tahapan yang dilakukan untuk menyelesaikan penelitian ini, yaitu sebagai berikut :

1. Studi Literatur

Studi Literatur digunakan untuk mempelajari teori-teori yang berhubungan dengan perancangan *mobile* robot pengikut manusia. Pada studi literatur ini, materi yang akan dipelajari sebagai berikut :

- a. Cara kerja *webcam* pada *mobile* robot pengikut manusia sehingga mampu mengikuti pergerakan manusia yang diikuti.
- b. Cara kerja *webcam* *meng-capture* warna pada pakaian manusia yang diikuti.
- c. Proses pengolahan citra gambar yang di-*capture* oleh *webcam* pada Raspberry Pi
- d. Mempelajari ruang warna HSV untuk segmentasi warna yang mampu menoleransi perubahan intensitas cahaya untuk mendeteksi warna baju.
- e. Mempelajari metode *template matching* dengan membandingkan nilai HSV warna yang terdeteksi dengan nilai HSV warna yang tersimpan dalam variabel

2. Perancangan

Pada tahap perancangan ini, akan dilakukan pengerjaan baik dari segi *hardware* maupun *software* terhadap system yang akan dibangun. Adapun proses yang akan dilakukan pada tahap ini adalah :

a. Proses Perancangan *hardware*

Pada proses ini, penulis merancang *hardware*/mekanik untuk *mobile* robot pengikut manusia, sebagai berikut :

- i. Rangka robot terbuat dari akrilik
- ii. Pemasangan *webcam* pada bodi robot untuk melakukan proses *tracking* manusia yang akan diikuti. Dimana *webcam* tersebut akan dikontrol oleh 2 buah motor DC yang berada pada roda robot yang

nantinya akan melakukan pergerakan terhadap proses tracking ketika robot berjalan

b. Proses Perancangan *software*

Pada proses perancangan software, menggunakan *open CV*, dengan bahasa pemrograman python untuk pengolahan citra, dan pengontrolan motor dc menggunakan Raspberry Pi dari inputan *webcam*. Proses pengolahan citra yang terjadi di dalam Raspberry pi, dapat dilihat pada Gambar 3.2



Gambar 3. 2 Proses pengolahan citra

3. Proses Pencarian Warna

Dalam perancangan *mobile* robot pengikut manusia, proses pencarian warna yang dituju dilakukan setelah semua proses pada tahap perancangan selesai. Pada tahap ini, *webcam* akan meng-*capture* dan mendeteksi warna baju. Sehingga didapatkan nilai HSV yang didapat dari masing-masing warna tersebut disimpan dalam sebuah variabel. Setelah proses mencari nilai HSV selesai, maka robot diberi perintah untuk berjalan mengikuti pergerakan manusia yang diikuti.

Pada proses *tracking webcam* meng-*capture* warna baju, dan kemudian warna yang di capture tersebut akan diproses didalam Raspberry Pi dengan melewati beberapa proses pengolahan citra. Setelah proses pengolahan citra selesai, maka didapatkan nilai HSV dan jumlah piksel maksimal dari warna baju tersebut, kemudian Raspberry pi akan mengirimkan perintah untuk menggerakkan

motor dc, agar *mobile* robot berjalan mengikuti pergerakan manusia yang diikutinya.

4. *End Process*

Tahap ini merupakan tahap akhir dalam penelitian, yang mencakup 3 proses, yaitu:

a. Hasil Penelitian

Hasil dari penelitian adalah *mobile* robot pengikut manusia mengikuti pergerakan manusia yang diikutinya sesuai dengan warna baju yang digunakan.

b. Analisa hasil tugas akhir

Pada tahap ini, dilakukan analisa system penelitian dengan membandingkan teori-teori yang ada dengan mengubah variabel-variabel yang dapat mempengaruhi hasil dari kinerja sistem.

c. Laporan dan dokumentasi hasil tugas akhir

Penyusunan laporan dilakukan untuk memberikan penjelasan berkaitan dengan penelitian yang dilakukan dan dokumentasi dari penelitian.

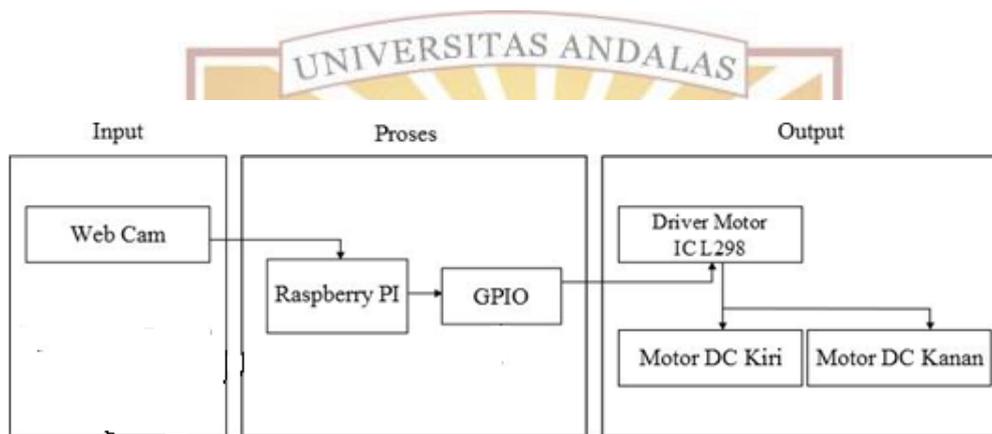
3.3 Perancangan Sistem

3.3.1 Blok Diagram Sistem

Dalam penulisan tugas akhir ini, blok sistem akan dibagi menjadi beberapa sub sistem, kemudian sub sistem tersebut dibagi lagi menjadi beberapa bagian sehingga lebih mudah dalam menentukan komponen yang digunakan. Dalam

perancangan *hardware mobile* robot pengikut manusia ini, terdapat 3 blok sistem yaitu: Input, proses dan output.

Blok sistem input, yaitu *webcam* sebagai sensor yang digunakan meng-capture warna baju pakaian. Input yang berasal dari *webcam* ini akan di proses pada Raspberry pi. Blok diagram perancangan sistem yang akan dibuat, dapat dilihat pada Gambar 3.3 berikut.



Gambar 3.3 Blok Proses

Perancangan sistem pada Gambar 3.3 terdiri dari :

1. *Webcam*

Webcam berfungsi untuk mendeteksi warna baju manusia yang akan diikuti. Inputan yang berasal dari *webcam* diolah di dalam Raspberry Pi, dengan menghubungkan kabel usb ke port usb yang ada pada Raspberry .

2. Raspberry Pi

Raspberry Pi merupakan otak yang akan memberi perintah melakukan proses pengolahan citra untuk mendeteksi warna baju - yang telah di-*capture* oleh *webcam*.

3. Motor *Shield* L298

Motor *shield* L298 berfungsi sebagai pengendali motor DC. Motor *shield* dapat mengendalikan 2 buah motor DC, yang dapat bekerja secara maksimal apabila menggunakan sumber arus eksternal sebagai daya, seperti dari baterai 12 volt.

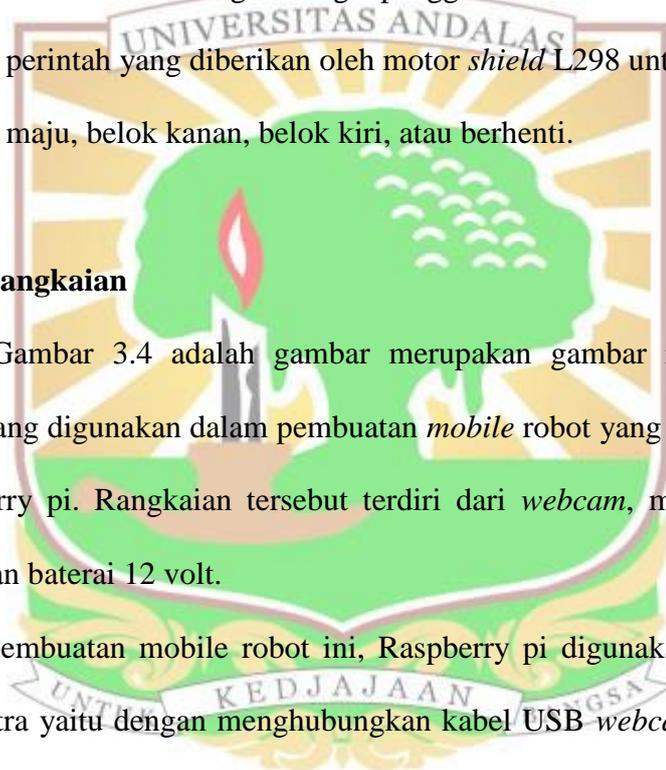
4. Motor DC

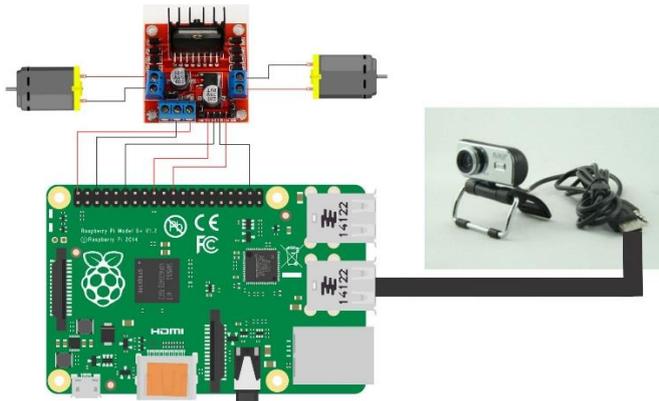
Motor DC berfungsi sebagai penggerak robot sesuai dengan perintah yang diberikan oleh motor *shield* L298 untuk bergerak maju, belok kanan, belok kiri, atau berhenti.

3.3.2 Skema rangkaian

Pada Gambar 3.4 adalah gambar merupakan gambar rangkain secara keseluruhan yang digunakan dalam pembuatan *mobile* robot yang dihubungkan ke board Raspberry pi. Rangkaian tersebut terdiri dari *webcam*, motor Dc, motor *shield* L298 dan baterai 12 volt.

Pada pembuatan *mobile* robot ini, Raspberry pi digunakan untuk proses pengolahan citra yaitu dengan menghubungkan kabel USB *webcam* ke port USB pada Raspberry pi. *Power* (sumber) berasal dari *power* bank yang memiliki tegangan 5 volt.

