（千葉大学学位申請論文）

油圧駆動型6脚ロボットCOMET-IVのLRFによるにマッピングと障害物回避に関する研究

2012年1月

指導教官：野波 健蔵 教授

千葉大学大学院工学研究科

人工システム科学専攻 機械コース

モハマドラザリ ビンダウド

ABSTRACT

Researches on obstacle avoidance based on environment map of unknown environment are not widely applied for walking robots, especially for large scale robots with hydraulically-driven actuators. In contrast, the walking robots are mainly applied to perform specific tasks in a predefined environment. This research aims to improve the capabilities and increase autonomy of the hydraulically-driven hexapod robot COMET-IV by improving mapping technique for unknown environment, obstacles avoidances and leg motion control assistance using a laser range finder (LRF) 3-D point clouds data.

The COMET-IV can be controlled from a remote place by an operator, but the operator has to know the surrounding area of the robot and its current conditions to determine the next walking command while steering a robot from a remote place. Therefore, a map of unknown environment is needed, and it is developed using the *Occupancy Grid Map* (OGM), but the cells are categorized into not only two categories as current existed achievement but multiple categories. On the other hand, for autonomous operation (the scope of this research), the information associated with the map is used as a reference to generate a walking path for robot. Moreover, in order to capitalize the capabilities of the robot, the *Grid-based Walking Trajectory for Legged Robot* (GWTLR) method is proposed to avoid, walk over and cross over an obstacle, including ascend and descend a cliff with support of proposed *Grid-cell Edge Detection* method to analyze the obstacle geometrics concerning the map of an unknown environment. The GWTLR method determines the height of the COMET-IV body, leg swing height and leg stride length, and where the robot should stop before its legs are moved to enable the tasks to be performed without collision with the obstacles. In addition, a proposed *Grid-based Walking Assistant for Legged Robot* (GWALR) method cascaded to the force control and impedance control as a dynamic input reference to increase the robustness for robot walking on unstructured terrain.

Experiment results of the proposed methods show that the trajectory planning can be done autonomously under the unknown environment, and it is also demonstrated to be effective to provide the surrounding environment map to the remote operator. Therefore, the proposed methods were proven to be highly potential to be applied for as a part of the overall system for actual stochastic terrain navigation.

TABLE OF CONTENTS

ABSTRACT ii
ACKNOWLEDGEMENTS iv
DEDICATION v
CHAPTER 1 1
1. Introduction 1
1.1 Background 1
1.2 Legged robot 5
1.3 Robot Navigation 7
1.3.1 Environment Mapping 8
1.3.2 Sensors for mapping 11
1.3.3 Path planning and Trajectory Planning 12
1.3.4 A* Algorithm 12
1.3.5 Growing Obstacles Method 14
1.4 Research Objectives and Methodology 15
1.5 Outline of thesis 16
CHAPTER 2 17
2. COMET-IV System Configuration 17
2.1 Hardware Configuration 17
2.1.1 Drive System and Electrical System 19
2.1.2 Autonomous Navigation System 20
2.1.3 Hydraulic System 26
2.2 Leg Control 29
2.2.1 Kinematics 29
2.2.2 Controller 31
CHAPTER 3 32
3. Common methodologies 32
3.1 Introduction 32
3.2 General process flow of the proposed system 32
3.3 Terrain data based on 3-D Point Clouds Data 33
3.3.1 3-D vs 2-D 33
3.4 Environment Mapping 36
3.4.1 3D Mapping 36
3.4.2 2D Mapping 38
3.5 Path planning 39
3.6 Trajectory planning 40

3.7 Omni-directional walking gait 41
CHAPTER 4 49
4. Autonomous Walking over Obstacles 49
4.1 Introduction 49
4.2 Walking patterns of COMET-IV 49
4.3 Environment mapping 49
4.4 Growing Obstacle method (GOB) application 51
4.5 Grid-based Path Planning for Legged Robot 52
  4.5.1 Traversable Cell Determination 53
  4.5.2 Walking path Determination 54
4.6 Experimental Setup 55
4.7 Results and discussion 58
4.8 Conclusion 64
CHAPTER 5 66
5. Autonomous Walking over Obstacles Laying Diagonally 66
5.1 Introduction 66
5.2 Edge detection-based path planning 66
  5.2.1 Edge detection 67
  5.2.2 Body turning angle and coordinate 68
5.3 Simulation results and discussion 70
5.4 Conclusion 72
CHAPTER 6 73
6. LRF Assisted Force-based Walking for Hexapod Robot COMET-IV 73
6.1 Introduction 73
6.2 Walking Trajectory and Pattern 73
6.3 Force threshold-based Trajectory 73
6.4 Obstacle's Geometric Requirement 75
6.5 Dynamic Stable Range for Force Control 77
6.6 Experimental setup 79
6.7 Results and discussion 79
6.8 Conclusion 84
CHAPTER 7 85
7. LRF Assisted Autonomous Walking in Rough Terrain for Hexapod Robot COMET-IV 85
7.1 Introduction 85
7.2 Environment mapping 86

7.3 Path planning 87
7.4 Locomotion strategies in stochastic environment 87
7.4.1 Cross-over an obstacle and ascending a cliff 88
7.5 Experiments and results 92
7.5.1 Crossing-over an obstacle 92
7.5.2 Crossing over an obstacle longer than 0.6 m 96
7.5.3 Ascending and descending a cliff 99
7.6 Conclusion 106
CHAPTER 8 .... **107**
8. Conclusion and future works 107
8.1 Conclusion 107
8.2 Future work 108
References **109**
Research Contributions **112**

