Design of an Image-based Fuzzy Controller for Parking Problems of a Car-like Mobile Robot

YIN YIN AYE

Department of Intelligent Mechanical Systems
Okayama University

This dissertation is submitted for the degree of
Doctor of Philosophy in Engineering

March 2017

Acknowledgements

The success of research work in this thesis became possible due to a lot of help from many people during my Ph.D. study at Okayama University.

Firstly, I would like to express my deepest gratitude to my advisor, Professor Keigo Watanabe, for his valuable guidance, encouragement, patience, motivation and support in completing the research work in this thesis. His teaching and guidelines have served as a role model for me not only in academic but also in personal life.

I wish to express my thanks to Associate Professor Shoichi Maeyama and Assistant Professor Isaku Nagai for their suggestions and comments throughout my Ph.D. reserach. Their kind guidance, encouragement, and patience have been great support throughout my research.

I would like to thank my thesis committee members, Professor Mamoru Minami and Professor Akio Gofuku, for their encouragement, insightful comments and reviewing this dissertation. I feel proud and honoured that you have accepted to be on my committee.

My sincere thanks also go to all my friends and colleagues at our laboratory for the pleasant atmosphere. Everyone has been open-minded to discussions which were contributed to this dissertation. This thesis could not have been done without the help and support of numerous people. Especially, I would like to thank Dr. Kimiko Motonaka and Mr. Maierdan Maimaitimin for their fruitful discussions and invaluable assistant they provided me in all phases of my work, especially in image-based fuzzy control and optimization of fuzzy controller using a genetic algorithm.

I am deeply grateful to Mr. Junya Ukida , Mr. Hikaru Fujioka, and Mr. Toshiyuki Kageyu for their providing deep insights and cooperation on the development of image-based fuzzy control and for their technical support in the experiments.

Due to the support from people around me, I survived in Japan, in both academic and personal life. I also would like to thank all previous and current student members of Mechatronic Systems Laboratory, who have been very helpful not only in laboratory but also in personal matters. I also would like to thanks the staff from Department of Intelligent Mechanical Systems and Centre of Global Partnerships and Education for their kind help during my stay in Okayama University.

The writing of this dissertation became possible due to the financial support by Japan International Cooperation Agency (JICA), Mechatronic Systems Laboratory and Okayama University.

Finally, I would like to offer my special thanks to my parents, brother and sister, who always give encouragement, support and good advices.

Abstract

With increasing number of vehicles over years, parking systems have become an important issue in commercial environments such as shopping malls and airports. For this reason, the automatic parking systems of car-like mobile robots have attracted a great deal of attention from research organization and automobile industry. In many recent works, a number of different approaches have been developed to build an automatic parking system. Most of them are still expensive for commercialization.

In the middle of 80-th, the first ideas of automatic parking systems were developed. Under this topic, a fuzzy algorithm for garage parking system, a first nonlinear algorithm for parking control, and the main statements for parallel parking problem, were proposed as early researches. Most of the researches related to a parking problem are usually to find a path that connects the initial configuration to the final one by considering the nonholonomic constraints. On the other hand, a skill-based approach that used the fuzzy logic, was developed. There is no reference path to be followed by a robot and the control command is generated by considering the position and orientation of the robot relative to the parking space. An exact vehicle pose relative to the parking space is required in this approach. The automatic parking system consists of a parking lot detection module and an automatic parking algorithm. In the recent works on parking control systems, a method for detecting camera vision-based parking lots and a method which used the ultrasonic sensors to detect the parking lots, were discussed. For the experimental study of fuzzy garage parking control, the CCD camera was utilized to detect the overall vision of the parking lot, the six infrared sensors were adopted to measure the distances between the robot and the surroundings, and the sensor fusion techniques that combine the ultrasonic sensors, encoders, and gyroscopes with a differential GPS system were used to detect and estimate the dimensions of the parking lot in recent works. However, all are the expensive methods.

This thesis aims to build an automatic parking system of a car-like mobile robot. The automatic parking system that corresponds to the posture stabilization problem of car-like mobile robot, is presented in the first part of this dissertation. The goal is to stabilize the car-like mobile robot to a desired final posture starting from any initial posture (posture means the combination of the position and orientation of the robot). The switching and

non-switching controllers have been developed for stabilizing the car-like mobile robot that was a nonholonomic underactuated system with two inputs and four outputs. Since this system cannot be stabilized by a static continuous feedback with constant gains, there are several control methods by using a canonical form up to now such as chained form, a power form, a Goursat normal form and a double integrator model. In the first part of this thesis, after obtaining a chained form, the invariant manifold technique was applied for stabilizing the car-like mobile robot in the desired posture. The performances of the proposed controllers for parking problem were verified through some computer simulations.