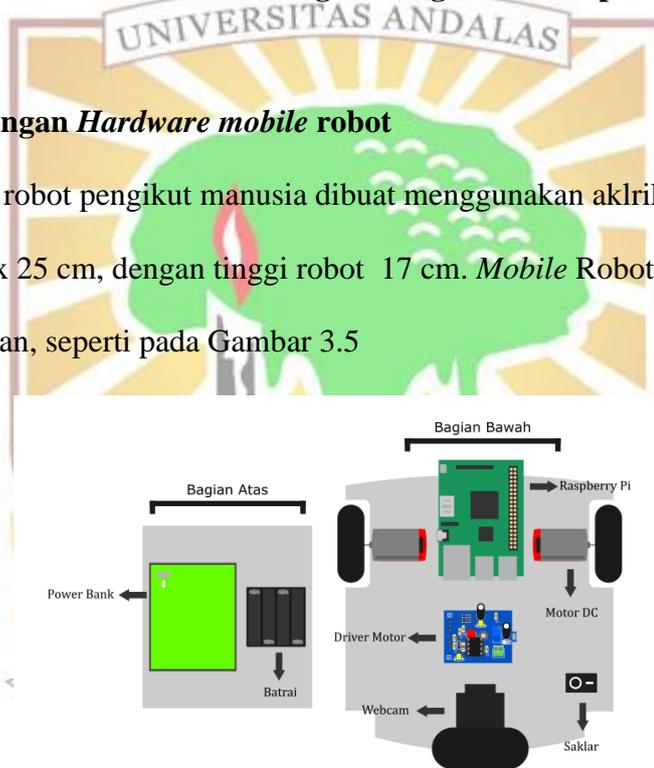




Gambar 3.4 Perancangan Rangkaian Komponen

3.3.3. Perancangan *Hardware mobile robot*

Mobile robot pengikut manusia dibuat menggunakan akrilik. Dengan ukuran 16 cm x 25 cm, dengan tinggi robot 17 cm. *Mobile Robot* didesain menjadi 2 bagian, seperti pada Gambar 3.5

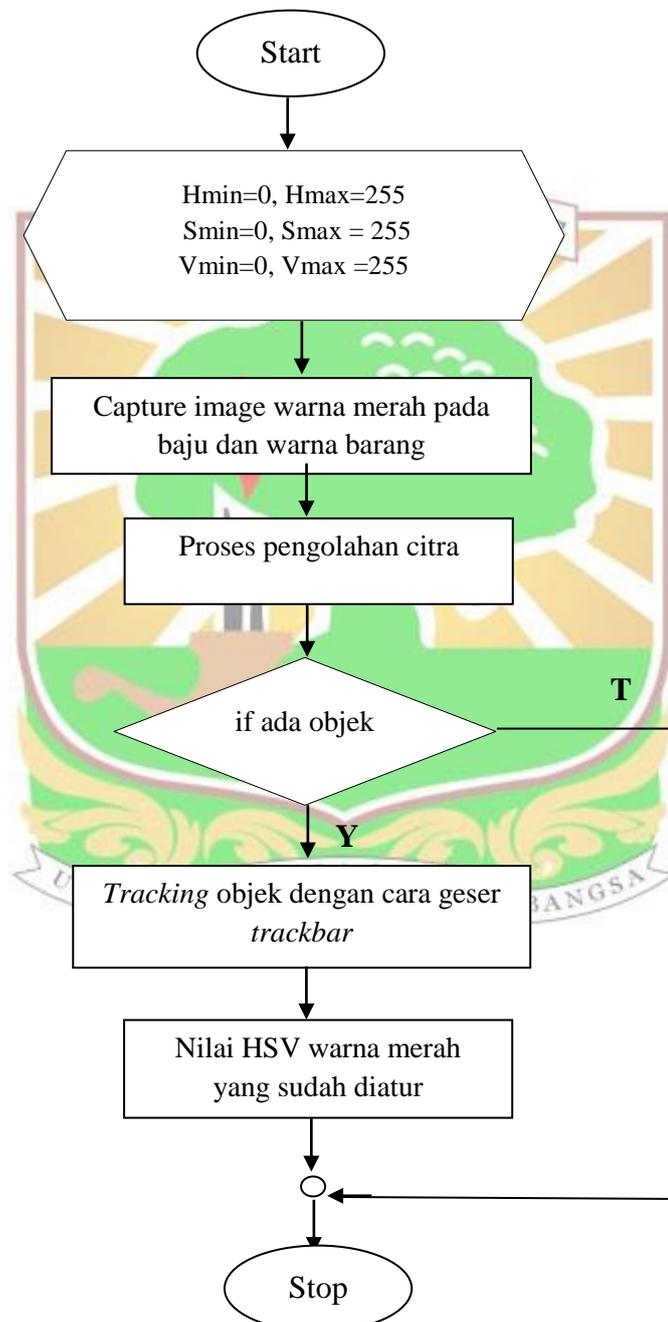


Gambar 3.5 Perancang Hardware (a) bagian atas, (b) bagian bawah

Perancangan mekanik *mobile robot* pada gambar (Gambar 3.5 (a)) digunakan untuk memasang motor dc, baterai 9,6 volt untuk sumber motor *shield*, dan Raspberry pi. Sedangkan pada bagian atas (Gambar 3.5 (b)) digunakan untuk meletakkan power bank yang berguna sebagai sumber untuk Raspberry Pi dan *webcam*.

3.3.4 Perancangan *Software*

Perancangan *Software* pada tugas akhir ini menggunakan open CV dengan bahas pemograman python. Pada Gambar 3.7 merupakan *flowchart* mencari nilai HSV.



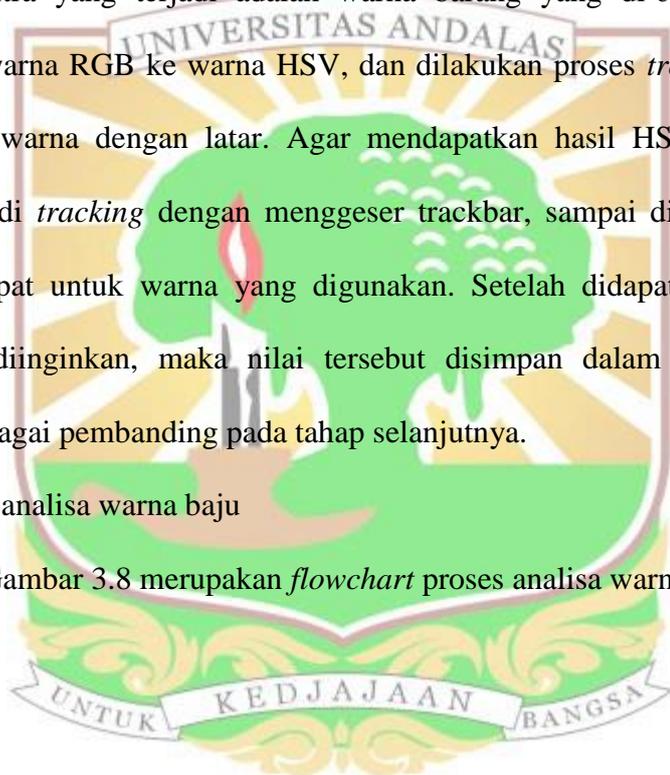
Gambar 3. 6 Flowchart mencari nilai HSV

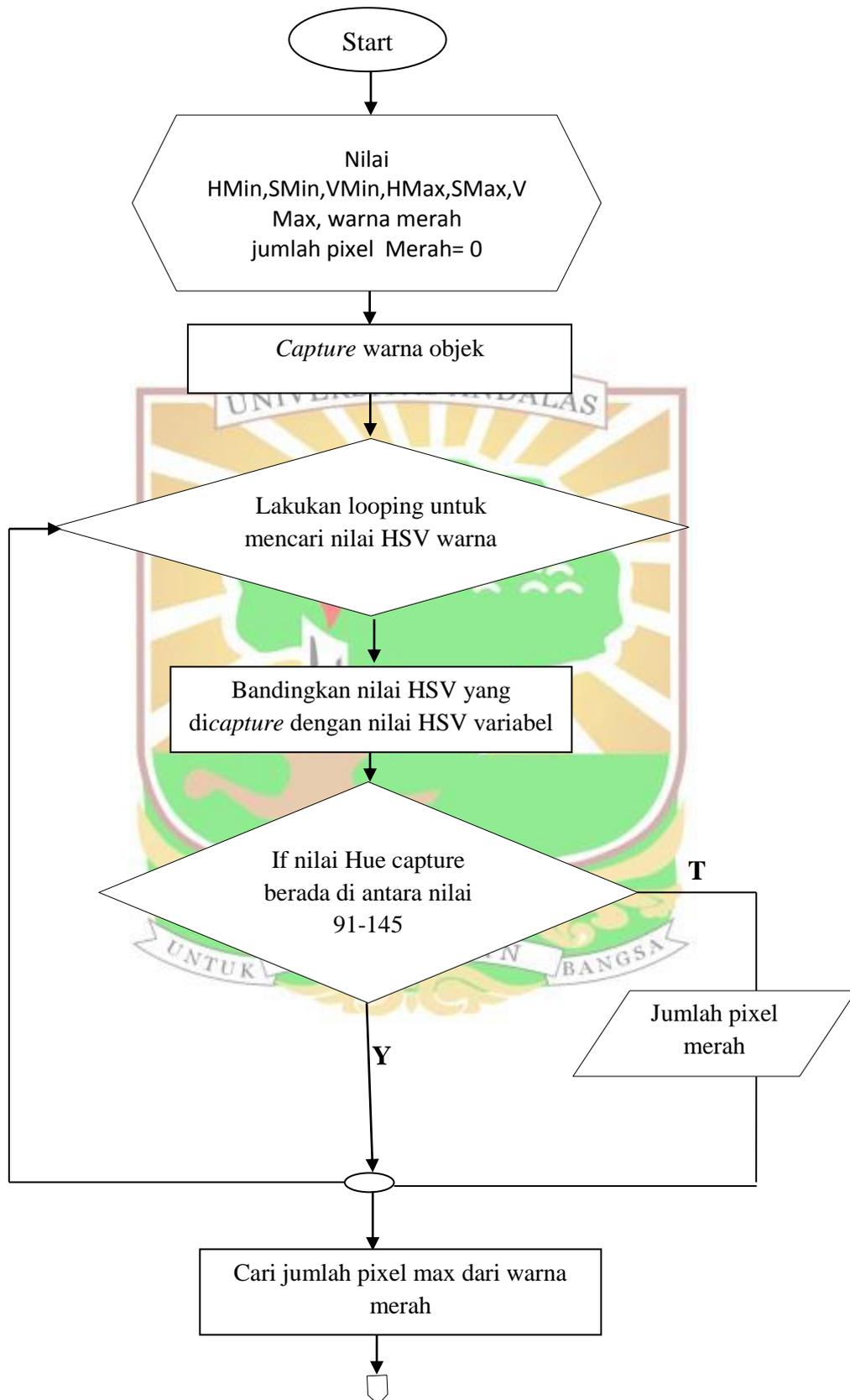
Pada *flowchart* gambar 3.7 proses mencari nilai HSV, nilai HSV pada trackbar di-setting terlebih dahulu, dimana untuk nilai HMin, SMin, dan VMin di-setting 0, dan nilai HMax, SMax, VMax di setting 255 karena pengolahan citra yang digunakan adalah pengolahan 8 bit.