LIST OF FIGURES

Figure 1.1: Surgical robot..2
Figure 1.2: Walking Forest Machine..2
Figure 1.3: iRobot..2
Figure 1.4: PatrolBot..3
Figure 1.5: Type of ground mobile robot..4
Figure 1.6: Legged robots (i) One leg hopper (ii) Biped robot – Asimo (iii) Hexapod robot – Genghis..5
Figure 1.7: Obstacles avoidance tasks (i) Walking over obstacle (ii) Crossing over obstacle..6
Figure 1.8: Performing specific tasks (i) The TITAN-XI (ii) The ROBOCLIMBER..7
Figure 1.9: Types of environment maps (i) Topological map (ii) Feature map with planned path..9
Figure 1.10: An environment map generated using odometry data..10
Figure 1.11: Odometry data tessellated into occupancy grid map..10
Figure 1.12: Travelling cost in A algorithm..14
Figure 1.13: Growing Obstacle method (i) Not applied (ii) Applied..15
Figure 2.1: Hardware configuration of COMET IV..18
Figure 2.2: Normally used dimension..18
Figure 2.3: Leg dimension..19
Figure 2.4: Overview of a leg control system..20
Figure 2.5: Overview of an overall control system..21
Figure 2.6: LRF (LMS200)..22
Figure 2.7: rotary stage..23
Figure 2.8: A LRF mounted on the rotary stage..23
Figure 2.9: Panasonic TOUGHBOOK CF 19FW1AXS..25
Figure 2.10: XSens digital attitude sensor..26
Figure 2.11: Hydraulic pressure circuit for COMET IV..28
Figure 2.12: Shoulder coordinate system of a leg of COMET IV..30
Figure 2.13: PID controller..31
Figure 3.1: General system configuration of obstacle avoidance system..32
Figure 3.2: Incrementally building 3-D map using 2-D LRF..33
Figure 3.3: Mapping with 2-D LRF..33
Figure 3.4: Building 3-D map by swinging 2-D LRF vertically..34
Figure 3.5: 2-D LRF is rotated around vertical axis..35

Figure 3.6: 2-D LRF is rotated around vertical axis (details) 35
Figure 3.7: Measurement parameters of LRF unit 37
Figure 3.8: Coordinate system for global coordinate mapping 37
Figure 3.9: An example of occupancy grid map 39
Figure 3.10: Generated way-point 40
Figure 3.11: Coordinate system of the omni-directional gait 42
Figure 3.12: Tripod legs movement 42
Figure 3.13: COB based omni-directional gait parameters 45
Figure 3.14: leg movement phases 46
Figure 3.15: Five directions movement with fixed yaw angle 48
Figure 4.1: A path is blocked by occupied cells 50
Figure 4.2: Condition of the path before and after shift 51
Figure 4.3: GOB area on a grid map 52
Figure 4.4: Dimensions of obstacles needed to be considered 52
Figure 4.5: The grid cell model of COMET IV 53
Figure 4.6: Cost of distance travelled 54
Figure 4.7: Shortest and safe walking path 55
Figure 4.8: Actual environment of Experiment 1 56
Figure 4.9: LRF output of Experiment 1 56
Figure 4.10: Actual environment of Experiment 2 57
Figure 4.11: LRF output of Experiment 2 57
Figure 4.12: Simulation for Experiment 1(1) 58
Figure 4.13: Simulation for Experiment 1(2) 59
Figure 4.14: Actual generated walking trajectory for Experiment 1 59
Figure 4.15: COB data of Experiment1 60
Figure 4.16: Snapshots of the Experiment 1 61
Figure 4.17: Possible trajectories of Experiment 2 62
Figure 4.18: Actual generated walking trajectory for Experiment 2 62
Figure 4.19: COB data of Experiment 2 63
Figure 4.20: Snapshots of the Experiment 2 64
Figure 5.1: Edges of obstacles 67
Figure 5.2: Turning angle 69
Figure 5.3: Turning coordinate 69
Figure 5.4: Experimental setup 70
Figure 5.5: Walking path in the free space 71
Figure 5.6: Walking path in the occupied space 72

Figure 6.1: General flow of ETT module for tripod walking pattern [35] 74
Figure 6.2: Scanning and measurement range 75
Figure 6.3: Required obstacles geometric 76
Figure 6.4: Edges of the obstacles 76
Figure 6.5: The different between generation (a) fixed generation[14] (b) dynamic generation 77
Figure 6.6: The Le(f) signal generation 78
Figure 6.7: Proposed vision assisted PPf control using GCED method 78
Figure 6.8: Actual experiment environment 79
Figure 6.9: Grid cell map of the environment (focus area only) 80
Figure 6.10: Snapshots of experiment running in 81
Figure 6.11: Sample of Le(f) signal generated during experiment(Leg1,2,3) 82
Figure 6.12: Z axis position of Leg 1 82
Figure 6.13: Roll angle of the robot`s body 83
Figure 6.14: Pitch angle of the robot`s body 83
Figure 6.15: Robot`s body height 84
Figure 7.1: Flow chart for crossing-over, ascending and descending tasks 86
Figure 7.2: Fisrt cell hits an obstacle 87
Figure 7.3: Leg swing path and related parameters 88
Figure 7.4: Parameters and conditions for crossing-over an obstacle 89
Figure 7.5: Descending a cliff with outer legs 90
Figure 7.6: Descending a cliff with inner leg 90
Figure 7.7: Leg movement during descending a cliff and required parameters 91
Figure 7.8: Experiment setup for crossing over task 93
Figure 7.9: Occupancy grid map of crossing-over task 94
Figure 7.10: Occupancy grid map with enlarged view 94
Figure 7.11: Changes is walking conditions for crossing-over task 95
Figure 7.12: Snapshots of crossing over experiment 95
Figure 7.13: Experiment setup for climbing and cross over obstacle 96
Figure 7.14: Occupancy grid map of crossing over an obstacle experiment 97
Figure 7.15: Leg swing height, body height and stride length of crossing over an obstacle experiment 97
Figure 7.16: Snapshots of crossing over an obstacle experiment 98
Figure 7.17: Snapshots of descending task (stride length = 1.2 m) 100
Figure 7.18: Snapshots of descending task (stride length = 0.9 m) 101
Figure 7.19: Occupancy grid map of descending task (stride length = 1.2 m) 102