An image-based fuzzy controller for autonomous parking of a car-like mobile robot is presented in the second part of this thesis. In recent past, vision sensor and digital image processing technology become easily available with the development of computer technology. Moreover, a vision sensor can be used to measure the environment without physical contact. Visual servoing, which control the motion of the robot using one or more cameras, are conventionally classified in four groups: position-based visual servoing (PBVS), image-based visual servoing (IBVS), hybrid visual servoing (HVS), and motion-based visual servoing (MBVS). The aim of the second part of this thesis is to build a parking controller upon the concepts of IBVS. In the IBVS, both of the control objective and the control law design are directly performed in the image feature parameter space and thus the full model of the object need not be known accurately. To combine the behavior of the robot with the image features, a model-free fuzzy controller was designed without estimating the image depth in this work. The second part of thesis presents an automatic parking system of a car-like mobile robot using image-based fuzzy control. The proposed system receives the image information about the parking frame from a web camera which is equipped on the top of the robot, and then generates the desired target line to be followed by the robot using Hough transform. A fuzzy controller is designed with a reasoning mechanism composed of two inputs, which are the slope and intercept of the target line, and one output that is the steering angle of the robot. The results of both simulation and real robot experiments confirmed that our image-based fuzzy controller is effective and feasible for parking problem.

In the third part of this thesis, a genetic algorithm (GA) was used to optimize the parameters of membership functions of an image-based fuzzy parking controller. It is well known that the process of manually tuning a fuzzy logic controller is a very complex task, which consists of choosing the type of fuzzy logic controller, the number and shape of membership functions of inputs and outputs, and the rule base. The image-based fuzzy controller for parking problem of the car-like mobile robot was developed in the second part of this thesis, where the membership functions and fuzzy rules were tuned manually by trial and error. The aim of the last part of this thesis is to optimize the parameters of the

membership functions by using the GA against the complicated tuning of the controller. The fuzzy rules are constructed based on the skills of the experienced human drivers in advance and the GA was not used to optimize the rules set of the fuzzy controller. Simulation results are given to demonstrate the effectiveness of the optimized image-based fuzzy controller by GA.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Motivation

As the number of vehicles is increasing over years, the streets and the parking lots are becoming more crowded, which makes the car parking more difficult even for experienced drivers. The parking maneuver is a difficult task because much attention and driving experience are needed to control the vehicle in a special constraint environment. For this reason, the research of automatic parking control has gained much attention from academics and automobile industries in recent years. The recently developed approaches on parking control still need to improve on safety and reduce on cost for commercialization.

Motivated by the above problems, this thesis proposes new techniques for parking problem of a car-like mobile robot in order to improve the parking capability of the vehicle. In recent works, it is considered that image based localization will be a more practical choice for automatic parking. The researches used the visual robot control for parking problem, estimated the position and orientation of the robot by matching previous knowledge of 3D environmental models with a captured image and controlled it directly. It is normally called the position-based control method. On the other hand, the robot is controlled without referring to its position and it is called the image-based control method (IBCM). The IBCM does not need the knowledge of the 3D model of the environment and can operate a robot by controlling only image information which is acquired from a camera image without using the robot position. The IBCM seems to be similar to the processing that is performed by human in not referring to own position and it has advantages of calculation costs due to saving the position estimation. The research in this thesis aims to build an automatic parking system upon the concepts of image-based control system.

1.2 Aims of the Thesis

The research work carried out in this thesis intends to achieve three main aims. The first aim is to develop the switching and non-switching controllers based on an invariant manifold theory for point-stabilizing (parking problem) a car-like mobile robot. The second aim of this thesis is to build an image-based fuzzy controller for an automatic parking system of the car-like mobile robot. The third aim of this thesis is to optimize the parameters of the membership functions of our image-based fuzzy controller using a genetic algorithm.

1.3 Major Contributions

This thesis made three contributions to deal with the parking problems of a car-like mobile robot. The major achievements are as follows:

- **New invariant manifold-based stabilization controllers (Chapter 3)**

  The field of mobile robot control is a challenging subject for both its theoretical and practical value. The design of stabilizing control laws for this system can be considered a challenge due to the existence of nonholonomic constraints. Various researches under this topic have been performed so far. However, their research studies for stabilizing control of nonholonomic mobile robots are all about the stabilizing problem at the origin.

  This thesis presents some stabilizing controllers for point-to-point control (parking problem), which allow a car-like mobile robot to reach the desired pose starting from any initial pose. For stabilizing the car-like mobile robot in the desired pose, three switching and one non-switching controllers based on an invariant manifold theory were developed. Our invariant manifold approach composes an incomplete manifold, which is invariant under a linear state feedback controller, on which all the states converge to the desired position except the vehicle orientation. We found that there is a limitation on the vehicle orientation due to the incomplete manifold in the system. Therefore, we enhance the linear state feedback controller to make the constructed manifold with the method of Tayebi *et al*. [7]. The simulation results illustrate the effectiveness of the proposed method.

- **New image-based fuzzy controller (Chapter 4)**

  In recent years, there has been increasing interest in the use of vision-based robot control. It is conventionally classified in four groups and one of them is used for this research.

This thesis presents an automatic parking system of the car-like mobile robot using an image-based fuzzy controller. To the best of our knowledge, there is no IBVS controller developed for automatic parking system of car-like mobile robots. The image-based control method was able to reduce the amount of calculations because it did not need to perform the position estimation processes. The proposed system enabled the car-like mobile robot to park towards the red color parking frame drawn on the floor. The results of simulation and real robot experiments are given to prove the effectiveness of the proposed control scheme.