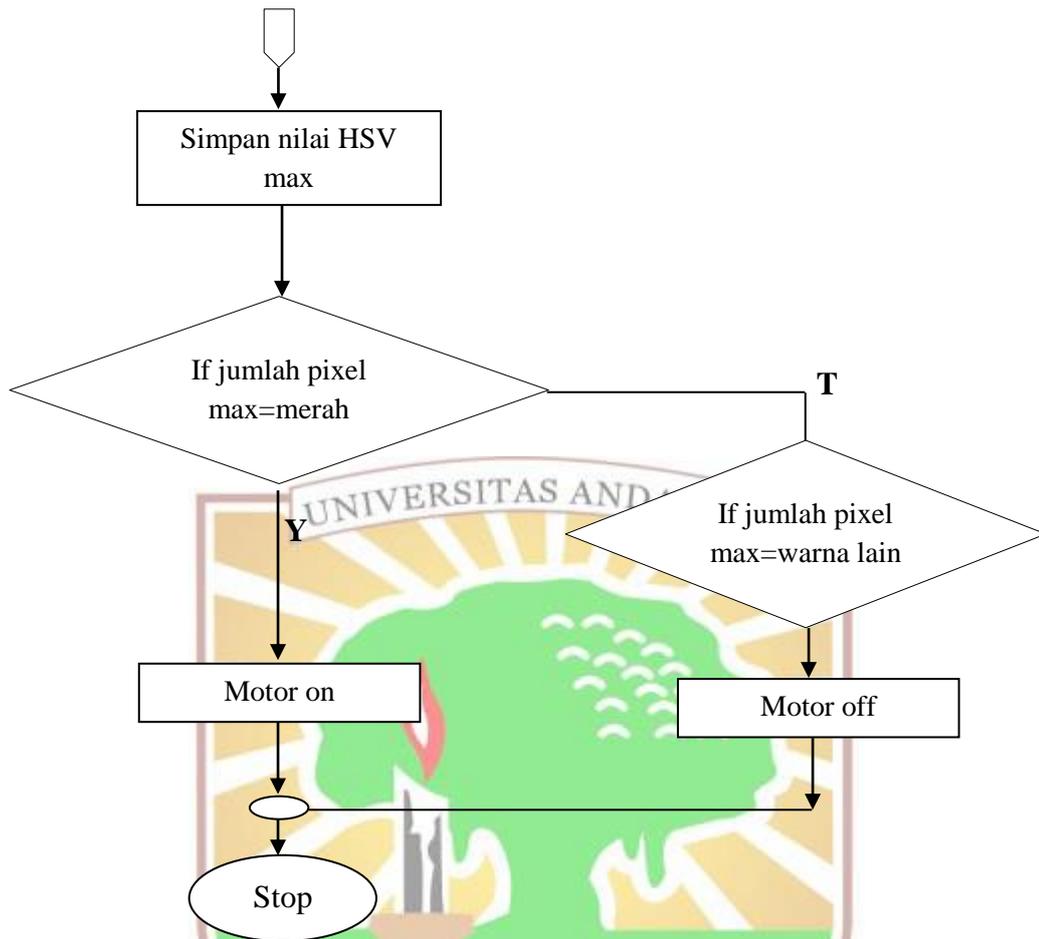
Webcam akan meng-*capture* warna baju , hasil inputan *webcam* tersebut akan diproses didalam Raspberry Pi untuk proses pengolahan citra. Proses pengolahan citra yang terjadi adalah warna barang yang di-*capture* akan di *convert* dari warna RGB ke warna HSV, dan dilakukan proses *thresholding* untuk membedakan warna dengan latar. Agar mendapatkan hasil HSV yang cocok, maka barang di *tracking* dengan menggeser trackbar, sampai didapatkan nilai HSV yang tepat untuk warna yang digunakan. Setelah didapatkan nilai HSV warna yang diinginkan, maka nilai tersebut disimpan dalam variabel untuk digunakan sebagai pembanding pada tahap selanjutnya.

1. Proses analisa warna baju

Pada Gambar 3.8 merupakan *flowchart* proses analisa warna.

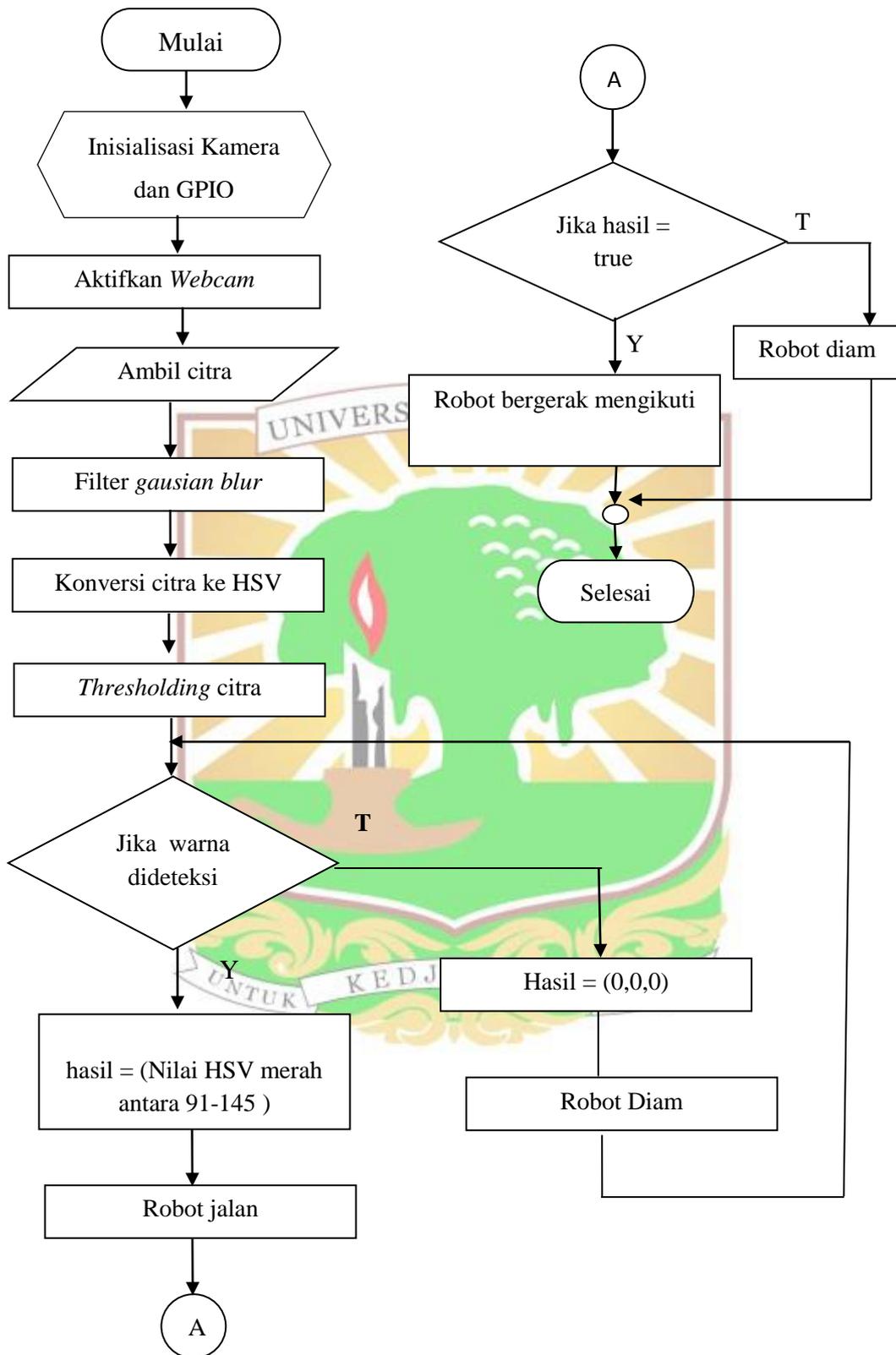






Gambar 3.7 Flowchart Analisa Warna Baju

Dalam proses analisa warna baju, akan dilakukan looping untuk mencari nilai HSV masing-masing *pixel*. *Pixel* yang digunakan dalam proses analisa warna yaitu 160x120. Nilai HSV yang terdeteksi akan dibandingkan dengan nilai HSV warna yang tersimpan pada variabel yang didapat proses mencari nilai HSV di tahap sebelumnya. Setelah dibandingkan maka akan didapatkan jumlah piksel maksimal dari warna merah tersebut.



Gambar 3. 8 *Flowchart Robot Mobil Pengikut Manusia*

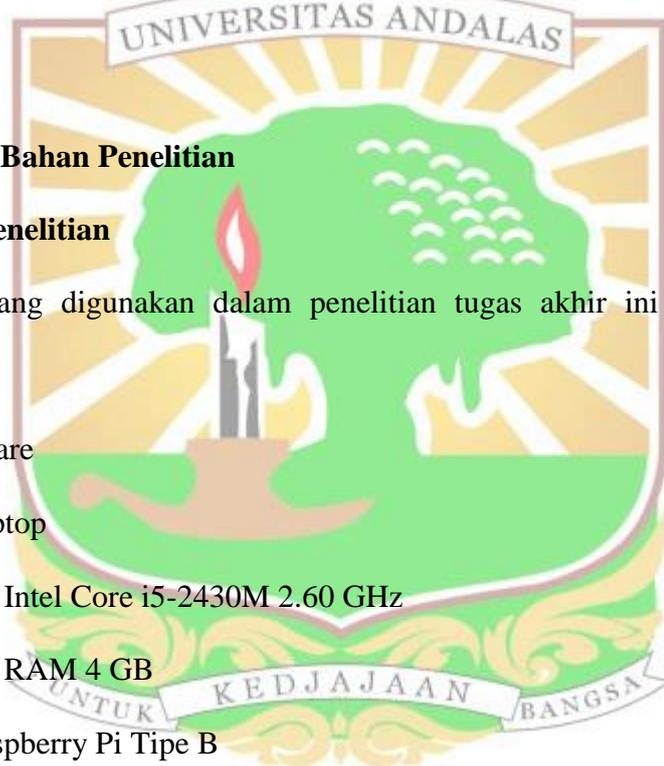
Flowchart pada Gambar 3.10 pergerakan robot menggambarkan bagaimana alur pembuatan program dengan bahasa Python 2.7 dan *library* openCV. Sistem ini dimulai dengan menginisialisasi *webcam* serta pin GPIO pada raspberry pi yang akan berfungsi sebagai kontrol robot. Setelah itu aktifkan *webcam* yang terhubung pada *raspberry pi*. Sistem yang digunakan adalah *autorun*. Setelah terhubung, maka robot akan melakukan *compile* program dengan sendiri nya sehingga robot melakukan pergerakan.

