

# **Design and Implementation of Q-Learning for Mobile Robot Navigation in Dynamic Environments**

A THESIS PROPOSAL SUBMITTED TO  
THE SCHOOL OF ELECTRICAL ENGINEERING

**BY**

**FAUZI ABDUL ROHIM**

**2101182035**



IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE  
OF  
MASTER OF ELECTRICAL ENGINEERING  
IN  
THE SCHOOL OF ELECTRICAL ENGINEERING

**TELKOM UNIVERSITY  
2021**

# ABSTRACT

**Keywords:** *Autonomous robot, Q-Learning, Neuro Fuzzy, Behavior-Based Robot*

## ABSTRAK

Menavigasi robot otonom di lingkungan yang tidak terkendali itu menantang karena membutuhkan seperangkat subsistem yang bekerja secara bersama. Agar dapat beradaptasi dengan navigasi di lingkungan yang tidak dikenal, mobile robot harus memiliki kemampuan yang cerdas, seperti kognisi lingkungan, keputusan perilaku, dan pembelajaran. Robot kemudian akan menavigasi antara hambatan ini tanpa menabrak dan mencapai titik tujuan tertentu.

Penelitian ini koncern pada navigasi otomatis pada mobile robot dari posisi awal menuju posisi tujuan. Untuk memecahkan beberapa sub-masalah yang terkait dengan navigasi otomatis di lingkungan yang tidak terkendali. Simulasi Monte Carlo dilakukan untuk mengevaluasi kinerja algoritma untuk menunjukkan dalam kondisi apa algoritma berkinerja lebih baik dan lebih buruk. Menggunakan kerangka kerja deep reinforcement learning, untuk mendapatkan pemetaan posisi untuk mengoptimalkan aksi pada robot mobile. Reinforcement learning memerlukan jumlah sampel pelatihan yang banyak, yang mana sangat sulit untuk dapat langsung diaplikasikan pada skenario navigasi robot mobile secara nyata. Untuk memecahkan masalah dilatih di lingkungan simulasi Gazebo platform middleware Robot Operating System (ROS), diikuti dengan penerapan pelatihan deep Q-Learning pada skenario navigasi mobile robot dunia nyata.

Diharapkan melalui simulasi dan percobaan, efektivitasnya dapat terbukti. Setelah dilatih dengan metode ini, robot dapat bergerak dengan aman mencapai target, navigasi di lingkungan yang tidak diketahui tanpa demonstrasi sebelumnya. Selain itu, evaluasi kuantitatif dan kualitatif yang mendalam dari metode ini disajikan dengan perbandingan dengan metode perencanaan jalur normal yang didukung oleh peta lingkungan global sebelumnya.

**Kata kunci:** Autonomous Robot Mobile, Q-Learning, Gazebo, Robot Operating System.

# CONTENTS

<b>ABSTRACT</b>	<b>ii</b>
<b>ABSTRAK</b>	<b>iii</b>
<b>CONTENTS</b>	<b>iv</b>
<b>LIST OF TABLES</b>	<b>v</b>
<b>LIST OF FIGURES</b>	<b>vi</b>
<b>LIST OF TERMS</b>	<b>vii</b>
<b>LIST OF NOTATIONS</b>	<b>viii</b>
<b>1 INTRODUCTION</b>	<b>1</b>
1.1 Latar belakang masalah . . . . .	1
1.2 Rumusan masalah . . . . .	2
1.3 Tujuan . . . . .	3
1.4 Hipotesis . . . . .	3
1.5 Scope of Work . . . . .	3
1.6 Research Method . . . . .	4
1.7 Metodologi . . . . .	4
1.7.1 Studi Literatur . . . . .	5
1.7.2 Requirement identification . . . . .	5
1.7.3 Perancangan dan Pembuatan Sistem . . . . .	6
1.7.4 Implementasi dan Pengujian Sistem . . . . .	6
1.7.5 Pengambilan Data dan Analisa Sistem . . . . .	6
1.7.6 Kesimpulan dan Saran . . . . .	6
1.8 Schedule . . . . .	6
<b>2 REVIEW OF LITERATURE AND STUDIES</b>	<b>7</b>
2.1 Autonomous Trash Collector Robot . . . . .	7
2.2 Autonomous Mobile Robot . . . . .	7
2.3 Mobile Robot Control Architectures . . . . .	8
2.4 Reinforcement Learning . . . . .	9
2.4.1 Q learning in robotics . . . . .	10
<b>3 RESEARCH METHODOLOGY</b>	<b>11</b>
3.1 Model system dan Skenario . . . . .	11
3.2 Skenario Simulasi . . . . .	12
3.3 Skenario Pengambilan data . . . . .	12
3.4 Skenario Analisis . . . . .	13
<b>BIBLIOGRAPHY</b>	<b>14</b>

# LIST OF TABLES

## LIST OF FIGURES

1.1	Metode Penelitian . . . . .	5
1.2	Usulan Penelitian . . . . .	6
2.1	Autonomous Navigation System . . . . .	8
2.3	Diagram Interaksi antara Learner dan Environment . . . . .	10
3.1	Behavior Based Kontrol pada Autonomous Trash Collector Robot . . . . .	12

## LIST OF TERMS

Terms	Definition
Classes	Number of individual in biometrics data
Sample	Number of images can be used to represent population in a class.
N	Natural numbers
R	Real numbers
AGV	Automated guided vehicle
AMR	Autonomous mobile robot
ANN	Artificial neural network
AUV	Autonomous underwater vehicle
BPNN	Backpropagation neural network
CNN	Convolutional neural network
DP	Dynamic programming
FFNN	Feedforward neural network
GPI	Generalized policy iteration
IRL	Inverse reinforcement learning
LfD	Learning from demonstration
MAP	Maximum a posteriori
MC	Monte Carlo
MDP	Markov decision process
ML	Maximum likelihood
NN	Neural network
NNQL	Neural network based Q-learning
PI	Policy iteration
POMDP	Partial observable Markov decision process
RL	Reinforcement learning
SGD	Stochastic gradient descent
TD	Temperal difference
UAV	Unmanned aerial Vehicle
UGV	Unmanned ground vehicle
VI	Value iteration
...	...