- **New optimized fuzzy controller by GA (Chapter 5)**

  In many recent works, the GA has been applied to optimize the fuzzy logic controller and some self-tuning methods using GA have been proposed to reduce the required tuning efforts by human operators. To obtain a better performance of our image-based fuzzy controller, it is very important to adjust the widths of membership functions of input and output variables.

  This thesis presents an automatic parking system using an optimized image-based fuzzy controller by a GA. It was proved that tuning of the shape of membership functions of input and output variables, can be automated by using the GA to optimize the proposed fuzzy controller in this thesis. The GA is not used to tune the fuzzy reasoning rules and these are constructed based on the skills of the experienced human drivers in advance. In our research, genetic algorithm has introduced to reduce the required tuning efforts of the 81 fuzzy control rules by a human operator. The simulation results illustrate the effectiveness of the developed schemes.

1.4 Organization

The chapter contents of this dissertation can be briefly summarized as follows:

Chapter 2 describes the background information that leads to conduct the research in this thesis.

Chapter 3 proposes new switching and non-switching controllers based on an invariant manifold theory for parking problems of a car-like mobile robot.

Chapter 4 devotes a novel method to design a car-like mobile robot that possesses autonomous angle parking and perpendicular parking capability by using an image-based fuzzy controller.

Chapter 5 presents GA optimized image-based fuzzy controller for an automatic parking of

a car-like mobile robot.

Chapter 6 concludes this thesis with a summary of the work carried out.

1.5 Publications

The research work presented in this thesis has resulted in the following publications.

Journals

1. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Invariant manifold-based stabilizing controllers for nonholonomic mobile robots," *Artificial Life and Robotics*, vol. 20, no. 3, pp. 276–284, 2015.

2. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Image-based fuzzy parking control of a car-like mobile robot," *International Journal on Smart Material and Mechatronics*, vol. 3, no. 1, pp.160–164, 2016.

3. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Design of an image-based fuzzy controller for autonomous parking of four-wheeled mobile robots," *International Journal of Applied Electromagnetics and Mechanics*, vol. 52, no. 3-4, pp. 859–865, 2016.

4. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "An intelligent parking system for vehicles using an image-based fuzzy controller," *International Journal on Smart Material and Mechatronics*, (Accepted: 30, Dec. 2016).

5. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "An automatic parking system using an optimized image-based fuzzy controller by genetic algorithms," *Artificial Life and Robotics*, vol. 22, no. 1, pp. 139–144, 2017.

International Conferences

1. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Controllers based on an invariant manifold approach for stabilizing a nonholonomic mobile robot," in *Proceedings of the 7th International Conference on Soft Computing and Intelligent Systems and the 15th International Symposium on Advanced Intellligent Systems (SCIS & ISIS)*, pp. 134–139, Fukuoka, Japan, Dec. 2014.

2. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Invariant manifold-based stabilizing controllers for nonholonomic mobile robot ," in *Proceedings of the*

5th International Conference on Science and Engineering (ICSE), Yangon, Myanmar, Dec. 2014.

3. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, " Stabilization of nonholonomic mobile robot using controllers based on an invariant manifold theory," in *Proceedings of the 20th International Symposium on Artificial Life and Robotics (AROB)*, pp. 537–542, Beppu, Japan, Jan. 2015.

4. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Design of an image-based fuzzy controller for autonomous parking of four-wheeled mobile robots," in *Proceedings of the 17th International Symposium on Applied Electromagnetics and Mechanics (ISEM)*, 2P1-D-1 isem2015-076.pdf, Kobe, Japan, Sept. 2015.

5. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Automatic parking of a car-like mobile robot using an image-based fuzzy controller," in *Proceedings of the 2nd International Conference on Smart Material and Mechatronics (ISSMM)*, pp. 62–66, Makassar, Indonesia, Oct. 2015.

6. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Generation of time-varying target lines for an automatic parking system using image-based processing," in *Proceedings of International Conference on Robotics and Biomimetics (IEEE ROBIO)*, pp. 423–427, Zhuhai, China, Dec. 2015.

7. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Optimization of an image-based fuzzy controller for an automatic parking system using a genetic algorithm," in *Proceedings of the 21th International Symposium on Artificial Life and Robotics (AROB)*, pp. 354–357, Beppu, Japan, Jan. 2016.

8. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Image-based fuzzy control of a car-like mobile robot for parking problems," in *Proceedings of International Conference on Mechatronics and Automation (IEEE ICMA)*, pp. 502–507, Harbin, China, Aug. 2016.

9. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "An intelligent parking system for vehicles using an image-based fuzzy controller," in *Proceedings of the 3rd International Conference on Smart Material and Mechatronics (ISSMM)*, pp. 66–70, Makassar, Indonesia, Nov. 2016.

10. Kyaw Thiha, Yin Yin Aye, Keigo Watanabe, and Isaku Nagai, "Autonomous parking system of a car-like mobile robot using an image-based fuzzy controller," in *Proceed-*

ings of the 7th International Conference on Science and Engineering (ICSE), Yangon, Myanmar, Dec. 2016.