3.4 Alat Dan Bahan Penelitian

3.4.1 Alat Penelitian

Alat yang digunakan dalam penelitian tugas akhir ini adalah sebagai berikut:

- a. Hardware
 1. Laptop
 - Intel Core i5-2430M 2.60 GHz
 - RAM 4 GB
 2. Raspberry Pi Tipe B
 3. Webcam logitech
 4. 2 buah Motor DC dan gear box.
 5. IC L298 Motor Driver.
 6. Baterai 9,6 Volt
- b. Software
 1. Bahasa program yang digunakan Python

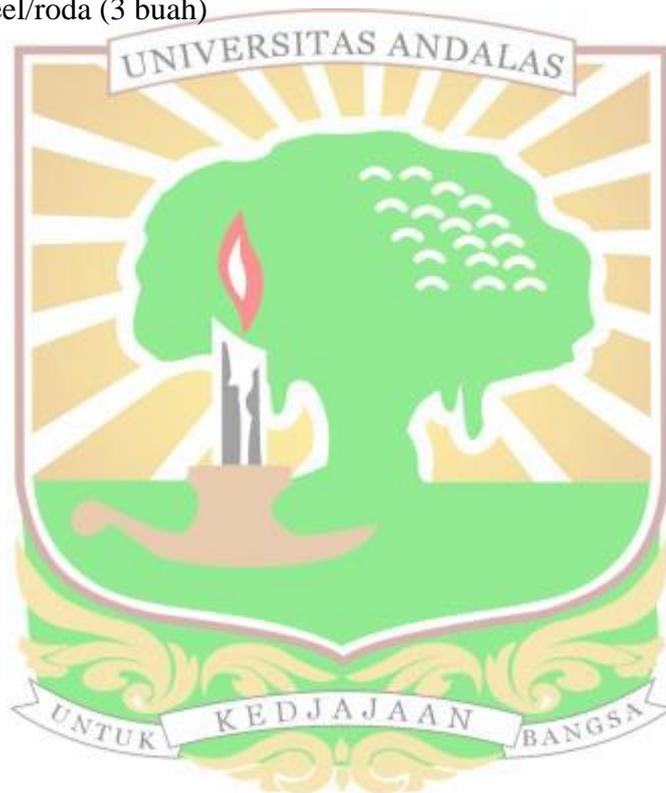


2. Library SimpleCV
3. OS Raspbian

3.4.2 Bahan Penelitian

a. Komponen Mekanik

1. Mur dan baut
2. Wheel/roda (3 buah)



BAB IV

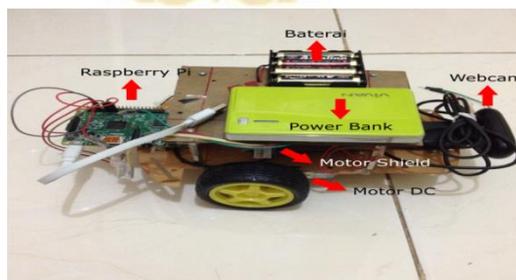
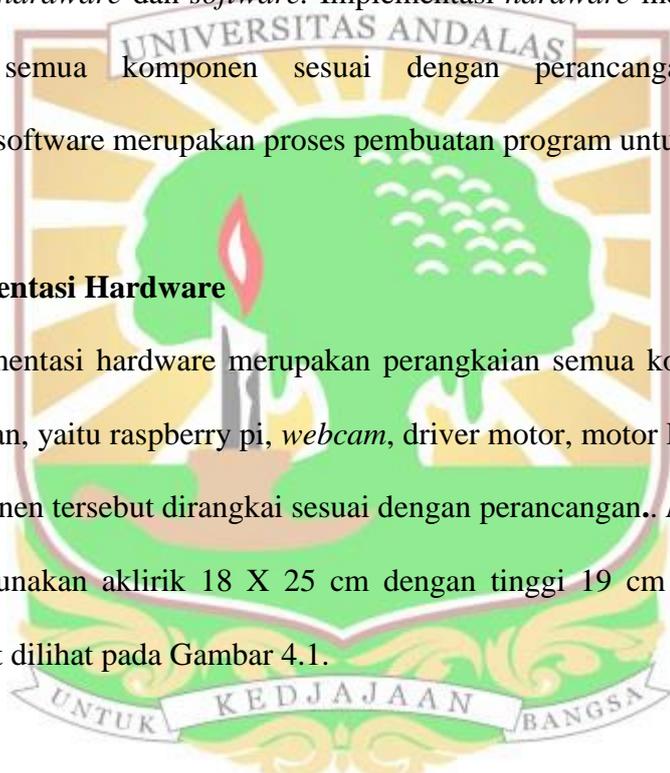
HASIL DAN ANALISA

4.1 Proses

Tahap proses merupakan tahap mengimplementasikan hardware dan software yang telah dirancang pada bagian perancangan. Tahap proses terdiri dari implementasi *hardware* dan *software*. Implementasi *hardware* merupakan proses perangkaian semua komponen sesuai dengan perancangan, sedangkan implementasi *software* merupakan proses pembuatan program untuk sistem.

4.1.1 Implementasi Hardware

Implementasi hardware merupakan perangkaian semua komponen sistem yang diperlukan, yaitu raspberry pi, *webcam*, driver motor, motor DC, dan baterai. Semua komponen tersebut dirangkai sesuai dengan perancangan. *Mobile robot* ini dibuat menggunakan akrilik 18 X 25 cm dengan tinggi 19 cm . *Mobile robot* pengikut dapat dilihat pada Gambar 4.1.



Gambar 4. 1 Komponen setelah dirangkai

4.1.2 Implementasi *Software*

Implementasi software terdiri dari proses instalasi dan pembuatan program. Proses instalasi terdiri dari instalasi GPIO, instalasi *Webiopi*, dan instalasi *Motion*.

4.1.2.1 Instalasi GPIO

Untuk mengakses port GPIO, terlebih dahulu harus dilakukan instalasi library RPi.GPIO. Cara instalasi adalah sebagai berikut :

```
pi@raspberrypi ~ $ sudo apt-get update
```

```
pi@raspberrypi ~ $ sudo apt-get upgrade
```

```
pi@raspberrypi ~ $ sudo apt-get install python-rpi.gpio
```

4.2 Implementasi Pencarian *Template*

Implementasi pencarian *Template* ini dibuat menggunakan library *OpenCV* dan bahasa pemrograman python. Pada implementasi ini terdapat beberapa teknik pengolahan citra yang digunakan. Beberapa diantaranya yaitu:

4.2.1 Pemotongan citra (*cropping*)

Untuk menghubungkan *webcam* dengan raspberry pi digunakan library *OpenCV*. Pada library *OpenCV* ini juga terdapat beberapa program untuk pengolahan citra menggunakan *webcam*. Pada proses *template matching* pemotongan gambar atau *cropping* digunakan untuk mendapatkan gambar

template dari gambar *source* yang ada agar didapatkan gambar *template* yang sesuai kondisinya dengan gambar *source* yang akan dibandingkan, sehingga mendapatkan hasil yang maksimal.

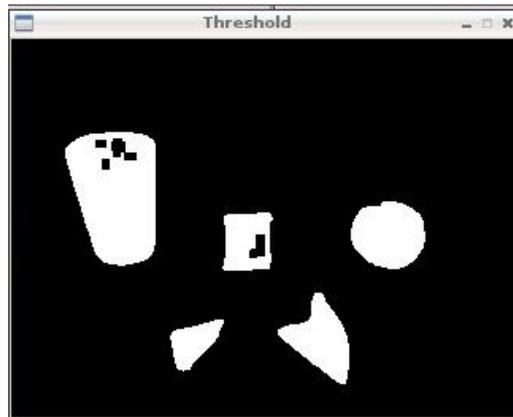
Pada penelitian ini yang akan di *crop* adalah baju yang digunakan untuk dijadikan *template* seperti Gambar 4.2



Gambar 4.2 Cropping Citra

4.2.2 Proses Konversi ke *grayscale*

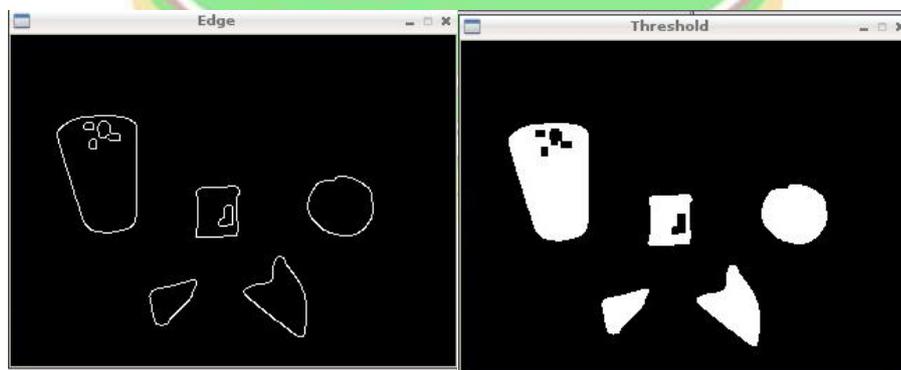
Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi citra *grayscale*, hal ini digunakan untuk menyederhanakan model citra dan mempermudah proses perubahan *grayscale* menjadi biner. Perubahan gambar RGB menjadi *grayscale* menggunakan *library openCV* dengan bahasa pemrograman python menggunakan perintah `cv2.COLOR_BGR2GRAY`. Proses perubahan *grayscale* dapat dilihat pada Gambar 4.3



Gambar 4. 3 Citra grayscale

4.2.3 Proses *Thresholding*

Proses *thresholding* dilakukan untuk mengubah citra berwarna / berderajat keabuan menjadi citra biner (hitam / putih) sehingga dapat diketahui mana yang termasuk objek dan *background*. *Thresholding* atau pengembangan adalah nilai piksel pada citra yang memenuhi syarat nilai ambang yang kita tentukan. Fungsi pemrosesan *thresholding* pada *library opencv* ini digunakan $threshold = 0.5$.



Gambar 4. 4 Citra Threshold

4.2.4 Proses Pendeteksian warna

Pendeteksian warna dilakukan pada daerah bagian bawah dan setengah bagian dari lebar badan manusia. Area yang akan dilakukan deteksi warna

ditunjukkan pada Gambar 4. Dilambangkan dengan kotak bergaris diagonal. Pada pengenalan warna diperlukan proses binerisasi. Binerisasi diperlukan untuk membedakan warna yang dimaksud dengan warna latar belakang lainnya. Setelah itu jika target ditemukan, kemudian target akan ditandai. Kemudian langkah terakhir adalah mencari titik pusat target.



Gambar 4.5 Area pendeteksian warna

4.3 Pengujian *Hardware*

Pengujian *hardware* bertujuan untuk mengetahui bahwa *hardware* yang digunakan bekerja baik atau tidak. Untuk mengetahuinya, maka perlu dilakukan serangkaian pengujian *hardware* yang akan digunakan didalam sistem tersebut.

Dalam pembuatan *mobile robot* pengikut seperti pada Gambar 4.2, terdiri dari Raspberry pi, motor DC, otor *shield* L298P, baterai 9,6 Volt, *power bank*, *webcam*, Pada implementasi alat ini, menggunakan *supply* dari *power bank* dan baterai. Baterai 9,6 volt digunakan sebagai *supply* untuk motor *shield*, sedangkan *power bank* digunakan untuk *supply* ke Raspberry pi.