Figure 7.20: Occupancy grid map of descending task (stride length = 0.9 m) 102
Figure 7.21: Changes in walking condition for descending task (stride length = 1.2 m) . 103
Figure 7.22: Changes in walking condition for descending task (stride length = 0.9 m) . 103
Figure 7.23: Snapshots of ascending task 104
Figure 7.24: Occupancy grid map of ascending task 105
Figure 7.25: Changes in walking condition for ascending task 105

LIST OF TABLES

Table 2.1: Specifications of LRF (LMS200) 22
Table 2.2: Rotary stage specifications 24
Table 2.3: Specifications of XSens sensor 25
Table 2.4: Link parameters of a leg of the COMET-IV 29
Table 3.1: Full platform specification 41
Table 3.2: Parameters of some gait patterns 44
Table 3.3: SCS control input for side/zig-zag movement 45

LIST OF ABREVIATIONS

COMET-IV	: Chiba University Operating Mine Detection Electronics Tools - Fourth version
LRF	: Laser Range Finder
GWTLR	: Grid-based Walking Trajectory for Legged Robot
GWALR	: Grid-based Walking Assistant for Legged Robot
DOF	: Degree of Freedom
GPS	: Global Positioning System
GCED	: Grid-cell Edge Detection
COB	: Center of Body

CHAPTER 1

1. Introduction

1.1 Background

For more than a century researches in robotics has been carried out in order to provide efficient assistance for the convenience of human beings, including replaces human for dangerous works, by made it partially or fully autonomous. The surgical robot, DaVinci, shown in **Figure 1.1**, is an example of the partially autonomous robot arm, and it was developed to overcome the limitations of minimally invasive surgery. The surgery job can be performed through a telemanipulator or by computer control. On the other hand, an example of the partially autonomous mobile robot is the Walking Forest Machine, shown in **Figure 1.2**, is developed by Plustech Company. The leg coordination of the robot is automated, but navigation is done by a human operator on the robot.

There are many examples of the fully autonomous robots that are developed, but majorities of them are daily life assistant, such as the home cleaning robot, iRoomba as shown in **Figure 1.3**, and a service robot, PatrolBot as shown in **Figure 1.4**. However, to perform some works in hazardous environment, human-robot cooperation is needed, but to gain higher quality achievement of work and reduced operations effort envisioned for complicated tasks, enhance robotic capabilities and increased robot autonomy are required.

In general, the robots are categorized into three main types depends on its moving mechanism: wheeled, legged and tracked types. **Figure 1.5** shows the examples of the mechanisms. In addition, each of a mechanism provides some advantageous and disadvantageous to each other.