## LIST OF NOTATIONS

Symbols	Definition
$\mapsto$	Mapping operator
$(x_i, y_i)$	Data point
$ A $	Cardinality of A
$ p - q $	Absolute of $p - q$
$\mathbb{R}$	Sets of real number



# CHAPTER 1

## INTRODUCTION

### 1.1 Latar belakang masalah

Setiap tahunnya, lebih dari 2 (dua) juta ton plastik dibuang ke sungai dan akhirnya hanyut ke laut [1], sehingga sistem pembuangan sampah menjadi sektor yang cukup krusial [1][2]. Metode pengelolaan sampah secara manual menjadi metode yang sering digunakan untuk mengatasi krisis tersebut[3]. Namun, terdapat beberapa masalah pada pengelolaan sampah secara manual, seperti keselamatan para tenaga kerja, tidak dapat menjangkau daerah terpencil, tingginya biaya pengoprasian, dan lainnya[3]. Autonomous Trash Collector Robot (ATCR) menjadi solusi untuk mengatasi permasalahan tersebut, karena dapat mengurangi resiko kecelakaan, dapat menjangkau daerah terpencil, dan dapat melakukan pekerjaan secara berulang [3, 4]. Autonomous mobile Robot telah banyak dikembangkan pada beberapa penelitian, seperti: Robot pembersih dinding [5], pembersih air[6], dan pembersih lantai[4, 7, 8].

Navigasi otomatis pada robot diperlukan untuk pemetaan lingkungan agar robot dapat berjalan dengan baik. Data dari sensor robot dapat dipetakan dan digunakan oleh robot untuk navigasi dan perencanaan pergerakan. selain itu data dari sensor juga digunakan untuk estimasi posisi robot yang dibutuhkan saat pemetaan lingkungan sekitarnya.

Arsitektur behavior based robotic merupakan suatu sistem kendali yang tidak berbasiskan model, karena memiliki struktur behavior yang bekerja bersama secara paralel, bersamaan dan asinkronus (Brooks, 1986). Pada pendekatan ini, sistem diuraikan menjadi beberapa modul yang masing-masingnya bertanggung jawab untuk melakukan satu perilaku (behavior). Tiap behavior mengandung jalur lengkap mulai dari sensing sampai aksi. Semua modul yang mewakili satu behavior bekerja bersama-sama[2]. Semakin banyak tugas sistemnya semakin kompleks, sehingga dapat menimbulkan konflik antar behavior. Oleh karena itu, dikembangkan metode koordinasi antar behavior. Perhatian utama diberikan pada dua pendekatan mekanisme koordinasi, yaitu competitive/arbitrer dan cooperative/command fusion. Pada metode Koordinator kompetitif memastikan kekokohan pengontrol, hanya satu behavior yang diijinkan memberikan sinyal kendali. Sedangkan koordinasi kooperatif menggabungkan semua keluaran behavior yang ada dan menentukan kinerja lintasan robot. Dua lapisan utama dalam skema lapisan deliberatif yang membagi misi robot menjadi satu set tugas, dan lapisan berbasis perilaku yang bertanggung jawab untuk menyelesaikan tugas-tugas ini. tesis hanya berpusat pada lapisan berbasis perilaku. Pendekatan koordinasi perilaku diusulkan. Fitur utamanya adalah koordinasi perilaku yang hibrid, antara pendekatan kompetitif dan kooperatif. Selain arsitektur yang tepat, juga diperlukan mekanisme pembelajaran yang tepat pada robot untuk mengatasi hal hal tak terduga.

Penguatan learning memungkinkan agen untuk memilih kebijakan perilaku yang optimal melalui pelatihan interaksi trial-and-error dengan mereka lingkungan [1]. Baru-baru ini, penguatan belajar memiliki sukses besar dalam banyak tugas seperti video game dan kontrol simulasi [2]. Juga, berbagai yang luar biasa masalah robotika mungkin akan diungkapkan secara alami sebagai salah satu dari penguatan belajar. Tidak seperti pembelajaran mesin lainnya metode seperti yang diawasi dan pembelajaran tidak diawasi, penguatan belajar menyediakan umpan balik dalam hal lingkungan perilaku antar-bereaksi

fungsi sehingga untuk mengukur kinerja satu langkah robot. Dalam proses penguatan belajar, agen akhirnya akan menyimpulkan kebijakan perilaku optimal setelah menjelajahi lingkungan. Penguatan belajar (RL) teknik dapat belajar tindakan yang tepat dari keadaan-keadaan lingkungan. Di dalam proses interaksi antara agen dan eksternal lingkungan, Agen berulang kali belajar melalui percobaan dan kesalahan, berkaitan informasi lingkungan, dan terus-menerus mengoptimalkan strategi aksi agen [3]. Ini optimisasi metode memberikan RL keputusan yang sangat baik- membuat kemampuan [4]. Saat ini, penguatan belajar telah berhasil diterapkan di jalur perencanaan mobile robot [3]. Algoritma Penguatan belajar memecahkan masalah keputusan sekuensial yang diajukan sebagai Proses keputusan Markov (MDPs), mempelajari kebijakan dengan membiarkan agen mengeksplorasi efek dari tindakan yang berbeda dalam situasi yang berbeda ketika mencoba untuk memaksimalkan sinyal hadiah. RL telah berhasil diterapkan ke berbagai skenario. Belajar dari demonstrasi adalah pendekatan ke robot / agen belajar bahwa mengambil sebagai masukan demonstrasi dari seorang manusia dalam rangka untuk membangun aksi atau tugas model.

Metode perencanaan jalur global (Global Path Planning) meliputi metode ruang konfigurasi [1], metode medan potensial (potential field method)[2], diagram Voronoi tergeneralisasi [3], dan metode pencarian graf [4]. Metode-metode ini telah dilakukan secara off-line di lingkungan yang benar-benar dikenal. Namun, metode ini tidak cocok untuk navigasi di lingkungan yang kompleks dan berubah secara dinamis di mana hambatan yang tidak diketahui mungkin terletak di jalur yang direncanakan secara apriori. Oleh karena itu, diperlukan perencanaan jalur lokal berbasis sensor, yang disebut dengan penghindaran rintangan, yang dilakukan secara on-line dalam navigasi. Perencanaan jalur lokal memanfaatkan informasi yang disediakan oleh sensor seperti sensor ultrasonik, penglihatan, pencari jarak laser, sensor jarak, dan sakelar bumper. Meskipun banyak algoritma Q-learning kontinyu diusulkan, namun hanya beberapa yang diterapkan pada aplikasi robot real untuk sistem navigasi autonomous mobile robot[20].