4.4 Pengujian dan Analisa Sistem

Dalam melakukan pengujian sistem ada beberapa hal yang dilakukan, yaitu pengujian mencari nilai HSV, dan pengujian sistem secara keseluruhan.

4.4.1 Pengujian *webcam* dalam mendeteksi objek

Sudut yang digunakan pada *webcam* dalam mendeteksi objek sehingga *mobile* robot dapat bergerak menuju arah objek yang dideteksi. Pada tabel 4.1 merupakan tabel pengujian sudut *webcam*.

Tabel 4. 1 Pengujian sudut webcam

NO	POSISI OBJEK DARI WEBCAM	OBJEK YANG DIDETEKSI	HASIL
1	0 ⁽⁰⁾	-	Tidak terdeteksi
2	30 ⁽⁰⁾		Terdeteksi

3	45 ⁽⁰⁾		Terdeteksi
4	60 ⁽⁰⁾		Terdeteksi
5	90 ⁽⁰⁾		Terdeteksi



Pengujian untuk pencarian sudut *webcam* yang tepat untuk dinding tujuan dilakukan sebanyak 10 kali, yaitu dari sudut 0° sampai sudut 90°. Dari 5 kali percobaan didapatkan sudut *webcam* yang cocok dalam mendeteksi objek warna baju adalah sudut *webcam* untuk kiri (sudut 30°), lurus (sudut 60°), dan kanan (sudut 90°). Posisi awal *webcam* di *setting* ke keadaan lurus, dengan sudut 60°.

Dari pengujian yang dilakukan untuk memutar motor , didapatkan hasil sesuai dengan yang diharapkan, yaitu berputar ke kiri, lurus dan kekanan dengan baik.

4.4.2 Pengujian Motor *Shield* L298 terhadap kecepatan motor

Pengujian motor *shield* L298 bertujuan untuk mengetahui hasil kecepatan PWM yang dihasilkan oleh motor DC 12V. Pengujian dilakukan dengan cara menghubungkan motor *shield* L298P dengan *supply* yang berasal dari baterai 9,6 Volt, dan motor DC. Tabel 4.2 adalah tabel pengujian tegangan keluaran dari kecepatan PWM

Tabel 4. 2 PengujianMotor Shield L298P

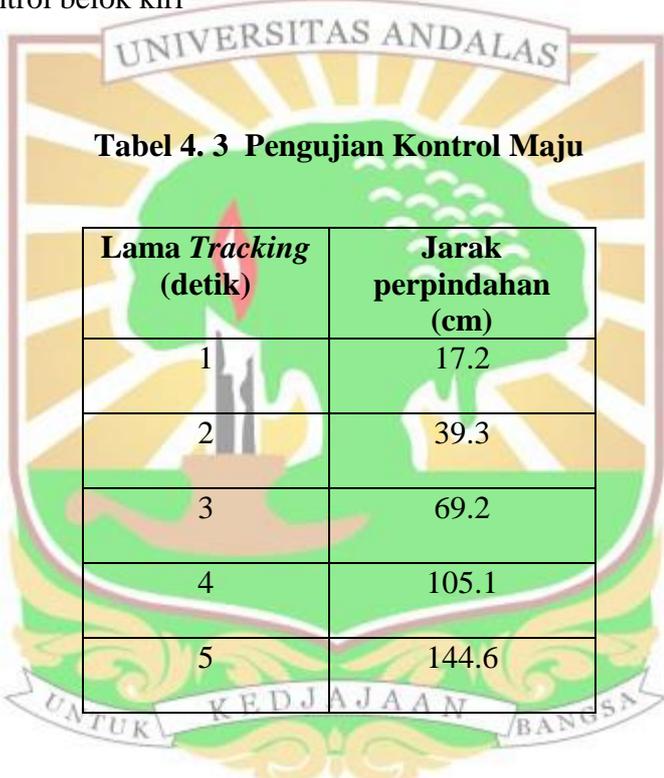
NO	Tegangan masukan (volt)	Kecepatan PWM	Tegangan keluaran	
			Motor 1	Motor 2
1	10,62	0	0	0
2	10,62	50	1,72	1,60
3	10,62	100	5,1	5,04
4	10,62	150	6,4	6,38
5	10,62	200	7,5	7,49

Pengujian yang dilakukan pada motor *shield* dengan memberikan tegangan masukan yang sama, didapat hasil tegangan keluaran pada motor 1 dan motor 2 berbeda, seperti terlihat pada Tabel 4.2. Hal ini terjadi karena tegangan keluaran

pada motor mempengaruhi kecepatan PWM, semakin besar tegangan keluaran maka semakin besar pula PWM yang dihasilkan.

Setelah nilai PWM untuk masing-masing kontrol diberikan, dilakukan pengujian. Proses pengujian ini dilakukan terdiri dari :

1. Kontrol maju
2. Kontrol belok kanan
3. Kontrol belok kiri



Tabel 4. 3 Pengujian Kontrol Maju

Lama Tracking (detik)	Jarak perpindahan (cm)
1	17.2
2	39.3
3	69.2
4	105.1
5	144.6

Dari tabel pengujian diatas, robot akan mengalami perpindahan ketika proses *tracking*. Jarak perpindahan yang terjadi setiap detik adalah :

$$\text{Rata-rata} = \frac{\text{Jumlah perpindahan}}{\text{Jumlah waktu}} = \frac{375.4}{15} = 25.03 \text{ cm / detik}$$

Tabel 4. 4 Pengujian Kontrol Belok Kanan

Lama Tracking (detik)	Sudut belok robot (derajat)
1	37
2	74
3	92
4	115
5	164

Dari tabel pengujian diatas, robot akan mengalami perubahan arah ketika objek bergerak ke kanan . Perubahan arah robot yang terjadi setiap detik *webcam* bergerak adalah :

$$\text{Rata-rata} = \frac{\text{Jumlah derajat perubahan arah}}{\text{Jumlah waktu}} = \frac{482}{15} = 32.13 \text{ derajat / detik}$$

Tabel 4. 5 Pengujian Kontrol Belok Kiri

Lama tracking (detik)	Sudut belok robot (derajat)
1	41
2	78
3	111
4	165
5	191

Dari tabel pengujian diatas, robot akan mengalami perubahan arah ketika objek bergerak kekiri . Perubahan arah robot yang terjadi setiap adalah :

$$\text{Rata-rata} = \frac{\text{Jumlah derajat perubahan arah}}{\text{Jumlah waktu}} = \frac{586}{15} = 39.07 \text{ derajat / detik}$$

4.4.3 Pengujian Kontrol Gerak Robot Mengikuti Manusia

Pengujian kontrol pergerakan robot dilakukan untuk mengetahui bagaimana respon robot terhadap manusia yang diikuti. Robot akan bergerak sesuai dengan nilai PWM yang telah didefinisikan pada program. Setelah nilai PWM untuk masing-masing kontrol diberikan, dilakukan pengujian. Proses pengujian ini dilakukan terdiri dari :

1. Kontrol maju.
2. Kontrol belok kanan.
3. Kontrol belok kiri.

4.4.3.1 Kontrol Gerak Maju

Pengujian untuk kontrol maju dilakukan dengan menggerakkan robot untuk berjalan maju selama lima detik. Pengujian dilakukan sebanyak 30 kali dan dihitung jarak yang ditempuh robot pada setiap pengujian. Hasil pengujian dapat dilihat pada Tabel 4.6 :

Tabel 4. 6 Pengujian Kontrol Gerak Maju

Pengujian Ke	Jarak (cm)
1	172
2	160
3	157
4	166
5	151
6	151
7	201
8	182
9	202
10	188
11	197
12	192
13	165
14	202
15	217
16	215
17	208
18	198
19	206
20	208
21	199
22	212
23	194
24	197
25	194
26	195
27	195
28	180
29	189
30	183

Dari tabel pengujian di atas, dapat dihitung rata-rata jarak yang ditempuh oleh robot untuk gerak maju :

$$\text{Rata-rata} = \frac{\text{Jumlah jarak}}{\text{Jumlah pengujian}} = \frac{5676}{30} = 189,2 \text{ cm}$$

4.4.3.2 Kontrol Gerak Belok Kanan

Pengujian kontrol belok kanan dilakukan dengan menggerakkan robot berbelok sejauh 30, 45, 60, 90 derajat. Pertama robot diletakkan pada posisi 0 derajat, kemudian dilakukan input manusia berada pada sudut 30, 45, 60, 90 . Pengujian dilakukan sebanyak 20 kali dengan menghitung perubahan derajat gerak robot untuk masing-masing pengujian. Hasil pengujian dapat dilihat pada Tabel 4.7.

Tabel 4. 7 Pengujian Kontrol Gerak Belok Kanan

Pengujian Ke	Sudut yang diinginkan (derajat)	Sudut sebenarnya (derajat)	Selisih (derajat)
1	90	87	3
2	90	89	1
3	90	97	7
4	90	88	2
5	90	89	1
6	30	26	4
7	30	22	8
8	30	28	2
9	30	24	6
10	30	29	1
11	45	42	3
12	45	47	2
13	45	48	3
14	45	39	6
15	45	41	4
16	60	62	2
17	60	59	1
18	60	52	8
19	60	67	3
20	60	66	4

Dari tabel di atas, dapat dihitung rata-rata selisih derajat pergerakan robot untuk belok kanan sejauh 90 derajat.

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{14}{5} = 2,8 \text{ derajat}$$

Pergerakan robot belok kanan sejauh 30 derajat

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{18}{5} = 4,2 \text{ derajat}$$

Pergerakan robot belok kanan sejauh 45 derajat

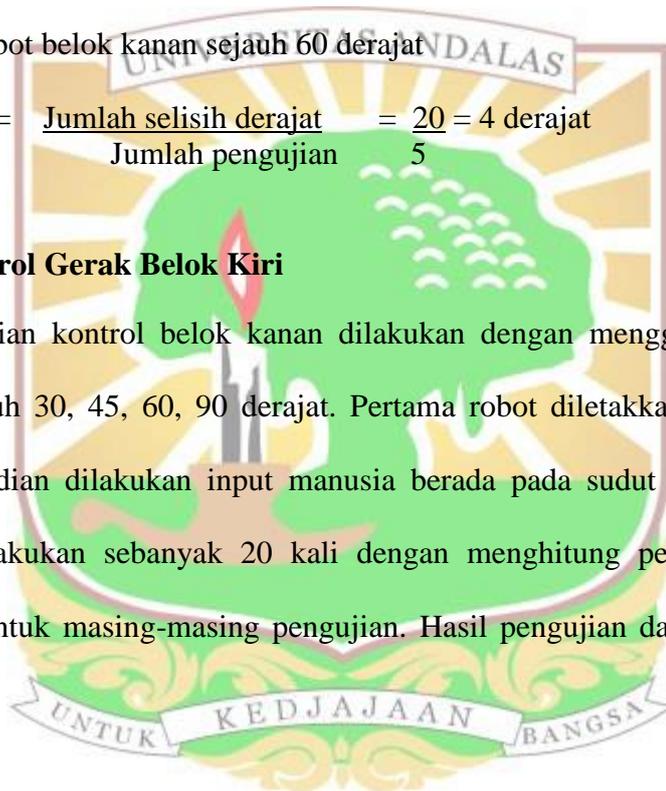
$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{21}{5} = 3,6 \text{ derajat}$$

Pergerakan robot belok kanan sejauh 60 derajat

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{20}{5} = 4 \text{ derajat}$$

4.4.3.3 Kontrol Gerak Belok Kiri

Pengujian kontrol belok kanan dilakukan dengan menggerakkan robot berbelok sejauh 30, 45, 60, 90 derajat. Pertama robot diletakkan pada posisi 0 derajat, kemudian dilakukan input manusia berada pada sudut 30, 45, 60, 90. Pengujian dilakukan sebanyak 20 kali dengan menghitung perubahan derajat gerak robot untuk masing-masing pengujian. Hasil pengujian dapat dilihat pada Tabel 4.8.