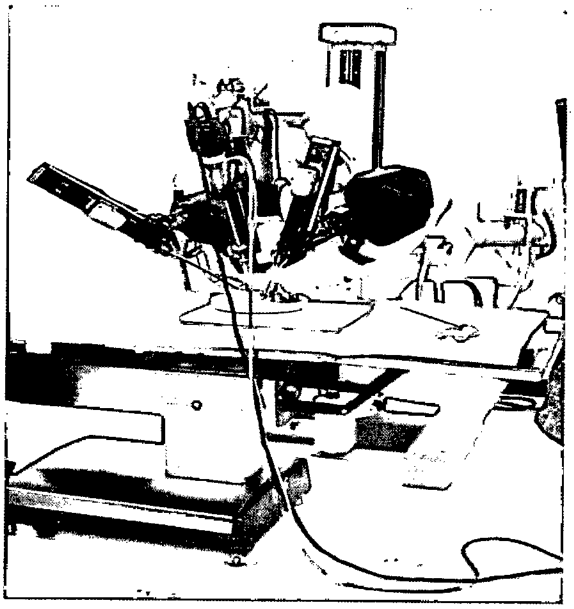


Figure 1.1: Surgical robot

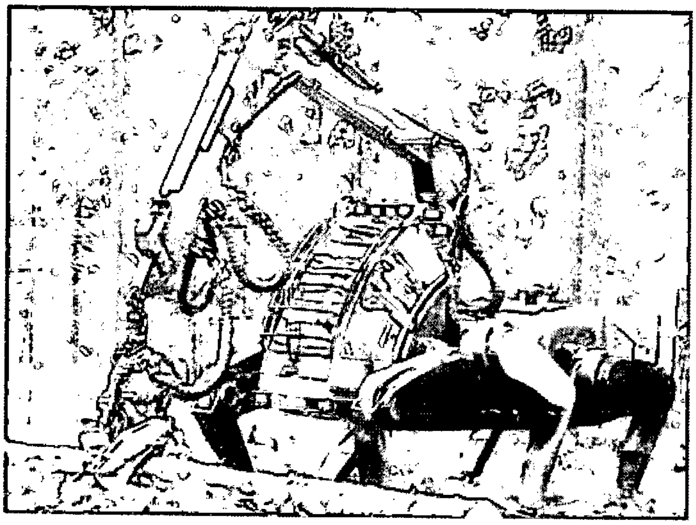


Figure 1.2: Walking Forest Machine

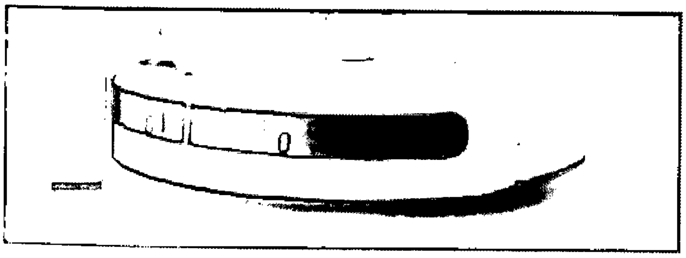


Figure 1.3: iRobot

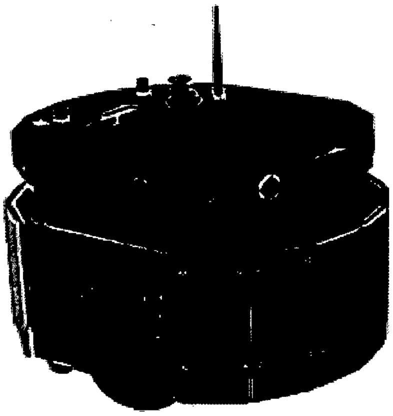
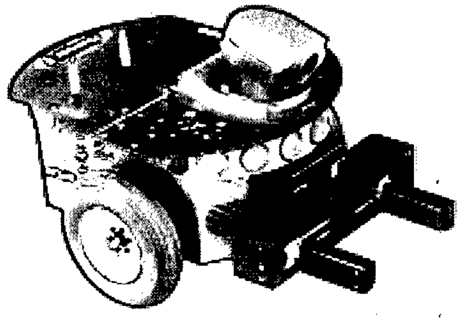
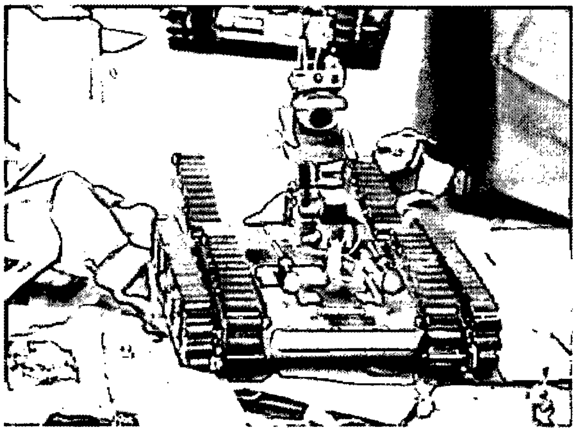


Figure 1.4: PatrolBot

Although wheeled mechanism is used for almost all of transports that we can see in our daily life, it is just for hard and even terrains, and not proper for rough and unknown terrain. On the other hand, the tracked mechanism has no problem for the mentioned terrains, since the palettes has a wide surface contact with ground. But, the mechanism can damage the terrains, especially during turning because of the palettes has to be with slip friction. In such conditions, the legged mechanism can be an alternative to them. Because of the legged locomotion mechanism is inspired by the legged animals in nature, it is suitable for the loose, rough and uneven terrains. The ground condition between a set of point contacts between the robot and the ground is not affected the movement of the robot so long as the robot can maintain adequate ground clearance [1]. Furthermore, each of its legs can be controlled individually. Besides that, the mechanism can act as an active suspension, and the body height can be adjusted, and can be kept in horizontal condition in almost of all time. However, the wheeled mechanism performs better compared to other two types in terms of energy consumption, simple technology to be manufactured and to be used, easy to change in direction, and able to move with high speed. The legged mechanism is disadvantageous in terms of energy consumption, having complicated kinematics and dynamics, and needs a lot of actuators for movement control purpose.