National Conferences

1. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, " Image-based fuzzy parking control of nonholonomic vehicles," in *Proceedings of Symposium on Fuzzy, Artificial Intelligence, Neural Networks and Computational Intelligence*, Osaka, Japan, Oct. 2016.

2. Yin Yin Aye, Keigo Watanabe, Shoichi Maeyama, and Isaku Nagai, "Image-based fuzzy garage parking control of a car-like mobile robot," in *Proceedings of the 17th SICE System Integration Division Annual Conference*, Sapporo, Japan, Dec. 2016.

Chapter 2

Background Research

The objective of this chapter is to conduct a thorough background research on the areas of interest of this thesis. Since the thesis focuses on an image-based fuzzy parking control of a car-like mobile robot, the contents of this chapter can be divided into the following four main sections.

- **Stabilizing Control Using Invariant Manifolds:** This section introduces about the mobile robots and discusses about the point-stabilization problem of nonholonomic mobile robots.
- **Visual Servoing:** This section discusses the importance of the visual servoing in the current research on mobile robotics. Four types of visual servoing, position-based, image-based, hybrid, and motion-based visual servoing problems are also given.
- **Mobile Robot Control Using Fuzzy Logic:** The recent developments of fuzzy systems used in the mobile robot control problems are discussed. Fuzzy logic controllers for parking problem of car-like mobile robots are given the major focus.
- **Optimized Fuzzy Controller by Genetic Algorithms:** This section presents the fundamental concept of genetic algorithms and discusses about the genetic fuzzy controllers in mobile robotics.

2.1 Stabilizing Control Using Invariant Manifolds

The field of mobile robot control has been the focus of active research in the past decades. Mobile robots, which are omnidirectional or nonholonomic, are highly nonlinear, and especially nonholonomic constraints have motivated the development of highly nonlinear

control techniques. Mobile robots are devices that can move from one place to another autonomously within a predefined workspace to achieve their desired goals. The most popular one among the mobile robots is the wheeled-mobile robot because it is appropriate for typical applications with relatively low mechanical complexity and energy consumption. Figure 2.1 shows the wheeled mobile robots which are commercially available mobile robots for current research.

Despite the simplicity of the kinematic model of a wheeled mobile robot, the design of stabilizing control laws for this system can be considered a challenge due to the existence of nonholonomic constraints. The two principal approaches for the problem of controlling nonholonomic systems have been described in [1]. The first approach uses the open-loop control strategies to generate the feasible trajectories and the second one uses the feedback control strategies to solve the path-following and stabilizing problems. The stabilizing problem consists in finding adequate feedback control laws that allow the mobile robot to reach a desired state starting from any initial state. Moreover, it needs more elaborate nonlinear techniques. The stabilization problem is a challenging one, because it has been proven that the kinematic model of a nonholonomic vehicle is open-loop controllable, but not stabilizable by pure smooth, time-invariant feedback law [2]. This fact makes the control of nonholonomic systems extremely challenging and stimulates researchers to construct time-varying or discontinuous feedback controllers for the control of nonholonomic systems.

Many control strategies, such as smooth time-varying strategies [3], [4] and discontinuous time-invariant control laws have been proposed for various nonholonomic systems [5], [6]. Moreover, Tayebi *et al*. [7] constructed a discontinuous time-invariant feedback control method for n-dimensional nonholonomic chained systems. The switching control and quasi-continuous control based on invariant manifold are proposed for a power system with two inputs and three states or two inputs and n-states [8], [9]. The quasi-continuous exponential stabilizing controller that can be applied to both a kinematic-based model and a dynamic-based model was extended [10] to a double integrator form with two inputs and three states or five states. The switching control based on the invariant manifold assures that all the states smoothly converge to the origin [11]. Moreover, references [12], [13], [14], and [15] also proposed the invariant manifold techniques to control underactuated systems. Lee *et al*. [16] investigated a parking problem with the point stabilization problem of nonholonomic car-like mobile robots. The algorithm is divided into two steps: stabilization to a desired line and stabilization to a desired point. By using a nonlinear state feedback control law, the steering operation is determined.

(a)

(b)

(c)

(d)

Fig. 2.1 Wheeled mobile robots: (a) Pinoneer 3-AT, (b) 4WD aluminum, (c) 3WD 100 mm omni wheel mini, and (d) Ris-RSummit

2.2 Visual Servoing

As mentioned in Section 2.1, the wheeled mobile robot is usually suffered from nonholonomic constraints, which make it more difficult to be controlled. Hence, the motion control problem of wheeled mobile robots can be more challenging compared with that of robot manipulators. In recent years, various approaches for motion control of wheeled mobile robots, such as chained form method [17], smooth time-varying strategies [18], [19], or the combination of them [20] have been proposed by researchers to achieve high-control performance of nonholonomic systems. All above mentioned approaches were developed under the assumption that the robot states can be exactly obtained for the feedback control purpose. However, this assumption is usually not satisfied in most real applications due to the uncertainties in the kinematic model and slippage of the wheels on the ground.