Tabel 4. 8 Pengujian Kontrol Gerak Belok Kiri

Pengujian Ke	Sudut yang Diinginkan (derajat)	Sudut Sebenarnya (derajat)	Selisih (derajat)
1	90	90	0
2	90	97	8
3	90	85	5
4	90	95	5
5	90	84	6
6	30	32	2
7	30	26	4
8	30	24	6
9	30	27	3
10	30	22	8
11	45	47	2
12	45	51	6
13	45	40	5
14	45	39	6
15	45	41	4
16	60	64	4
17	60	58	2
18	60	51	9
19	60	67	3
20	60	63	3

Dari tabel di atas, dapat dihitung rata-rata selisih derajat pergerakan robot untuk belok kiri sejauh 90 derajat.

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{28}{20} = 1,4 \text{ derajat}$$

Pergerakan robot belok kiri sejauh 30 derajat

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{28}{20} = 1,4 \text{ derajat}$$

Pergerakan robot belok kiri sejauh 45 derajat

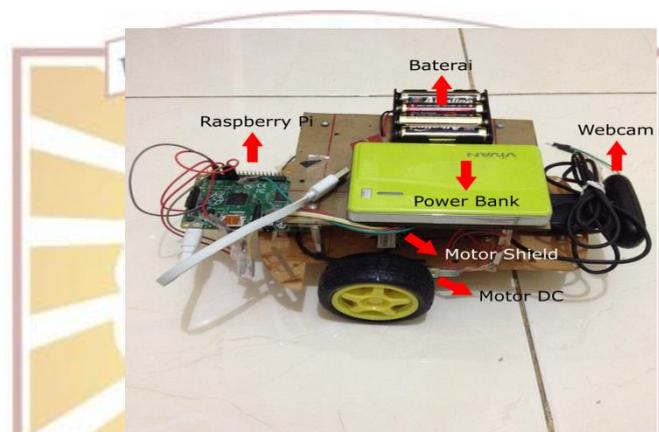
$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{23}{20} = 1,15 \text{ derajat}$$

Pergerakan robot belok kiri sejauh 60 derajat

$$\text{Error} = \frac{\text{Jumlah selisih derajat}}{\text{Jumlah pengujian}} = \frac{17}{20} = 0,85 \text{ derajat}$$

4.5 Implementasi Alat

Implementasi alat pada *mobile* robot pengikut manusia dapat dilihat pada Gambar 4.2.



Gambar 4.6 Mobile robot

Dalam pembuatan *mobile* robot pengikut seperti pada Gambar 4.2, terdiri dari Raspberry pi, motor DC, motor *shield* L298P, baterai 9,6 Volt, *power* bank, *webcam*, *usb* hub,. Pada implementasi alat ini, menggunakan *supply* dari *power* bank dan baterai. Baterai 9,6 volt digunakan sebagai *supply* untuk motor *shield*, sedangkan *power* bank digunakan untuk *supply* Raspberry pi.

4.5.1 Pengujian dan Analisa Sistem

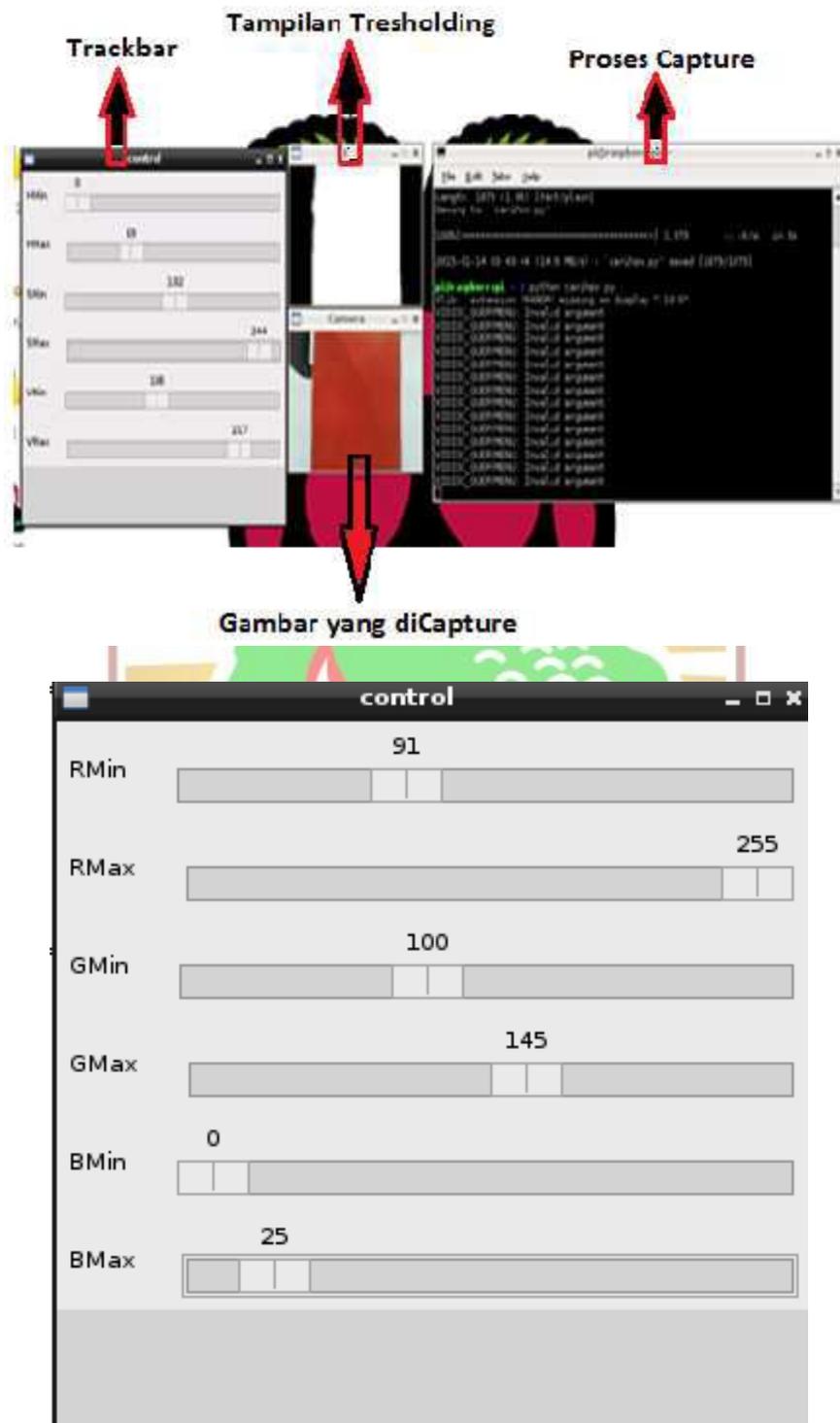
Dalam melakukan pengujian sistem ada beberapa hal yang dilakukan, yaitu pengujian mencari nilai HSV dan pengujian analisa objek di titik *start* berdasarkan intensitas cahaya.



UNIVERSITAS ANDALAS
Gambar 4.7 Mobile Robot

4.5.2 Pengujian mencari nilai HSV

Dalam pembuatan *mobile* robot pengikut manusia ini, menggunakan metode *template matching*. Sehingga sebelum *mobile* robot menganalisa warna pada titik awal, dibutuhkan sebuah nilai yang berguna sebagai pembanding. Untuk mencari nilai HSV yang digunakan sebagai pembanding, dalam pengujian sistem untuk *tracking* warna baju yang akan digunakan pada *trackbar*. *Trackbar* digeser sampai didapatkan nilai HSV yang cocok. Pada Gambar 4.3 adalah *screenshot* dari proses mencari nilai HSV.



Gambar 4.8 Mencari nilai HSV

Dari Gambar 4.3 dapat dilihat setelah program python untuk mencari nilai HSV di jalankan, maka akan muncul jendela tampilan untuk menampilkan gambar yang di *capture*, hasil *tresholding* gambar yang di *capture*, dan jendela *trackbar*. Pada proses mencari nilai HSV pada *trackbar disetting* menjadi 0, dan

Nilai HSV *disetting* menjadi 255. Nilai 0 sampai 255 digunakan, karena dalam pengujian yang dilakukan menggunakan pengolahan citra 8 bit, sehingga nilai maksimal yang didapatkan dalam pengujian adalah 255. Pada jendela tampilan hasil *tresholding* dapat dilihat, perubahan tampilan apabila *trackbar* digeser.

Berikut adalah data respon *webcam* terhadap intensitas cahaya untuk warna dengan perbedaan kondisi pencahayaan.