(i) Wheeled type robot



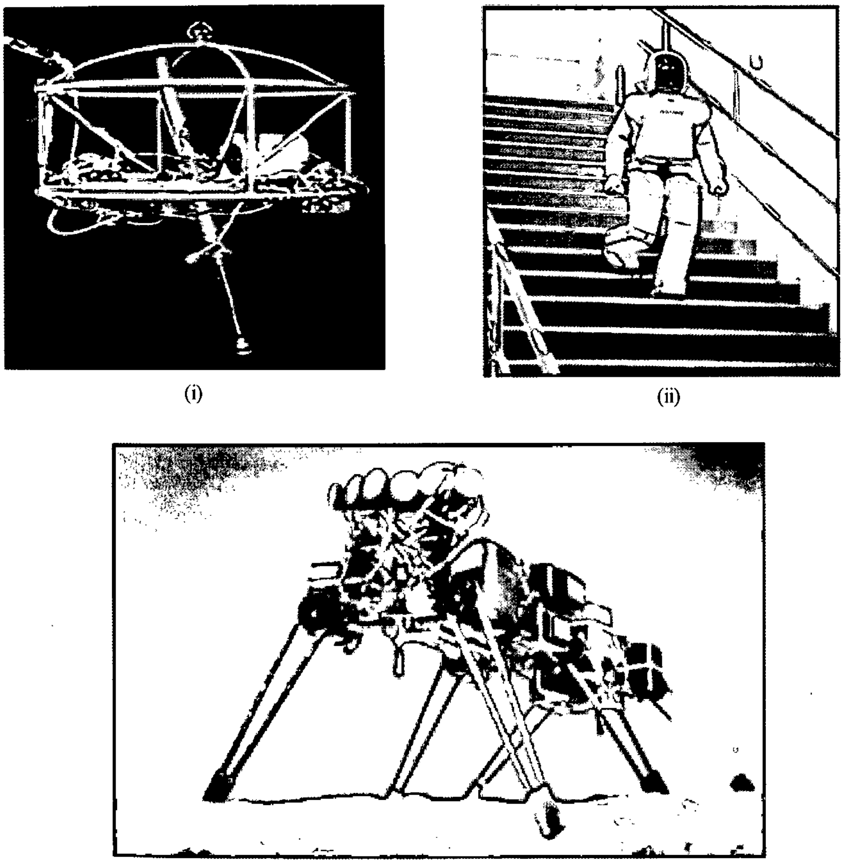
(ii) Crawler type robot

(iii) Legged type robot

Figure 1.5: Type of ground mobile robot

1.2 Legged robot

Many of the mechanisms for legged robots are configured with one, four and six legs, as shown by **Figure 1.6**. Among the legged robots, the six legged robot is the most favorable choice because static stability during walking, in which it's maintain three legs on the ground while walking, and due to that the control complexity is reduced [1]. In statically stable gaits, the ground projection of the center of gravity of the system always lies in the polygon determined by the supporting legs. In every step, a new polygon is formed and the center of gravity always stays inside these polygons. The disadvantage of such gaits is that the locomotion is considerably slow in order to sustain the static stability. The walking speed is not the priority for large scale hydraulically-driven robots, such as the COMET-IV [2], TITAN-XI [3], MECANT [4] and ROBOCLIMBER [5].



(i) (ii)

(iii)

Figure 1.6: Legged robots (i) One leg hopper (ii) Biped robot – Asimo (iii) Hexapod robot – Genghis

In general, the legged robot has some advantages over a wheeled or tracked robot for operating in stochastic terrain because it can perform many types of obstacle avoidance tasks, such as walks-over, crosses-over, climbing and descending a step. **Figure 1.7** illustrates some of the different obstacle avoidance tasks performed by the legged robot. These operations would not be feasible for wheeled or tracked robot. The capabilities of the legged robot to perform the tasks are depending on the leg's joint design structures, materials and the number of degrees of freedom (DOF) of the robot. Higher DOF gives more capabilities to the robot. In contrast, the control algorithm used becomes more complex.

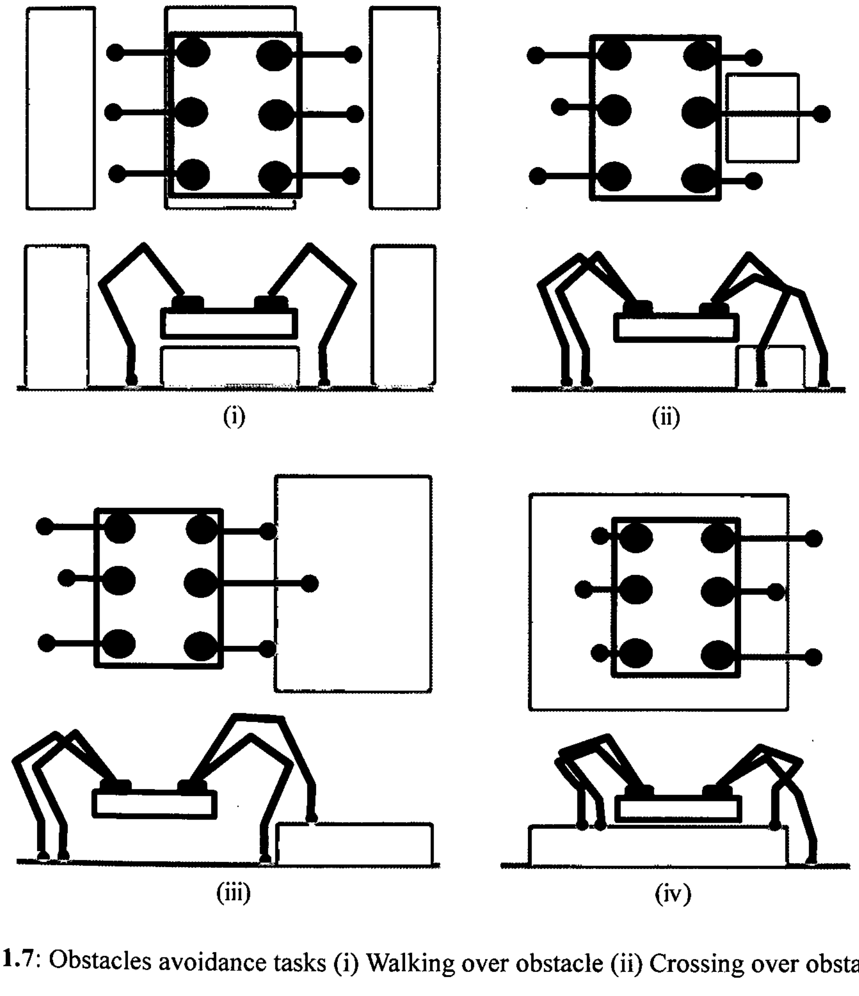
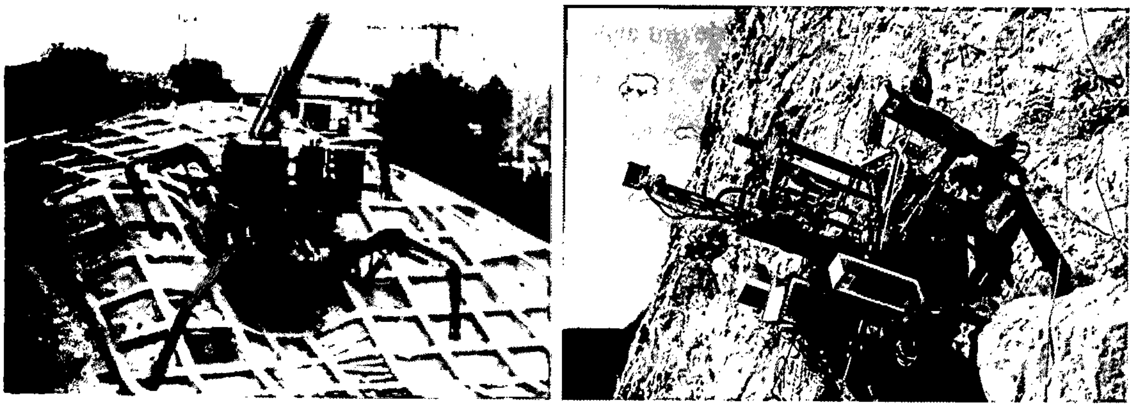