An alternative approach, which can make the feedback loop directly closed at the sensor layer, has been developed to handle the uncertainties in real environments and improve the control performance of mobile robot systems. Such a strategy is known as the sensor based control of mobile robots. According to this motivation, a lot of attention has been focused on the research of visual servoing approaches for mobile robots. In recent years, vision sensor and digital image processing technology become easily available with the development of computer technology. Moreover, vision is an important robotic sensor because it can be used for environmental measurements without physical contact. Visual robot control or visual servoing is a feedback control methodology which uses one or more sensors (cameras) to control the motion of the robot. Specifically, the control inputs for the motors used in robots are produced by processing image data, i.e., they are normally given by extracting contours, features, corners, and other visual primitives. Vision-based control is classified into four groups: position-based, image-based, hybrid, and motion-based control system in [21].

The purpose of visual control in robotic manipulators is to control the pose of the robot's end-effector relative to a target object or a set of target features. In mobile robots, the vision controller's task is to control the vehicle's pose with respect to some landmarks. As shown in Fig. 2.2, there are two basic configurations to perform a visual servoing task in the area of visual servoing for mobile robots as well as robot manipulators. The camera is rigidly attached to the mobile base in the eye-in-hand configuration and the camera is fixed on the ceiling in the fixed camera configuration.

2.2.1 Position-based visual servoing (PBVS)

The PBVS estimates the position and posture of the robot by matching previous knowledge of 3D environmental models with a captured image and controls it directly. Therefore, both of the control objective and the control law design are performed in the 3-D Cartesian space in the PBVS approach. Figure 2.3 shows the block diagram of a position-based visual servoing.

The problems of pose stabilization, following paths, wall following, and vehicle leader-follower were treated by using the position-based visual control in [22], [23], [24], [25], [26] and [27], where the vision systems provide the estimations of the parameters that are needed to implement conventional controllers. The PBVS approach for nonholonomic wheeled mobile robots was presented to reduce the visual servoing task to a control problem in the Cartesian space [28], [29], where a pan-tilt camera was used to increase the degrees of freedom of the camera sensor. A piecewise smooth visual feedback control scheme was proposed to control a nonholonomic cart without the capabilities of dead reckoning [30]. A stable vision-based control scheme for nonholonomic vehicle was developed to keep a landmark in the camera field of view [31]. The metrical information about the feature

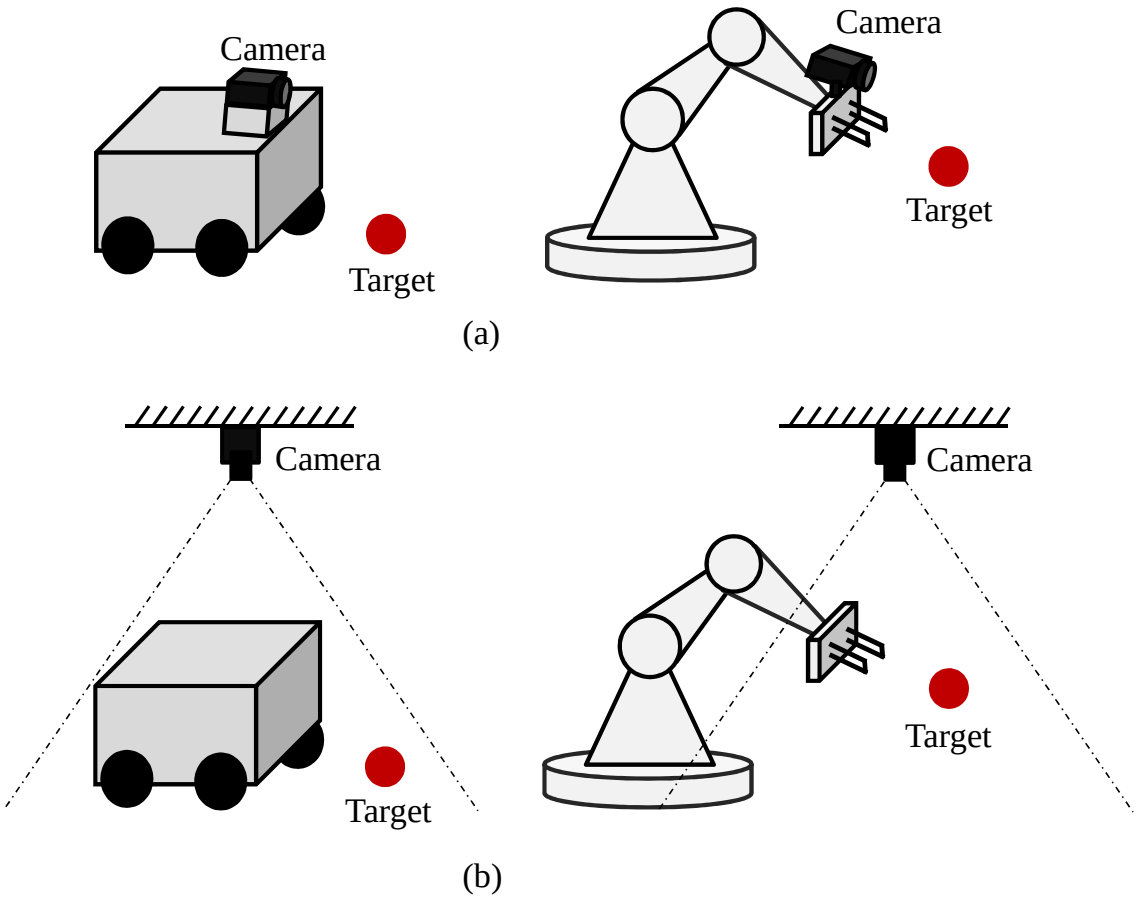


Fig. 2.2 Examples of vision-based robot control: (a) eye-in-hand system and (b) fixed camera system

position with respect to the camera robot frame is required for all these methods to obtain all of the states for feedback control. Thuilot *et al*. [32] proposed a control strategy for keeping a moving target object in a field of vision by controlling a Cartesian robot when using the PBVS approach. Ha *et al*. [33] designed a controller for an eye-in-hand manipulator.