Pada Penelitian ini akan membuat arsitektur perilaku hierarkis karena dapat memisahkan tugas merancang perilaku (behavior) primitif atau mengadaptasi strategi yang diawasi yang berkoordinasi dengan perilaku belajar (learning behavior) [9]. Terinspirasi dari makalah [4] dan makalah[10–13], penelitian thesis ini bertujuan untuk merancang *Autonomous Trash Collector Robot* (ATCR) menggunakan *Deep Reinforcement Learning* [14]. Robot dilengkapi dengan sensor dan sistem navigasi untuk mendeteksi posisi robot. Selain itu, ATCR juga dilengkapi sistem kendali pada motor[15]. Pada penelitian ini akan dirancang robot platform middleware ROS dengan arsitektur behavior based control. ROS mempunyai banyak contoh source code yang disediakan, termasuk sensor dan perangkat yang akan menambah fungsionalitas baru pada robot seperti navigasi otonom dan persepsi dari sensor. Sistem operasi Robot (ROS) dan Gazebo digunakan untuk mensimulasikan lingkungan virtual. Kemudian juga akan ditambahkan Q learning sebagai mekanisme pembelajaran robot. Robot akan melakukan navigasi otonom untuk menghindari halangan dan menemukan target.

## 1.2 Rumusan masalah

- Bagaimana merancang sistem lokalisasi dan pemetaan untuk referensi pergerakan ATCR?
- Bagaimana mengimplementasikan Reinforcement learning untuk membantu meningkatkan

kemampuan perencanaan lokal robot menghindari halangan baik yang dipetakan sebelumnya ataupun tidak?

### 1.3 Tujuan

- Merancang sistem mobil robot menggunakan Navigation stack ROS dengan sistem pemetaan robot dapat mengelilingi ruangan untuk menghindari dan menemukan target.
- Mengimplementasikan algoritma Q-Learning sorta mengintegrasikannya dengan ROS Navigation Stack pada robot mobil

### 1.4 Hipotesis

Bagian implementasi perilaku robot mengusulkan penggunaan algoritma pembelajaran untuk mempelajari pemetaan internal antara keadaan lingkungan dan tindakan robot. Reinforcement Learning sebagai teori pembelajaran yang cocok untuk pembelajaran robot. Penerapannya secara online dan tidak diperlukannya informasi sebelumnya tentang lingkungan adalah keuntungan yang paling penting. Selain itu, disajikan algoritma pembelajaran Q, yang secara khusus memadai untuk kemampuannya dalam pembelajaran di luar kebijakan. Kelemahan yang paling penting adalah masalah generalisasi. Algoritma Pembelajaran Penguatan didasarkan pada representasi diskrit dari keadaan dan ruang tindakan. Ketika algoritme ini diterapkan pada variabel kontinu, seperti yang dibutuhkan sebagian besar aplikasi robotika, diskritisasi variabel menyebabkan sejumlah besar status dan waktu pembelajaran yang lama. Generalisasi tersebut membuat penerapan Reinforcement Learning dalam robotika menjadi tidak praktis. Namun, beberapa teknik disajikan yang mencoba memecahkan masalah ini. mengusulkan algoritma Pembelajaran Penguatan yang dirancang untuk diterapkan pada robotika. Tujuan dari algoritma Q-Learning adalah untuk mempelajari perilaku robot. Ini didasarkan pada algoritma pembelajaran Q dan memecahkan masalah generalisasi dengan menggunakan Jaringan Syaraf Tiruan dan database sampel pembelajaran yang paling representatif.

diperlukan perencanaan jalur lokal berbasis sensor, yang disebut dengan penghindaran rintangan, yang dilakukan secara on-line dalam navigasi. Perencanaan jalur lokal memanfaatkan informasi yang disediakan oleh sensor seperti sensor ultrasonik, penglihatan, pencari jarak laser, sensor jarak, dan sakelar bumper.

### 1.5 Scope of Work

1. Lokasi pengujian dilakukan dan menggunakan lingkungan buatan.
2. Terdapat obyek dan halangan untuk dihindari yang disimpan secara acak.
3. Robot hanya berjalan ke titik tujuan yang ditetapkan. Beberapa fungsi lanjutan seperti mengambil objek, transportasi objek, dan fungsi-fungsi lain diabaikan.
4. Proses training algoritma dilakukan melalui simulasi kemudian model hasil learning ditransfer ke robot.
5. Jenis robot yang digunakan adalah non platform mobile robot yang dirancang sendiri.

6. Pada perancangan sistem tidak dilakukan perhitungan terhadap parameter motor dc seperti gaya gesek dan inersia.
7. Sistem kendali gerak hanya mengontrol kecepatan dan arah putar motor dc pada mobile robot.
8. Sistem navigasi mobile robot ini beroperasi pada lintasan yang datar.
9. sensor ultrasonik digunakan untuk menentukan jarak aman robot dalam melakukan pergerakan

## 1.6 Research Method

Tujuan dari model Q-Learning mobile robot menemukan jalur optimal untuk tujuan dari lokasi tanpa bertabrakan dengan hambatan yang diberikan peta. Untuk menyelesaikan model Q-Learning yang mengambil data sensorik diberikan ke robot sebagai masukan (atau pengamatan), keluaran dari model adalah instruksi untuk gerakan robot, seperti bergerak maju, belok kiri, belok kanan, dan pindahkan kebelakang, yang dijalankan dengan memodifikasi posisi roda robot. mobile virtual robot dan lingkungan untuk tugas navigasi menggunakan Gazebo [11], yang menyediakan sebuah 3D multi-robot lingkungan dalam format open-source. Seperti disebutkan dalam sebelumnya, Q-Learning berbasis pendekatan memiliki beberapa keterbatasan, termasuk membutuhkan sejumlah besar data untuk melatih model dan memproduksi hasil yang sulit untuk ditafsirkan. Sebagai pembelajaran progresses, model belajar dengan jarak antara robot dan hambatan atau tujuan. Jika robot bertabrakan dengan hambatan, model belajar tentang hambatan yang menerima hukuman (seperti, hadiah negatif) berdasarkan jarak antara setiap sensor pada robot dan hambatan.