Tabel 4. 9 Pengujian Respon webcam pada ruangan intensitas cahaya terang

No	Gambar hasil <i>template matching</i>	Intensitas cahaya (lux)	Jarak (cm)	Hasil	keputusan
1		21	40	terdeteksi	Robot berjalan

2		20	80	Terdeteksi	Robot berjalan
3		17	120	Terdeteksi	Robot berjalan
4		12	160	Tidak terdeteksi	Robot diam
5		10	20	Tidak terdeteksi	Robot diam
6		37	35	Tidak terdeteksi	Robot diam

7		13	150	Terdeteksi	Robot maju
---	---	----	-----	------------	------------

$$\%Keberhasilan = \frac{\text{Jumlah pengujian} - \text{Pengujian gagal}}{\text{Jumlah pengujian}} \times 100\%$$

$$= \frac{7 - 2}{7} \times 100\% = 71,4 \%$$

Pada Percobaan diatas, dari tujuh kali percobaan tingkat keberhasilannya pendekteksian warna baju mencapai 71,4 persen. Ini dikarenakan intensitas cahaya yang cukup pada saat pengcapturean gambar sehingga proses *template matching* dapat medeteksi *template* (baju) sesuai dengan hasil yang diharapkan dan jarak pada saat pengcapturean gambar yang ideal. Dengan tingkat keberhasilan 71,4% persen, maka dapat dinyatakan alat mampu mendeteksi *template* baju berwarna merah dengan baik. tingkat *error* dalam pengujian yang dihasilkan berkisar 28,6%

Tabel 4. 10 Pengujian Respon webcam pada ruangan dengan intensitas cahaya Redup

No	Gambar hasil <i>template matching</i>	Intesitas cahaya (lux)	Jarak (cm)	Hasil	Keputusan
1		10	70	terdeteksi	Robot berjalan
2		7	130	Tidak terdeteksi	Robot diam
3		3	150	Tidak terdeteksi	Robot diam
4		5	210	terdeteksi	Robot berjalan

Pada Percobaan diatas, dari 4 percobaan tingkat keberhasilannya pendekteksian objek mencapai persen. Rendahnya tingkat keberhasilan pendeteksian dikarenakan ini disebabkan oleh pengaruh cahaya, posisi kamera, warna objek dan nilai *thresholding* yang membuat proses peng-*capturean* sulit.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari pembahasan implementasi pada robot *mobile* pengikut manusia diatas, dapat disimpulkan beberapa hal sebagai berikut:

1. Berdasarkan hasil pengujian, hasil pendeteksian yang dilakukan Robot dapat mengenali objek manusia yang diikuti dari berdasarkan warna baju dan barang dan mampu membedakan dengan objek lainnya.
2. Robot *tracking* membutuhkan pengolahan citra untuk bisa mengenali objek manusia dengan warna merah.
3. Untuk mendeteksi warna dengan metode *template matching*, dilakukan beberapa tahap, yaitu: pengambilan citra, konversi ke HSV, *thresholding*.
4. Pengaturan sudut webcam yang tepat untuk mendeteksi objek manusia ke kiri, lurus dan ke kanan adalah 30 derajat, 60 derajat, dan 90 derajat.
5. Pencahayaan sangat berpengaruh terhadap penentuan nilai warna yang diperoleh. Dari 11 kali pengujian sistem yang dilakukan pada semua kondisi pencahayaan, didapatkan 4 kali kegagalan , dan 8 kali keberhasilan dengan persentase keberhasilan sebesar 71,4% dan *eror* 28,6%.
6. Robot dapat bergerak mengikuti pergerakan manusia dengan memberikan nilai PWM pada motor DC untuk perintah maju, belok kanan, belok kiri. Nilai PWM yang diberikan pada perintah maju dan perintah mundur 204,6

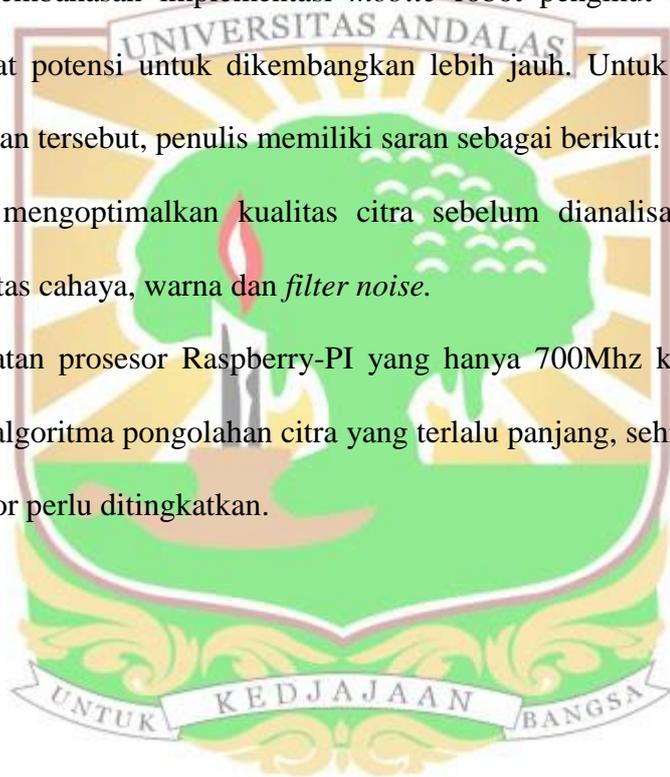
pada masing – masing motor DC, perintah belok kanan 48,8 dan 204,6 ,
perintah belok kiri 204,6

7. Robot memiliki tingkat respon yang baik ditempat cahaya terang (48,0 lux) dibandingkan tempat yang lebih redup (28,0 lux).

5.2 Saran

Dari pembahasan implementasi *mobile* robot pengikut manusia diatas, penulis melihat potensi untuk dikembangkan lebih jauh. Untuk pengembangan dari pembahasan tersebut, penulis memiliki saran sebagai berikut:

1. Lebih mengoptimalkan kualitas citra sebelum dianalisa, baik itu dari intensitas cahaya, warna dan *filter noise*.
2. Kecepatan prosesor Raspberry-PI yang hanya 700Mhz kurang memadai untuk algoritma pengolahan citra yang terlalu panjang, sehingga kecepatan prosesor perlu ditingkatkan.



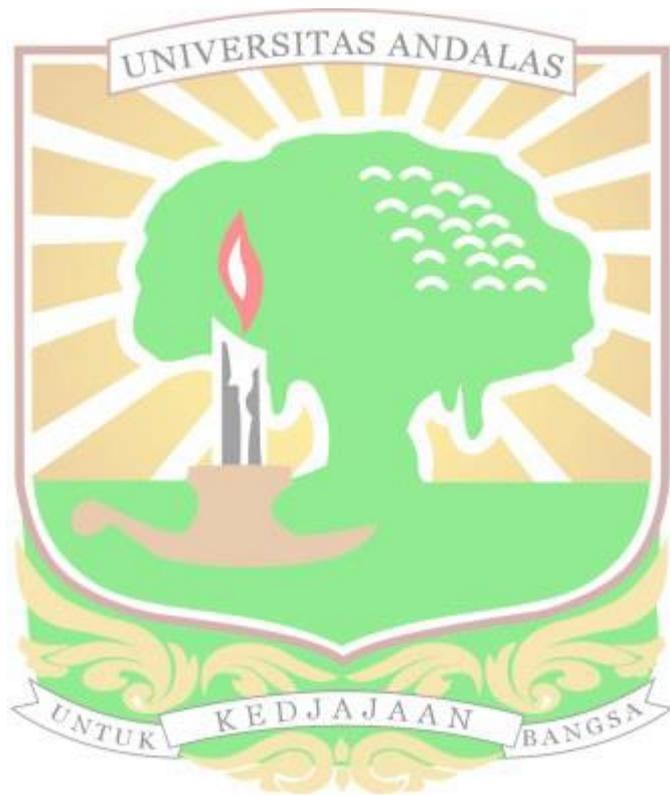
DAFTAR PUSTAKA

- [1] Umbaugh, Scott E. C. 1998. *Computer Vision and Image Processing*, Englewood Cliffs,NJ:Prentice-Hall.
- [2] Basuki, Achmad. 2007. Pengantar Pengolahan Citra. PENS-ITS Surabaya.
- [3] Smith, R., and Cheeseman, P., “On the Representation of and Estimation of Spatial Uncertainty,” *International Journal of Robotics Research*, vol. 5, 1986, pp. 56–68.
- [4] Munir, Rinaldi. 2004. *Pengolahan Citra Digital dan Teknik Pemrogramannya serta bagian-bagian Webcam*. Institut Teknologi Bandung : Bandung.
- [5] Jatmiko, Wisnu dkk. 2012. *Robotika: Teori dan Aplikasi*. Jakarta. Fakultas Ilmu Komputer, Universitas Indonesia.
- [6] Budiharto, Widodo. 2012. *Robotika Modern*. Yogyakarta: Penerbit Andi.
- [7] Richardson, M dan S. Wallace. 2013. *Getting Started With Raspberry Pi*. O’Reilly Media, Inc., USA
- [8] Maruch, S dan Maruch, A. 2006. *Phyton for Dummies for Dummies*, USA.
- [9] Winarno, Edy. 2009. *Pengolahan Citra*. Unisbank: Semarang
- [10] Nugraha, Ali. 2008. *Pengembangan Pembelajaran Sains pada Anak Usia Dini*. JILSI Foundation: Bandung.
- [11] Putranto, Benedictus dkk. 2010. *Segmentasi Warna Citra Dengan Deteksi Warna HSV Untuk Mendeteksi Objek*.
<http://download.portalgaruda.org/article.php?article=59719&val=4485>,

diakses tanggal 4 januari 2016, jam 20.05 Wib

- [12] Wardhana, dkk. 2008. *Penggunaan metode templete matching untuk identifikasi kecacatan pada PCB.*

<http://journal.uui.ac.id/index.php/snati/article/download/895/831>, diakses tanggal 4 januari 2016, jam 21.00 Wib.



LAMPIRAN

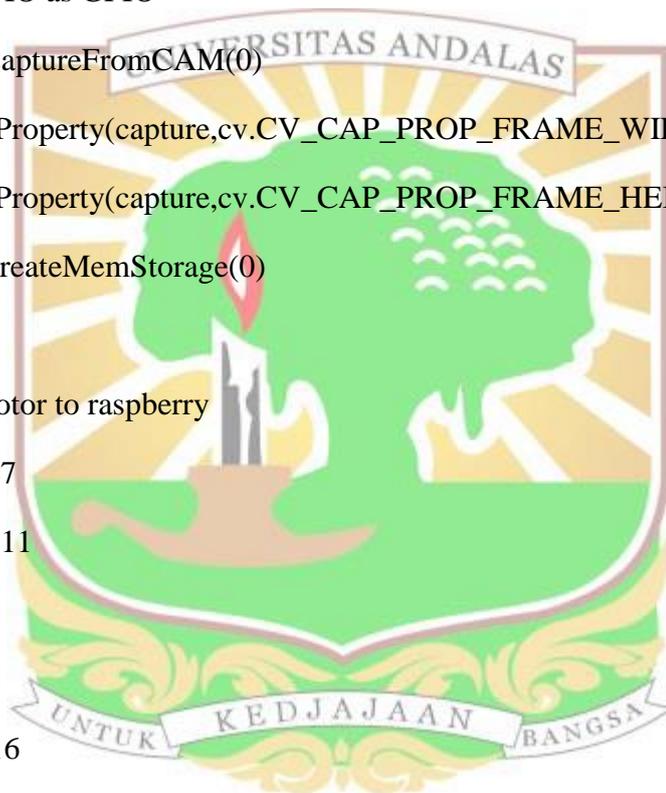
BAHASA PEMOGRAMAN

```
import cv
import time
from time import sleep
import serial
import RPi.GPIO as GPIO

capture = cv.CaptureFromCAM(0)
cv.SetCaptureProperty(capture,cv.CV_CAP_PROP_FRAME_WIDTH,200)
cv.SetCaptureProperty(capture,cv.CV_CAP_PROP_FRAME_HEIGHT,200)
storage = cv.CreateMemStorage(0)

#Inisial pin Motor to raspberry
motorkanan1=7
motorkanan2=11
motorkiri1=13
motorkiri2=15
#pwmkanan=16
#pwmkiri=18

GPIO.setmode(GPIO.BOARD)
GPIO.setup(motorkanan1, GPIO.OUT)
GPIO.setup(motorkanan2, GPIO.OUT)
GPIO.setup(motorkiri1, GPIO.OUT)
GPIO.setup(motorkiri2, GPIO.OUT)
#GPIO.setup(pwmkanan, GPIO.OUT)
```



```
#GPIO.setup(pwmkiri, GPIO.OUT)
```

```
#range warna hijau
```

```
"rMin=71
```

```
rMax=127
```

```
gMin=119
```

```
gMax=181
```

```
bMin=0
```

```
bMax=83"
```

```
#orange
```

```
rMin=175
```

```
rMax=255
```