2.2.2 Image-based visual servoing (IBVS)

The IBVS does not need the knowledge of the 3D model of the environment and can operate a robot by controlling only image information which is acquired from a camera image without using the robot position. In the IBVS, both of the control objective and the control law design are directly performed in the image feature parameter space and thus the full model of the object need not be known accurately. Figure 2.4 shows the block diagram of an image-based visual servoing. Compared to PBVS, IBVS can work with faster processing speed and higher

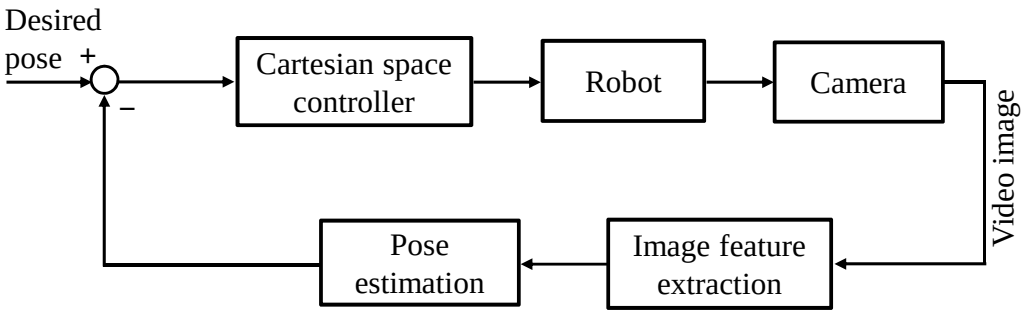


Fig. 2.3 Block diagram of a position-based visual servoing

robustness against camera calibration errors due to the exclusion of the position estimation process [34], [35].

An image-based visual servoing approach for the position control of a nonholonomic mobile robot was proposed in [36] to eliminate the dependence on a priori 3-D knowledge of the scene. In this approach, a discontinuous change of coordinates was used to design a closed-loop stabilizing control law. The image-based visual feedback controller was presented in [37] to control a mobile robot which can track a moving target of interest, where an adaptive back-stepping was developed to handle the unknown height of the target. Without using any a priori knowledge of the 3-D scene geometry, a new two-step IBVS strategy was proposed in [38], where an approximate input-output linearizing feedback is used to align the mobile robot with the goal in the first step and then feature points are used to drive the nonholonomic mobile robot to its desired configuration in the second step. Mariottini *et al*. [39] developed a similar two-step approach using a central catadioptric camera, which can guarantee the global asymptotic stabilization of nonholonomic mobile robots to a desired pose. The control schemes in [38] and [39] drove the robot away from the target while correcting the lateral error and then the robot was moved backward to the desired position to avoid the singularity. The image-based technique has been used in [40], where the differential flatness properties are used to generate the most effective path following strategies. In [41], the image-based approach for the path following problem of a mobile robot was proposed by controlling the shape of the curve in the image plane. The image-based visual servoing techniques for the trajectory tracking and obstacle avoidance of a mobile robot, were developed in [42] and [43]. The image-based fuzzy control approach was developed in [44], where the difference between a reference image and the current image was numerically expressed and directly used by a fuzzy control system using a human-like control law. The algorithm was applied to control an inverted pendulum system.