Figure 1.7: Obstacles avoidance tasks (i) Walking over obstacle (ii) Crossing over obstacle (iii) Ascending a cliff (iv) Descending a cliff

Some examples of legged robots that are developed to replace humans in doing works in hazardous environment are the TITAN-IV and ROBOCLIMBER, as shown in **Figure 1.8.** TITAN-XI is a quadruped robot which is developed to carry out autonomous drilling holes of rock-bolts or anchor-bolts on mountainous area to prevent steep slope failure or landslides. Similarly, the ROBOCLIMBER is also a quadruped robot which is developed for slope consolidation and monitoring, to remotely and automatically perform deep drilling on a mountain. However, both of them are transported to the vicinity of the site and the climbing operation is supported by an operator using tele-operation system.

The problem is when the robots are can't be transported to the vicinity of the site due to hazardous area, such as the vicinity of Fukushima Daiichi nuclear plant, where humans are prohibited to enter the area within a 30 km radius of the crippled Fukushima No. 1 nuclear power plant.

One of the possible solutions of the problems may be let the robot entering the hazardous area by autonomously, or navigate the robot from a remote place. However, both of the operating methods need an environment map of the area, a navigation system and a control system.



(i) (ii)

Figure 1.8: Performing specific tasks (i) The TITAN-XI (ii) The ROBOCLIMBER

1.3 Robot Navigation

Navigation is the process of monitoring and controlling the movement of a craft or vehicle from one place to another. For instance, a car is driven by a driver from a start point to a destination following direction and a map given by a navigation system. A car navigation system may suggest some routes for the driver to choose, whether to select a shortest distance but has to pay more for tolls, or a long distance using toll-free road. Normally, the location of the car at any time in this period is determined based on Global Positioning System (GPS) data of the system. Similarly, in the case of a robot navigation system, a map either is given or is developed by the robot itself. Based on the map, the navigation system of the robot selects a shortest path and relays the information to a controller that drives the robot from a start point to a targeted destination, while avoiding obstacles. The location of the robot may also be traced using odometer or other sensors such as GPS. The robot may be controlled by an operator for a manual navigation, and by a set of algorithm that is already uploaded into the robot computer for an autonomous navigation. However, as we can find in many articles [6-9], the mapping task is done by a wheeled robot, which has capabilities to move fast, immediate turning and immediate stops. The wheeled robot can explore a big area and constructs a map of unknown environment by itself, then plan a walking path after that. Furthermore, the information associated in a map developed by a wheeled robot, not included with obstacles height since the purpose of the map built is to navigate a robot from a start point to a destination point while avoiding those obstacles.

In this thesis, a map that is going to be used by the robot will be developed in real time by the robot itself, which includes the height of obstacles exist in the environment, to enable the legged robot to choose whether to avoid, walking-over, cross-over or step-on the obstacles. However, the map is not an entire map for a complete travel course from a start point to a final targeted goal, but a local map that will be updated for every 8 meters. A newly developed algorithm will be used to determine an appropriate travel path and to plan the motion of the robot to achieve an optimal travelling cost. The robot wills not only avoiding obstacles but rather will cross over or step on the obstacles if necessary as illustrated by **Figure 1.7**.