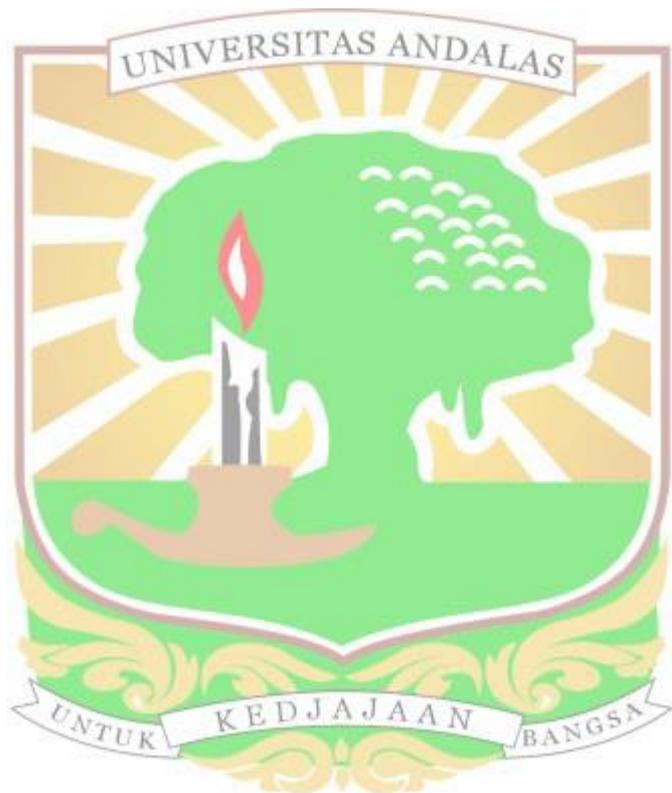
```
gMin=130
```

```
gMax=145
```

```
bMin=0
```

```
bMax=18
```

```
points = []
```



```
def GambarThreshold(gambarHSV):
```

```
    gambarThresh =
```

```
    cv.CreateImage(cv.GetSize(gambarHSV),cv.IPL_DEPTH_8U, 1)
```

```
    "rMin = cv.GetTrackbarPos('RMin', 'control')
```

```
    rMax = cv.GetTrackbarPos('RMax', 'control')
```

```
    gMin = cv.GetTrackbarPos('GMin', 'control')
```

```

gMax = cv.GetTrackbarPos('GMax', 'control')

bMin = cv.GetTrackbarPos('BMin', 'control')

bMax = cv.GetTrackbarPos('BMax', 'control')'''

cv.InRangeS(gambarHSV, cv.Scalar(bMin,gMin,rMin),
cv.Scalar(bMax,gMax,rMax), gambarThresh)

#cv.Erode(gambarThresh, gambarThresh, iterations=5)

#cv.Dilate(gambarThresh, gambarThresh, iterations=5)

return gambarThresh
def Track(gambarThresh):

obj = cv.FindContours(gambarThresh, storage, cv.CV_RETR_CCOMP,
cv.CV_CHAIN_APPROX_SIMPLE)

if not obj:

return(0,0,0,0)

else:

return cv.BoundingRect(obj)
'''if __name__ == '__main__':

def nothing(*argv):

pass

cv.NamedWindow('control', 0)

cv.CreateTrackbar('RMin', 'control', rMin, 255, nothing)

cv.CreateTrackbar('RMax', 'control', rMax, 255, nothing)

cv.CreateTrackbar('GMin', 'control', gMin, 255, nothing)

cv.CreateTrackbar('GMax', 'control', gMax, 255, nothing)

cv.CreateTrackbar('BMin', 'control', bMin, 255, nothing)

cv.CreateTrackbar('BMax', 'control', bMax, 255, nothing)'''

```



while True:

```
    frame = cv.QueryFrame(capture)

    gambarHSV = cv.CreateImage(cv.GetSize(frame),cv.IPL_DEPTH_8U, 3)

    cv.CvtColor(frame, gambarHSV, 40)

    gambarThresh = GambarThreshold(gambarHSV)

    cv.Smooth(gambarThresh,gambarThresh,cv.CV_GAUSSIAN,9,9)

    track = Track(gambarThresh)

    middle = (track[0]+(track[2]/2),track[1]+(track[3]/2))

    cv.Circle(frame,middle,((track[2]+track[3])/4),(0,210,0),3,8,0)

    cv.ShowImage("Camera", frame)

    #print track [0],track [1],track [2],track [3]

    #if track [0] > 66 :

    #if ((track [0] > 38 and track [0] <=106) and (track [1] > 47 and track [1] <=75)
and (track [2] > 47 and track [2] <=75) and (track [3] > 32 and track [3] <=50)) :

    if track [0] > 48 and track [0] <=106:

        GPIO.output(motorkanan1, GPIO.HIGH)

        GPIO.output(motorkanan2, GPIO.LOW)

        #GPIO.PWM(pwmkanan, 100)

        GPIO.output(motorkiri1, GPIO.HIGH)

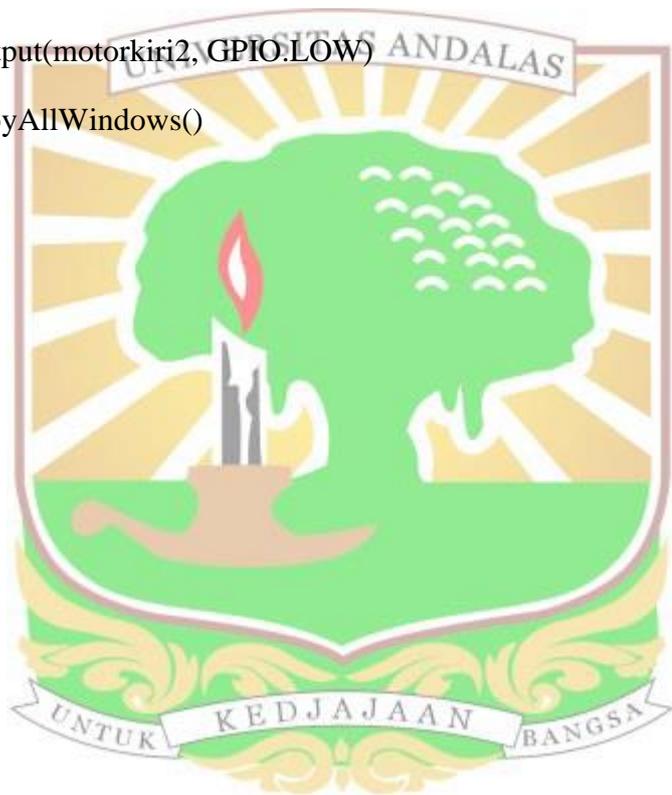
        GPIO.output(motorkiri2, GPIO.LOW)

        #GPIO.PWM(pwmkiri, 100)

    else:

        GPIO.output(motorkanan1, GPIO.LOW)
```

```
GPIO.output(motorkanan2, GPIO.LOW)
GPIO.output(motorkiri1, GPIO.LOW)
GPIO.output(motorkiri2, GPIO.LOW)
if cv.WaitKey(10) == 27:
    GPIO.output(motorkanan1, GPIO.LOW)
    GPIO.output(motorkanan2, GPIO.LOW)
    GPIO.output(motorkiri1, GPIO.LOW)
    GPIO.output(motorkiri2, GPIO.LOW)
    cv.DestroyAllWindows()
    break
```



DAFTAR RIWAYAT HIDUP (*CURRICULUM VITAE*)



Data Pribadi

Nama : Rendy Dartha Nugraha
Tempat, Tanggal Lahir : Bukittinggi, 26 Januari 1992
Jenis Kelamin : Laki-laki
Kewarganegaraan : Indonesia
Agama : Islam
Alamat : Jl. Syech Ibrahim Musa no38 B, Bukittinggi
Status : Belum Menikah
No.Hp : 081267991270
Email : rendydharta26@gmail.com

Riwayat Pendidikan

<u>Tahun</u>	<u>Pendidikan</u>
<u>1997 - 1998</u>	<u>TK Bhayangkari Bukittinggi</u>
<u>1998 - 2004</u>	<u>SD Negeri 07 Simpang Tembok, Bukittinggi</u>
<u>2004 - 2007</u>	<u>SMPS XAVERIUS Bukittinggi</u>
<u>2007 - 2010</u>	<u>SMA Negeri 5 Bukittinggi</u>
<u>2010 - 2015</u>	<u>S1 Jurusan Sistem Komputer, Fakultas Teknologi Informasi Universitas Andalas</u>

Pengalaman Organisasi dan Pekerjaan

<u>Tahun</u>	<u>ORGANISASI DAN PEKERJAAN</u>
<u>2011 - 2012</u>	<u>Staf Bidang INFOKOM BEM KM FMIPA UNAND</u>
<u>2012 - 2013</u>	<u>Staf Divisi INFOKOM FTI UNAND</u>
<u>2013 - 2014</u>	<u>Staf Divisi INFOKOM HMSK FTI UNAND</u>
<u>2014</u>	<u>Praktek Kerja Lapangan di PT Telkom Indonesia . Bandung</u>

Pengalaman Seminar dan Pelatihan

1. Seminar IT Pekan Elektro 2011 “General Manager Community Project KASKUS”,
2. Seminar Nasional Teknologi Robotik dan Kontribusinya terhadap Indonesia
3. Seminar Nasional Perkembangan Pendidikan Tinggi Komputer di Indonesia
4. Seminar Nasional “Awaken The Dragon Within Metasploit and Web Attack
5. Seminar Robotic Line Tracer
6. Training Introduction To Linux
7. ESQ Basic Training
8. Achievement Motivation Training
9. BBMK (Bina Bakat Minat dan Kepemimpinan) FKI RABBANI

Kemampuan Bahasa dan Komputer

Bahasa:

1. Bahasa Indonesia
2. Bahasa Inggris

Komputer:

1. Sistem Operasi
Windows, Linux
2. Bahasa Pemrograman
Pascal, Python, PHP, HTML, CSS
3. Data base
MySql
4. Software Office
Microsoft Office, Libre Office
5. Mikrokontroler dan Mini PC
Arduino, Raspberry Pi
6. Grafis dan Video
Sony Vegas, Inkscape