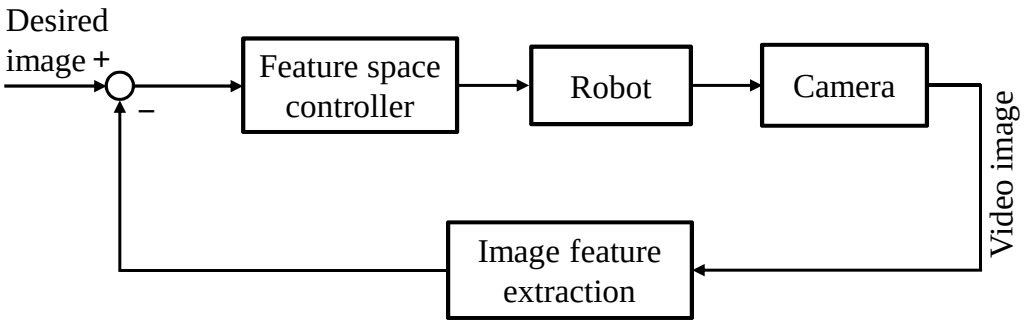


Fig. 2.4 Block diagram of an image-based visual servoing

2.2.3 Hybrid visual servoing

The main drawback of 3D visual servoing is that there is no control in the image which implies that the target may leave the camera field of view. Moreover, a model of the target is needed to compute the pose of the camera. The 2D visual servoing does not explicitly need this model. However, a depth estimation or approximation is necessary in the design of the control law. Furthermore, the main drawback of this approach is that the convergence is ensured only in a neighborhood of the desired position. To exploit the respective advantages of PBVS and IBVS approaches, a hybrid visual servoing scheme was first proposed in [45]. The hybrid approach is also called 2.5-D visual servoing in the literature, and its control objective and control law design are implemented both in the 3-D Cartesian space and in the 2-D image space. To remove the dependence on the exact knowledge of object models, homography calculation and decomposition are carried out to obtain the 3-D information used by the 2.5-D visual servoing scheme. Figure 2.5 shows the block diagram of a hybrid visual servoing.

In [46], Murrieri *et al*. developed a hybrid control approach for the parking problem of a nonholonomic mobile robot, where the visual feedback from low-cost cameras is only used for the stabilization of the vehicle to a given position and orientation. This approach can successfully keep the features in the camera field of view during the visual servoing process. To design the visual servoing scheme for unicycle mobile robots, the 2.5-D control scheme was proposed in [47].

2.2.4 Motion-based visual servoing

The recently developed visual servoing technique was a motion-based visual servoing. It is also called a model-free vision-based control technique because the optical flow in the image can be measured without having any a priori knowledge of the target. In this approach,

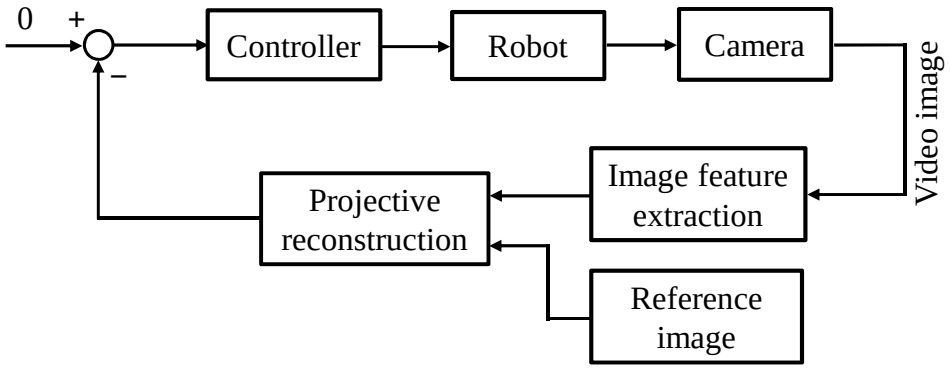


Fig. 2.5 Block diagram of a hybrid visual servoing

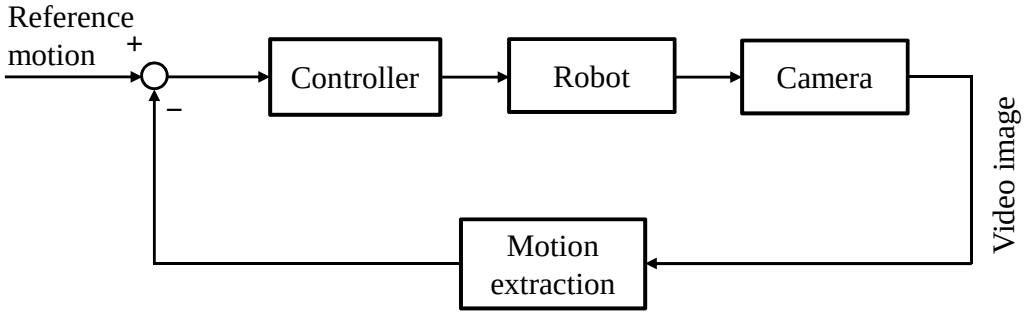


Fig. 2.6 Block diagram of a motion-based visual servoing

the error, which is used to compute the control law, is calculated as a function of the optical flow measured in the image and a reference optical flow which should be obtained by the system. This approach is suitable for the tasks such as the contour tracking [48], camera self-orientation and docking [49], and the application to control an eye-in-hand system in order to position the image plane parallel to a target plane [50]. There is a high servoing rate, which imposes the velocity of the small robot. This fact becomes the major problem of the motion-based visual servoing. However, the developments in the motion estimation algorithms will give the solutions which can make a faster visual control loop. Figure 2.6 shows the block diagram of a motion-based visual servoing.