Model DRL tampil baik dalam kasus di mana label sebenarnya tidak praktis bisa dicapai, tetapi dapat dihasilkan melalui implementasi dinamika sistem yang sebenarnya, karena hasil dari aksi di keadaan bagian yang berbeda sangat tidak diperkirakan. Selain itu, DRL baru-baru ini diterapkan dalam domain berbasis gambar seperti video game. Seperti ranah memiliki beberapa aturan yang diberikan lingkungan, oleh karena itu model DRL dipelajari menggunakan piksel permainan dan sebuah nilai hasil cetak skor berdasarkan aturan. Tidak banyak penelitian yang diterapkan pada tugas navigasi memakai DRL.

## 1.7 Metodologi

Adapun tahapan pada penelitian adalah sebagai berikut:

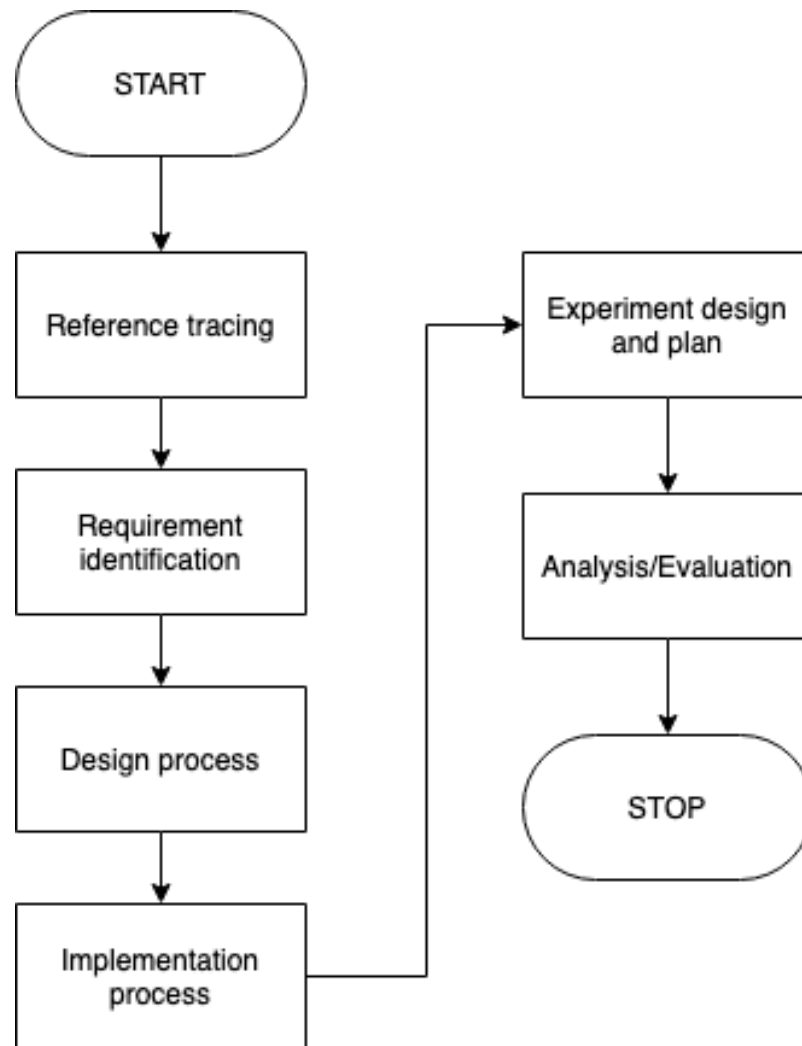


Figure 1.1

### 1.7.1 Studi Literatur

Studi literatur tentang : navigation mobile robot, Behavior based Robotic, Reinforcement Learning, Q-learning, Neuro Fuzzy Q-learning, metode Simulasi Mobil Robot dari penelitiain sebelumnya. seperti pada jurnal, karya tulis, serta buku-buku dan literatur referensi yang berkaitan dengan penelitiain.

### 1.7.2 Requirement identification

Pada tahap ini dilakukan analisis yang mencakup kebutuhan untuk melakukan penelitian, kebutuhan yang dianalisis dibagi menjadi analisa data dan juga analisa kebutuhan sistem. Analisis dilakukan agar sistem yang dibangun dapat berjalan sesuai dengan rancangan yang sebelumnya sudah ditentukan.

### 1.7.3 Perancangan dan Pembuatan Sistem

Melakukan desain robot dari perangkat lunak dan perangkat keras setelah mempelajari studi literatur dan state of the art sistem navigasi mobile robot. serta membuat struktur algoritma reinforcement learning, berdasarkan pada permasalahan topik penelitian, serta melakukan simulasi pada sistem yang dirancang.

### 1.7.4 Implementasi dan Pengujian Sistem

Melakukan implementasi dan pengujian pada sistem perangkat keras, pelatihan algoritma reinforcement learning yang digunakan, dan menguji perangkat lunak yang telah dirancang, lalu mengevaluasi sistem berdasarkan topik Penelitian.

### 1.7.5 Pengambilan Data dan Analisa Sistem

proses pengambilan data dan hasil sesuai dengan tujuan penelitian, analisa sistem dari hasil dari pengujian sistem.