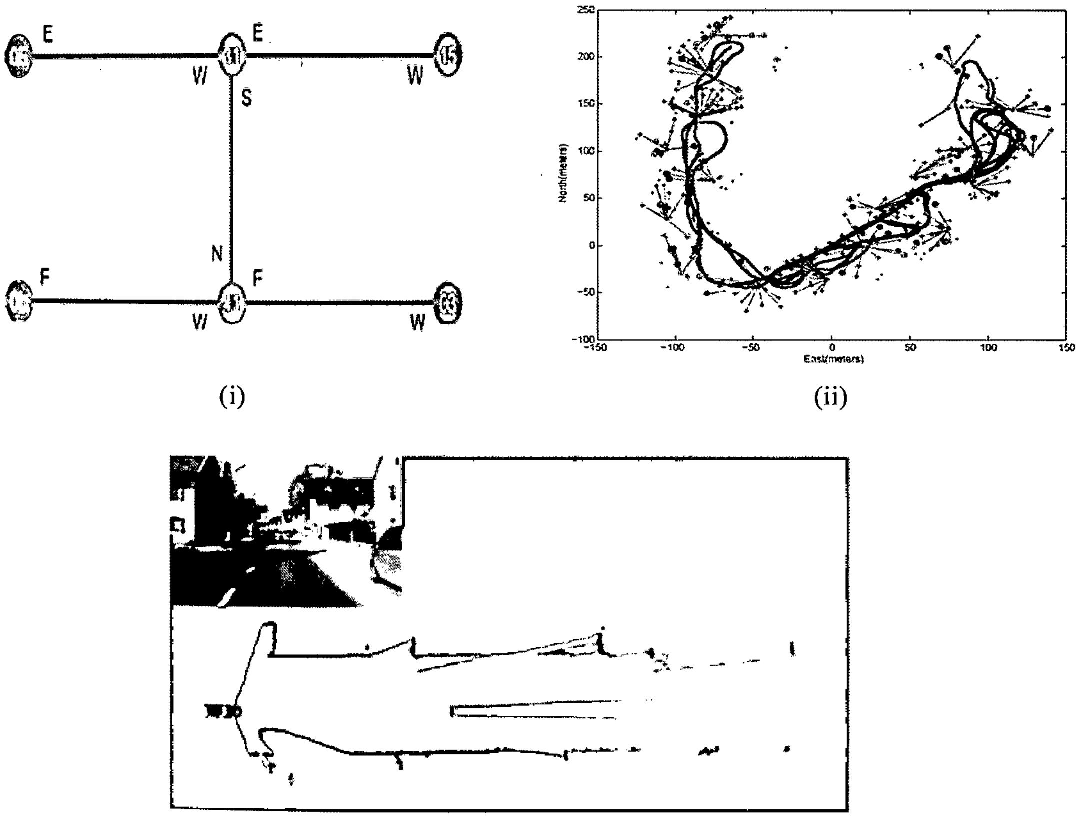
1.3.1 Environment Mapping

Environment mapping can be classified into three types: feature maps, topological maps and occupancy grid maps [10] as shown in **Figure 1.9**. They are generally represented a list of objects in the environment and their locations. Equation (1.1) depicts a list of objects exist in an environment [11]:

$$m = \{m_1, m_2, \ldots, m_i\} \tag{1.1}$$

Where m represents the map, i is the number of objects exist in the environment and m_i represent each of objects' properties. The occupancy grid maps use evenly-spaced grids, and each grid is considered as occupied or empty. The occupancy grid maps are also known as location based maps. A planar map is normally used for representing the locations of objects in an environment, and each of objects is represented as $m_{x,y}$ related to its location in global world coordinate system. The examples of works that employ this mapping method are [12],

[13], [14], [15]. On the other hand, in feature maps the environment is represented by parametric features such as points, lines and arcs as employed by [16], [17]. The value m_i represents the features and Cartesian distance of the object. Then, the topological maps use a graph in which the nodes in the graph may represent landmarks or places, without scale, but the relationship between points is maintained, as shown by [18]. Among the three environment mapping methods, the occupancy grid maps are favored because it is easy to maintain and construct [10]. On the other hand, environment mapping is also can be done by directly generate the map using vehicle's raw odometry information [11], as shown in **Figure 1.10.** Comparing with the figure generated using Occupancy grid map as shown in **Figure 1.11** gives understanding the difficulty of the mapping problem, and how the occupancy grid map method capable to produce a better map. However, the size of cell plays very important roles in accuracy level of the data stored in each cells. Smaller size gives higher accuracy but data processing cost also increases and vice-versa.



(i)

(ii)

(iii)

Figure 1.9: Types of environment maps (i) Topological map (ii) Feature map with planned path (iii) Occupancy grid map

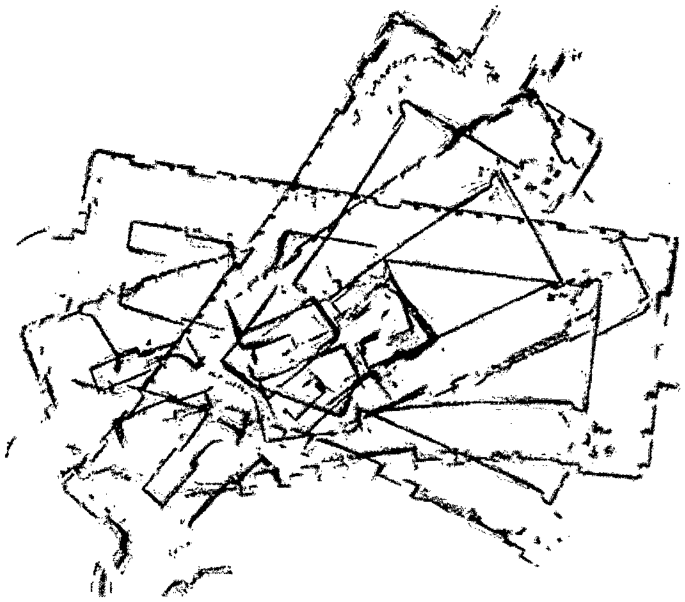


Figure 1.10: An environment map generated using odometry data

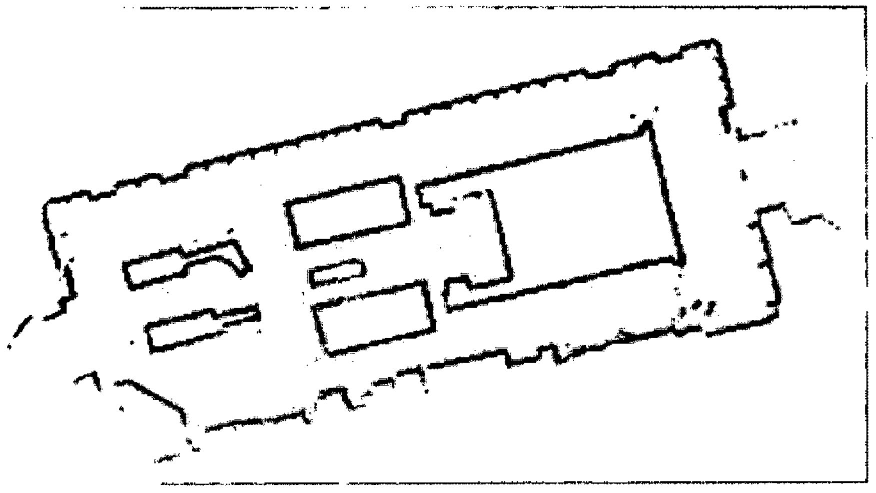


Figure 1.11: Odometry data tessellated into occupancy grid map

The study conducted by M. Perrolaz et al. [13] employed occupancy grids that are tessellated from stereo-vision data for geometry representation of the environment. Based on the information, a formal probabilistic model is used to calculate whether the cell is occupied or empty but did not consider the height of the associated objects. To avoid large processing of the cloud data, they also applied u-disparity approach and hence ensuring efficient processing of the data.