### 1.7.6 Kesimpulan dan Saran

## 1.8 Schedule

Penelitian ini direncanakan akan diselesaikan dalam tempo empat bulan. Rencana tersebut dijabarkan dalam tabel sebagai berikut:

Activity	WEEK I				WEEK II				WEEK III				WEEK IV			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Reference tracing																
Requirement identification																
Design process																
Implementation process																
Experiment design and plan																
Analysis/ Evaluation																

Figure 1.2: Usulan Penelitian

## CHAPTER 2

### REVIEW OF LITERATURE AND STUDIES

#### 2.1 Autonomous Trash Collector Robot

Beberapa penelitian terkait Autonomous Robot untuk pengolahan sampah telah banyak dilakukan pada beberapa penelitian. Salah satunya pada penelitian yang dilakukan oleh Aditya P. P. Prasety, etc [16], di mana pada penelitian tersebut Autonomous Robot dibuat menggunakan lengan manipulator untuk mengambil sampah. Posisi robot dikendalikan berdasarkan sensor ultrasonic dan sistem navigasi. Robot tersebut mampu membersihkan dua jenis sampah sebanyak 40 kali. Penelitian lain serupa juga dilakukan oleh Muhammad Abbas Khan, et al [3], dimana pada penelitian tersebut Robot Semi-autonomous dibuat untuk mengambil sampah berdasarkan perintah dari perangkat Smartphone melalui bluetooth. Robot juga dilengkapi sensor ultrasonic sehingga mampu mendeteksi posisi sampah. Pada Referensi [4], sebuah Autonomous Robot dibuat untuk mengambil sampah yang beroperasi di rumput. Robot mampu mendeteksi sampah secara otomatis dan akurat dengan menggunakan algoritma Deep Neural Network [10]. Robot tersebut dilengkapi sensor ultrasonic[11] dan sistem navigasi [12]. Hasil pengujian menunjukkan bahwa robot mampu mendeteksi sampah dengan akurasi 95%. Pada makalah[13], Autonomous Robot dirancang menggunakan algoritma Convolutional Neural Network (CNN) untuk mendeteksi berbagai jenis sampah, seperti kaleng, botol plastik, dan kotak makan siang secara otomatis. Sistem robot mampu mendeteksi sampah secara outdoor. Berdasarkan data hasil pengujian[13], robot tersebut mampu mendeteksi sampah dengan tingkat kepresisian sebesar 95,6% dan tingkat akurasi sebesar 96,8% .

Pada penelitian ini akan dirancang sebuah autonomous mobile robot berjenis mobile robot beroda dengan menggunakan metode differential drive untuk pergerakan robot menuju obyek. Robot menggunakan kamera untuk mendapatkan citra obyek (obstacle dan target), mendapatkan path planning dari hasil citra bola untuk pergerakan posisi robot.

#### 2.2 Autonomous Mobile Robot

Robot yang sepenuhnya otonom dapat memperoleh informasi tentang lingkungan, bekerja untuk waktu yang lama tanpa campur tangan manusia, memindahkan semua atau sebagian dari dirinya sendiri di seluruh lingkungan operasinya tanpa bantuan manusia, menghindari situasi yang berbahaya bagi manusia, properti. Robot otonom juga dapat belajar atau mendapatkan pengetahuan baru seperti menyesuaikan metode baru untuk menyelesaikan tugasnya atau beradaptasi dengan lingkungan yang berubah. Oleh karena itu, robot bergerak perlu memiliki kemampuan otonomi dan kecerdasan, dan untuk merancang algoritme yang memungkinkan robot berfungsi secara mandiri di lingkungan yang tidak terstruktur, dinamis, sebagian dapat diamati, dan tidak pasti menimbulkan tantangan bagi para peneliti untuk menangani masalah kunci. Masalah seperti ketidakpastian (baik dalam penginderaan dan tindakan), keandalan, dan respons waktu nyata.

Di masing-masing domain aplikasi mobile robot, mobilitas hampir tidak ada gunanya tanpa kemampuan bernavigasi. Gerakan acak, yang tidak memerlukan kemampuan navigasi, mungkin berguna untuk pengawasan atau operasi pembersihan tertentu, tetapi untuk sebagian besar aplikasi ilmiah atau industri robot bergerak, kemampuan untuk bergerak

dengan cara yang bertujuan diperlukan. Oleh karena itu, navigasi otonom memainkan peran kunci dalam keberhasilan robot, dan juga dasar untuk teknologi relatif robot bergerak otonom.

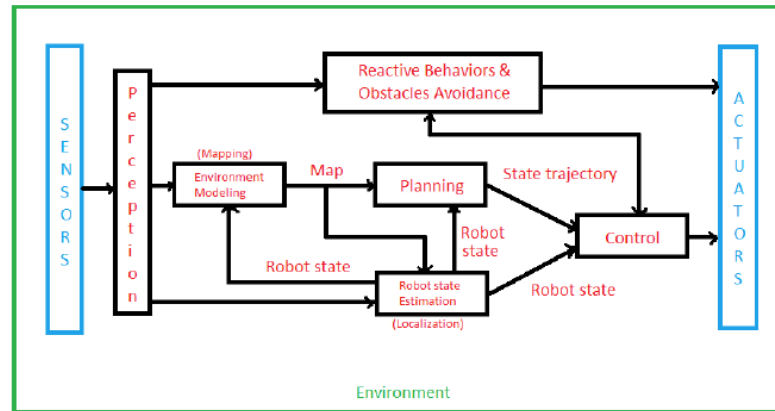


Figure 2.1: Autonomous Navigation System

Tugas navigasi mobile robot mengacu pada rencana jalur dengan penghindaran rintangan ke tujuan tertentu dan untuk melaksanakan rencana ini berdasarkan pembacaan sensor. Navigasi robot seluler secara kasar mencakup enam kompetensi yang saling terkait berikut ini 2.1.

1. Perception: untuk memperoleh dan menginterpretasikan informasi sensorik;
2. Exploration: strategi yang memandu robot untuk memilih arah selanjutnya;
3. Mapping: untuk membangun representasi spasial atau model lingkungan dengan menggunakan informasi sensorik yang dirasakan;
4. Localization: strategi untuk memperkirakan posisi robot dalam peta spasial yang terjadi secara bersamaan untuk kontrol navigasi;
5. Path planning: strategi mencari jalur menuju lokasi tujuan yang optimal atau tidak;
6. Path execution: untuk menentukan dan menyesuaikan aksi motorik dengan perubahan lingkungan, juga termasuk penghindaran rintangan.

Sebuah robot membutuhkan mekanisme yang memungkinkan untuk bergerak bebas di lingkungan, yaitu, harus mampu mendeteksi dan bereaksi terhadap situasi. Ini adalah sensor robot yang memainkan peran seperti mata robot, dan robot tahu di mana itu atau bagaimana caranya ke suatu tempat, atau untuk dapat menjelaskan ke mana ia pergi. Sensornya bisa fleksibel dan mobile untuk mengukur jarak yang telah ditempuh roda sepanjang tanah, untuk mengukur perubahan inersia dan struktur eksternal di lingkungan.

## 2.3 Mobile Robot Control Architectures

Arsitektur kontrol robot bergerak melibatkan proses mengambil informasi tentang lingkungan melalui sensor robot, memprosesnya seperlunya untuk membuat keputusan tentang



bagaimana bertindak, dan pelaksanaan tindakan di lingkungan. Arsitektur kontrol tradisional (deliberatif) diturunkan dari paradigma kecerdasan buatan (AI) tradisional, di mana perencana pusat menggabungkan semua pembacaan sensor, membangun model dunia, merencanakan tindakan selanjutnya, dan akhirnya mengarahkan robot. Gambar 2.2 menggambarkan arsitektur tersebut.

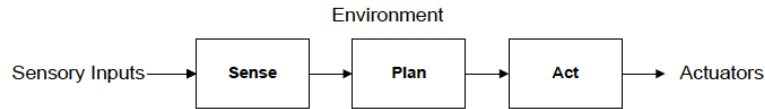


Figure 2.2: Traditional sense-plan-act architecture

Robot awal seperti Shakey [Nilsson, 1984] mengadopsi jenis arsitektur ini, yang pada intinya berusaha mengatasi ketidakpastian lingkungan dengan menciptakan model dunia. Musyawarah mengacu pada berpikir keras, dan didefinisikan sebagai perhatian dalam mengambil keputusan dan tindakan [Nattharith, 2010]. Sistem kontrol umumnya diatur menggunakan dekomposisi fungsional dari proses pengambilan keputusan, yang terdiri dari beberapa modul untuk pemrosesan sensorik, pemodelan dan perencanaan, penilaian nilai, dan eksekusi [Brooks, 1986]. Dekomposisi fungsional seperti itu memungkinkan operasi kompleks untuk dilakukan, tetapi menyiratkan independensi berurutan yang kuat antara modul pengambilan keputusan. Arsitektur ini mampu bekerja dengan sukses dalam lingkungan yang terstruktur. Misalnya, jika ada cukup waktu untuk menghasilkan rencana dan model dunia akurat, pendekatan ini memungkinkan robot menghasilkan tindakan terbaik untuk situasi tertentu.

Namun, seperti arsitektur cenderung gagal dalam lingkungan yang tidak terstruktur atau bahkan dalam lingkungan yang terstruktur secara longgar karena ketidakmampuan mereka untuk beradaptasi dengan lingkungan. Selain itu pendekatan tersebut terbatas dalam kegunaannya karena kurangnya reaktivitas waktu nyata, dan mungkin sepenuhnya gagal jika salah satu bagian gagal. Oleh karena itu arsitektur murni yang disengaja tidak lagi digunakan untuk sebagian besar robot bergerak fisik yang bekerja di lingkungan dunia nyata yang kompleks dan berubah secara dinamis [Peng, 2004; Natharith, 2010].

## 2.4 Reinforcement Learning

Reinforcement Learning (RL) adalah salah satu jenis proses learning dari Machine Learning (ML), yang melakukan pendekatan belajar dengan trial and error untuk mencapai tujuan. Oleh karena itu RL memerlukan reward dari lingkungannya sebagai pengganti data respon input dan outputnya. Reward digunakan untuk menguji state lingkungan, pengumpulan jumlah reward secara maksimal sangat penting karena reward menjadi signal feedback dalam proses learning.

Ada beberapa elemen utama yang muncul pada RL, yaitu learner, environment, dan reward. Learner disebut sebagai algoritma RL, karena learner akan berinteraksi dengan lingkungan (environment). Learner akan mempelajari segala kondisi state lingkungannya dan segera berinteraksi dengan lingkungannya. Cara learner berinteraksi dengan lingkungannya adalah dengan memberikan aksi kepada lingkungannya (environment). Kemudian environment akan berinteraksi pada aksi yang diberikan learner dengan state baru dan

reward. Reward adalah suatu nilai yang dibangkitkan/diberikan oleh fungsi reinforcement yang menguji current state dan aksi terakhir. Berikut adalah gambar diagram interaksi sistem dari RL:

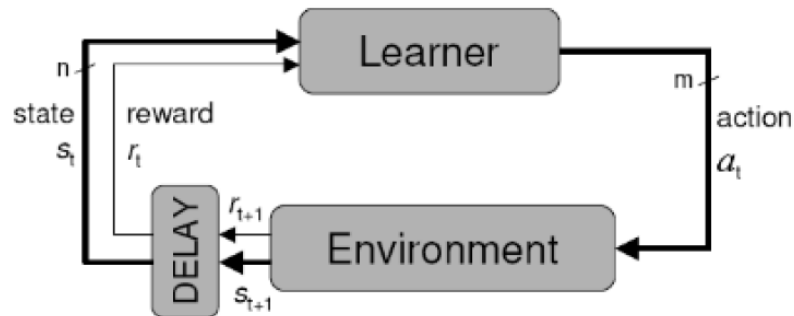


Figure 2.3: Diagram Interaksi antara Learner dan Environment

#### 2.4.1 Q learning in robotics

Q-learning adalah salah satu model-free learning dalam Teknik RL. Penggunaan Q-learning untuk mencari nilai optimal dari Q-value (action value function). Selama proses learning nilai Q akan terus terupdate, dari nilai Q yang lama menjadi nilai Q yang baru. Setiap perubahan nilai Q bergantung pada pemilihan aksi yang ada pada service.