In the work carried out by Cang Ye [12], the occupancy grid map is used to represent the environment by tessellating the data from a 2-dimensional (2D) LRF. Each of grid cells holds an index, namely PTI (polar Traversability Index) that shows traversability through it, based on the height of obstacle in it and the PTI histogram is used to navigate the robot. The measurement is updated for every 13.3 ms. However, since the robot used is a wheeled robot, it can traverse only on low-profile ramps. Therefore, the height value associated in each cell is limited to a certain value. Furthermore, since the measurement did using 2-D LRF, obstacles height that can be measured is also limited.

1.3.2 Sensors for mapping

An environment map is generated based on the information acquired using sensory modalities such as LRF, camera, sonar sensors etc. A variety of techniques have been applied in developing the map, some researchers have used a single sensor in their studies [12], [19], [20], while others have employed a combination of two types or more of sensors [21], [22]. However, each sensor has its advantages and disadvantages. Many literatures have been found describing the advantages and disadvantages of some sensors that are favorable in developing a map. For instance, the camera is sensitive to illumination and shadow and has limited field of view compared to LRF, but it provides a lot of information on elements such as texture and color. On the other hand, LRF only provides information regarding objects in the plane of the scanning laser beam, and has some difficulties in identifying some objects, such as transparent objects, and the objects that can absorb the laser beam used for scanning the objects [23], but the information provided by LRF is generally accurate. Furthermore, depending how it is operated, the LRF can provide 3-D information on the scanned object. Therefore, sensor integration is a must to ensure the system functions accordingly when they are used real environment.

1.3.3 Path planning and Trajectory Planning

Path planning is the subset of the motion planning problem. Path planning describes the design of only geometric (kinematic) specification of the positions and orientations of robots. Another subset of the motion planning problem is the trajectory planning that describes the design of linear and angular velocities. The premier objective of the motion planning problem is to produce a continuous movement plan from a start configuration to a goal configuration while ensuring no collision with obstacles in configuration space.

Traditional approaches to path planning can be classified into three basic categories: the Roadmap method [24], the Cell Decomposition method [25] and the Potential Field method [26]. The Roadmap approach is where the obstacle free area is modeled as a network of lines. This network is then searched for a path that connects the start and goal points. Cell Decomposition is where the obstacle free area is subdivided into cells that are interconnected to each other. These cells are then searched to find a path that connects the start and goal points. Potential field methods use imaginary forces acting on a robot. The goal position attracts the robot by pulling it towards the goal, whereas the obstacles repulse the robot by pushing it away. Recently, many new approaches have been proposed that include optimization of the path planning problems such as [12] [21][27] and [28]. Two examples are described below.

K. Joo et al. [27] employed a method for a cleaning robot to generate a topological map from an occupancy grid-map. After the topological map is generated and optimized using genetic algorithm, it is used to plan more efficient motion including room-to-room path planning and covering each room. However, this method is not proven to be used in other situation of environments, other than in the space which has some divided spaces, such in a house that has several rooms.

A. R. Forseca et al. [28] proposed a method using an efficient discretization method based on Constrained Delaunay Triangulation (CDT) and classical graph searching algorithms, A* algorithm and D* algorithm. In this work, thematic maps of different information are combined and the travel cost to each vertices are assigned and using one of the searching algorithms to determine the shortest path. Nevertheless, this method is not suitable for walk-over, cross over and step-on obstacles tasks to be performed.

1.3.4 A* Algorithm

A* algorithm is one of the best graph based path finding algorithm which automatically search the shortest, lowest cost and collision free path. Nevertheless, the algorithm works well if a robot has priori information of its environment [29]. Another examples of the searching algorithm are Dijktra's [30] and D* [31] algorithms. An environment that is usually applied this search algorithm is divided into a square grid, and it is reduced to two dimensional array. The A* algorithm is used to determine the shortest path from one node to another adjacent nodes based on heuristic information regarding the travelling cost of the unknown part of the traversable portion. Refer to **Figure 1.12**, lets *n* is the node of a graph, ***S*** is the start point, ***G*** is targeted destination, $g(n)$ is the minimum traversal cost from ***S*** to node *n*, and $h(n)$ is the minimum cost from the node n to the ***G***, then the total minimum cost from ***S*** to ***G***, $f(n)$ is defined by the following equation.

$$f(n) = g(n) + h(n) \qquad (1.1)$$

However, these functions are actually are known during the search. Thus, the algorithm is executed based on their estimates functions, which is shown in Equation 1.2.

$$\hat{f}(n) = \hat{g}(n) + \hat{h}(n) \qquad (1.2)$$

Where, $\hat{g}(n)$ is the observed minimum cost from S to node n, $\hat{h}(n)$ is an estimate of and $\hat{f}(n)$ is the estimated total minimum cost from a start configuration ***S*** to a goal configuration ***G***. A* search the minimum cost among the paths connected from a node to its adjacent nodes, and expands it to the rest of the nodes in the graph. **Figure 1.12** illustrates the searching process. The search process start at node ***S***, and the cells labeled with 1, 2, 3, 4 and 5 are its adjacent cells. The black colored cells are the obstacles exist in the environment. A* determines the minimum cost from node ***S*** to each adjacent node and subsequently added the estimated cost from the adjacent cells to the node ***G***, by ignoring whether the cells are occupied or not. As can be seen, the minimum cost path is when the robot travel from node ***S*** to node 2 and to node ***G***. Then, node 2 is considered as start point, and the same step is repeated, until the selected node is the node ***G***.

Any nodes that are already searched are called *CLOSED*, and the adjacent nodes to the selected node are included in a list called *OPEN*. The selected path from the ***S*** to the ***G*** is considered has minimum cost when it fulfills the following condition.