Q-learning mempunyai cara kerja dengan melakukan evaluasi pada setiap episode, proses dalam satu episode dikatakan berakhir jika agent sudah mencapai titik goal state, dan setiap melakukan action akan mempengaruhi nilai Q. Nilai Q dijadikan “brain” oleh agent selama proses learning. Untuk lebih jelasnya berikut adalah gambar mengenai algoritma Q-Learning:

## CHAPTER 3

# RESEARCH METHODOLOGY

Desain Autonomous Trash Collector Robot memerlukan beberapa hardware diantaranya single board computer, embedded board, actuator, baterai dan sensor kamera. Actuator yang digunakan dalam penelitian ini yaitu dinamo motor DC 12V yang berguna sebagai penggerak dengan tegangan arus yang searah dengan kumparan medan agar diubah menjadi gerak mekanik. Embedded board yang digunakan adalah driver motor L298N yang berfungsi untuk menjalankan actuator, mengatur arah dan kecepatan pada sebuah actuator. Single Board Computer (SCB) yang berfungsi sebagai otak sehingga mengatur sistem yang ada dan komunikasi sistem dalam robot. Sensor kamera di sini menggunakan 1 buah sensor kamera (monocular camera), kamera di sini dibutuhkan sebagai sensor utama pada robot yang berfungsi sebagai alat bantu robot untuk melakukan tugasnya dan terintegrasi langsung pada SCB robot.

Behavior based robotic merupakan sistem kontrol berbasis pada level kompetensi dan koordinator tertentu. Maka dibutuhkan beberapa tahapan perancangan algoritma untuk diterapkan pada Autonomous Trash Collector Robot (ATCR). Tahapan menghindari halangan, mencari target, dan menuju target. Tahapan perilaku tersebut dilakukan pembelajaran dengan menggunakan algoritma Neuro Fuzzy Q-Learning (NFQL). Robot dikendalikan secara langsung pada pengendali tingkat bawah (low level controller) berupa kontroler P yang berfungsi mengendalikan aktuatur motor driver dan motor DC. Robot juga akan mengirimkan sinyal persepsi, melalui sensor kamera pada pengendali tingkat atas (High Level Controller). Kemudian sinyal tersebut menjadi masukan untuk mengaktifkan behavior koordinasi. Output behavior coordination mengaktifkan low level controller agar mengendalikan robot.

### 3.1 Model system dan Skenario

Robot akan dikendalikan secara langsung oleh low level controller (berupa kontroler P) yang berfungsi untuk mengendalikan pergerakan robot dengan beberapa motor DC yang ada. Robot juga akan mengirim sinyal persepsi (melalui sensor – sensor yang dimilikinya) pada high level controller. Kemudian sinyal tersebut menjadi masukan stimulus yang akan mengaktifkan behavior – behavior tertentu (yang dikoordinasi oleh bagian behavior coordination). Output dari behavior coordination diberikan pada low level controller untuk pengendalian robot. Diagram Blok Keseluruhan sistem robot seperti pada Gambar 3.1 berikut ini.

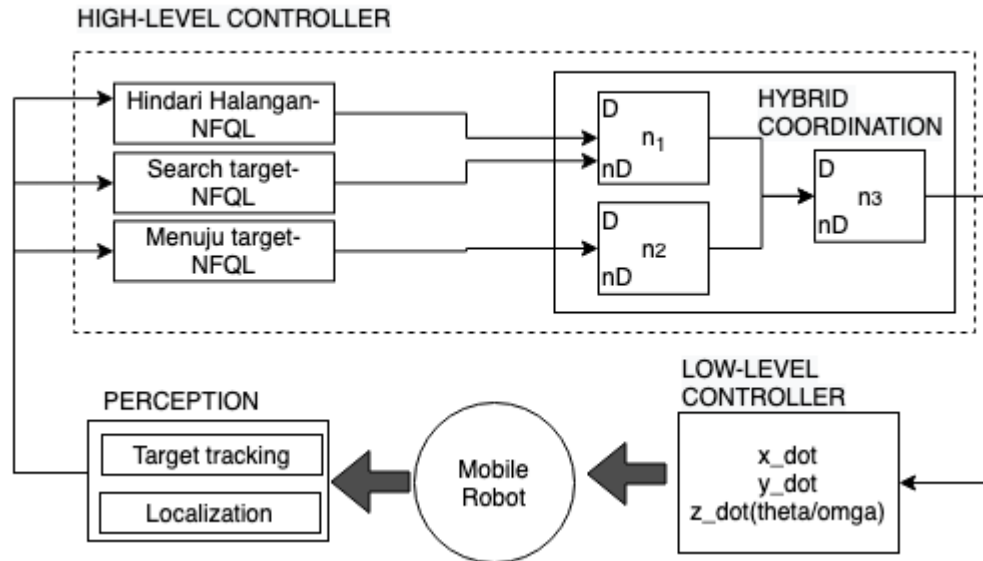


Figure 3.1: Behavior Based Kontrol pada Autonomous Trash Collector Robot

### 3.2 Skenario Simulasi

Perancangan behavior pada ATCR:

1. Menghindar halangan dan Mencari target

Tujuannya adalah menghindari setiap halangan yang dideteksi oleh sensor. Behavior ini belajar menggunakan Q-Learning. Dengan sensor, robot berupaya mencari posisi target. Selama tidak ada halangan, tidak tabrakan, behavior ini akan diaktifkan penuh. Jika target tidak terdeteksi, aktivasi bernilai 1, sehingga robot berusaha mencari target.

2. Menuju target

Behavior ini memiliki tingkat aktivasi paling rendah. Dia akan aktif penuh jika behavior-behavior sebelumnya tidak aktif. Setelah target didapatkan oleh behavior “mencari target”, dengan behavior ini, robot akan menuju target.

### 3.3 Skenario Pengambilan data

Untuk memastikan tingkat akurasi tertentu, pada thesis ini akan menggunakan “Confusion Matrix” untuk membandingkan klasifikasi prediksi dengan klasifikasi sebenarnya, teknik yang biasa digunakan ketika membandingkan pengklasifikasi pembelajaran seperti jaringan saraf, tidak hanya mendapatkan akurasi untuk dihitung, tetapi juga sensitivitas dan spesifisitas solusi.

Sensitivitas solusi adalah kemampuannya untuk menghindari mengabaikan hasil positif, sedangkan spesifisitas adalah kemampuan solusi untuk mengklasifikasikan negatif yang benar dengan benar. Kedua ukuran statistik berguna saat mengukur kemampuan detektor objek untuk tujuan mengemudi otonom – nilai sensitivitas rendah dapat mencegah sistem deteksi objek mengidentifikasi rintangan pada waktunya untuk menghindari tabrakan.

Oleh karena itu, solusi harus memiliki sensitivitas yang sangat tinggi untuk memastikan tingkat keandalan praktis.

Selain itu, solusi harus memiliki kekhususan yang signifikan sehingga tidak salah mengklasifikasikan objek dari latar belakang. Hal ini terutama berlaku dalam hal mendeteksi obyek dan rintangan, mungkin memiliki efek yang tidak diinginkan saat diproses oleh sistem navigasi (seperti mengemudi dengan kecepatan sangat tinggi jika salah mendeteksi obyek). Selain itu, solusi harus cukup cepat untuk dilakukan secara real time.

### 3.4 Skenario Analisis

Untuk menguji algoritma kontrol pada ATCS, dilakukan beberapa tahapan:

1. Pembuatan model 3D Robot. Model 3D diperlukan untuk keperluan simulasi algoritma kontrol robot menggunakan program 3D. Simulator robot 3D yang dapat digunakan untuk mendesain model 3D robot, membuat script pemrograman dan melakukan simulasi robot seperti kondisi real robot.
2. Simulasi Algoritma kontrol pada model robot. Hasil desain algoritma diujicobakan pada simulator robot sebelum diujicobakan pada robot sebenarnya. Simulator ini dibuat menggunakan software simulator dengan pemrograman algoritma kontrol.
3. Implementasi Algoritma pada robot sebenarnya. Setelah simulasi algoritma kontrol pada simulator robot berhasil, algoritma kontrol ditanam pada robot sebenarnya.
4. Pengujian Sistem robot pada medan sebenarnya. Analisa untuk akurasi robot dilakukan dengan membandingkan titik uji dengan titik yang dihasilkan oleh gerakan. Dari selisih kedua titik tersebut akan didapatkan akurasi dari robot. Analisa error pergerakan dilakukan dengan membandingkan bentuk data target dan hasil proses. Dari selisih waktu dan jarak yang dihasilkan akan didapatkan error pergerakan.

## BIBLIOGRAPHY

- [1] H. Othman, M. Iskandar Petra, L. Chandratilak De Silva, and W. Caesarendra, “Automated trash collector design,” *Journal of Physics: Conference Series*, vol. 1444, no. 1, 2020.
- [2] S. Hossain, B. Debnath, A. Anika, M. Junaed-Al-Hossain, S. Biswas, and C. Shahnaz, “Autonomous Trash Collector Based on Object Detection Using Deep Neural Network,” *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, vol. 2019-Octob, pp. 1406–1410, 2019.
- [3] M. A. Khan, “Garbage collector robot,” *Indian Journal of Science and Technology*, vol. 13, no. 20, pp. 2065–2070, 2020.
- [4] J. Bai, S. Lian, Z. Liu, K. Wang, and D. Liu, “Deep Learning Based Robot for Automatically Picking Up Garbage on the Grass,” *IEEE Transactions on Consumer Electronics*, vol. 64, no. 3, pp. 382–389, 2018.
- [5] Houxiang Zhang, Jianwei Zhang, Guanghua Zong, Wei Wang, and Rong Liu, “Sky Cleaner 3: a real pneumatic climbing robot for glass-wall cleaning,” *IEEE Robotics & Automation Magazine*, vol. 13, no. 1, pp. 32–41, 2006.
- [6] F. C. Yuan, H. L. Sun, S. J. Hu, and L. Z. Wang, “Design of cleaning robot for swimming pools,” *2011 International Conference on Management Science and Industrial Engineering, MSIE 2011*, pp. 1175–1178, 2011.
- [7] M. C. Kang, K. S. Kim, D. K. Noh, J. W. Han, and S. J. Ko, “A robust obstacle detection method for robotic vacuum cleaners,” *IEEE Transactions on Consumer Electronics*, vol. 60, no. 4, pp. 587–595, 2014.
- [8] J. Palacín, J. A. Salse, I. Valganón, and X. Clua, “Building a mobile robot for a floor-cleaning operation in domestic environments,” *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 5, pp. 1418–1424, 2004.
- [9] F. Hoffmann, “An Overview on Soft Computing in Behavior Based Robotics,” in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 2715, pp. 544–551, 2003.
- [10] A. Kong, D. Zhang, and M. Kamel, “A survey of palmprint recognition,” *Pattern Recogn.*, vol. 42, pp. 1408–1418, July 2009.
- [11] G. K. Ong Michael, T. Connie, and A. B. Jin Teoh, “Review article: Touch-less palm print biometrics: Novel design and implementation,” *Image Vision Comput.*, vol. 26, pp. 1551–1560, Dec. 2008.
- [12] J.-G. Wang, W.-Y. Yau, A. Suwandy, and E. Sung, “Person recognition by fusing palmprint and palm vein images based on ”laplacianpalm” representation,” *Pattern Recogn.*, vol. 41, pp. 1514–1527, May 2008.
- [13] Y. Arai, R. Miyagusuku, and K. Ozaki, “Development and Testing of Garbage Detection for Autonomous Robots in Outdoor Environments,” 2019.

- [14] K. A. A. Mustafa, N. Botteghi, B. Sirmacek, M. Poel, and S. Stramigioli, "TOWARDS CONTINUOUS CONTROL FOR MOBILE ROBOT NAVIGATION: A REINFORCEMENT LEARNING AND SLAM BASED APPROACH," vol. XLII-2/W13, pp. 857–863, jun 2019.
- [15] H. M. Saputra, A. Nurhakim, and S. Utomo, "Position Control of 1-DOF High-Precision Rotary Table using Adaptive Neuro-Fuzzy Inference System (ANFIS) Controller," *EMITTER International Journal of Engineering Technology*, vol. 7, no. 2, pp. 511–523, 2019.
- [16] A. P. Prasetyo, Rendyansyah, S. D. Siswanti, S. Nurmaini, and Abdurahman, "Garbage Collector Robot (Gacobot) Design for Dry Waste Distribution," *Journal of Physics: Conference Series*, vol. 1500, no. 1, 2020.