2.3 Mobile Robot Control Using Fuzzy Logic

Fuzzy control systems have found a wide range of applications in many fields of science and technology such as identification, planning, and model-free control of complex nonlinear systems. Fuzzy control was introduced with the proposal of fuzzy set theory by Prof. Lotfi

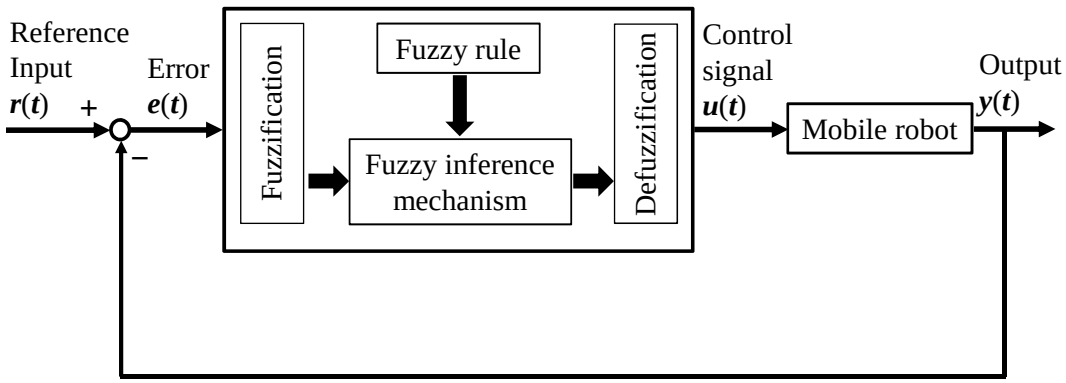


Fig. 2.7 Basic fuzzy mobile robot control loop

Zadeh in 1965 [51]. It is a methodology of intelligent control which mimics human thinking and reacting by using a multivalent fuzzy logic and elements of artificial intelligence [52]. The application of fuzzy logic does not need much detailed knowledge of the system and utilizes the human knowledge and intuition instead of the mathematical knowledge of the system. The human intelligence is easily represented by the fuzzy logic control structure. Most advanced control algorithms in autonomous mobile robots can benefit from fuzzy logic control. The basic structure of a fuzzy mobile robot control loop is shown in Fig. 2.7.

The automated vehicle control schemes were developed by using fuzzy logic control in [53], [54]. For such kind of problem, it was stated that the fuzzy controller from human experts' knowledge is more suitable in [54] and [55]. Moreover, the human experts' knowledge can directly build such fuzzy rules. In [56] [57], the fuzzy logic controller is manually designed, depending on the experiences of the human knowledge. Das and Kar [58] proposed an adaptive fuzzy controller for trajectory tracking of nonholonomic mobile robots. Peri and Simon [59] developed a fuzzy controller to control the motion of the robot along the predefined path. Singh *et al*. [60] proposed an intelligent controller for mobile robot navigation algorithm employing the fuzzy theory in a complex cluttered environment.

A special topic of recent research in the automotive industry is automatic parking control, which is useful not only for autonomous driving vehicles, but also in conventional cars as a parking assistance system. Since the kinematic equations of a nonholonomic mobile robot are nonlinear and time-varying differential equations, it is not possible to find an auto-driver by a traditional control method to maneuver the car to perform the garage parking or parallel parking. On the other hand, a human driver can smoothly park the car into the parking frame by some simple instinct rules without the knowledge of the motion kinematics of the car.

References [61], [62], [63], [64], [65], and [66] presented the garage and parallel parking control which are maneuvered by the fuzzy logic control scheme. Li and Chang [63] proposed

the experimental studies on fuzzy garage-parking control and fuzzy parallel-parking control using a car-like mobile robot, where the CCD camera was used for the overall vision of the car park, but it is an expensive method. For the experimental study of fuzzy garage parking control, Li *et al*. [64] adopted the six infrared sensors to measure the distances between the robot and the surroundings, and [65] used the sensor fusion techniques to combine the ultrasonic sensors, encoders, and gyroscopes with a differential GPS system to detect and estimate the dimensions of the parking lot. The new intelligent auto-parking system was described in [67], where the fuzzy-logic based trajectory generation algorithm was used for parallel parking without collisions. The problem of parallel and diagonal parking of wheeled vehicles was developed in [68], where the fuzzy logic was used to select the most suitable manoeuvre from the solution set according to the environment, dealing with optimality, path tracking performance and collision avoidance trade-off.

Daxwanger [69] proposed a skilled-based visual parking control using neural networks and fuzzy, where two control architectures, the direct neural control and the fuzzy hybrid control, were used to generate the automatic parking commands. A video sensor was used to measure the environment and the control architectures were validated by experiments with an autonomous mobile robot. For parking control, the state evaluation fuzzy control and the predictive fuzzy control were exploited to achieve the drive knowledge in [70]. In [71], the fuzzy traveling control of a mobile robot with the six supersonic sensors has been proposed, where the flush problem was considered. For maneuvering the vehicle in the parking lot, the development of a near-optimal fuzzy controller has been provided in [72]. The membership functions and rules of the fuzzy controller were generated by using the statistical properties of the individual trajectory groups. For parallel parking control of the car-like mobile robot, a fuzzy gain scheduling controller was investigated in [73], where a fuzzy sliding mode controller was firstly used to locally track a typical path. Several typical paths were formed to complete the parallel parking procedure.

2.4 Optimized Fuzzy Controller by Genetic Algorithms

In most fuzzy systems, membership functions, fuzzy reasoning rules and scaling factors are determined through trial and error by human operators, and it takes many iterations to converge to desirable parameters. Many researchers have explored the use of GAs to tune fuzzy logic controllers in order to optimize the parameters of them [74], [75]. The basic concepts of GA were introduced in [76] and his work was developed in the several research works [77]. GA is a robust optimization technique and does not need any prior information

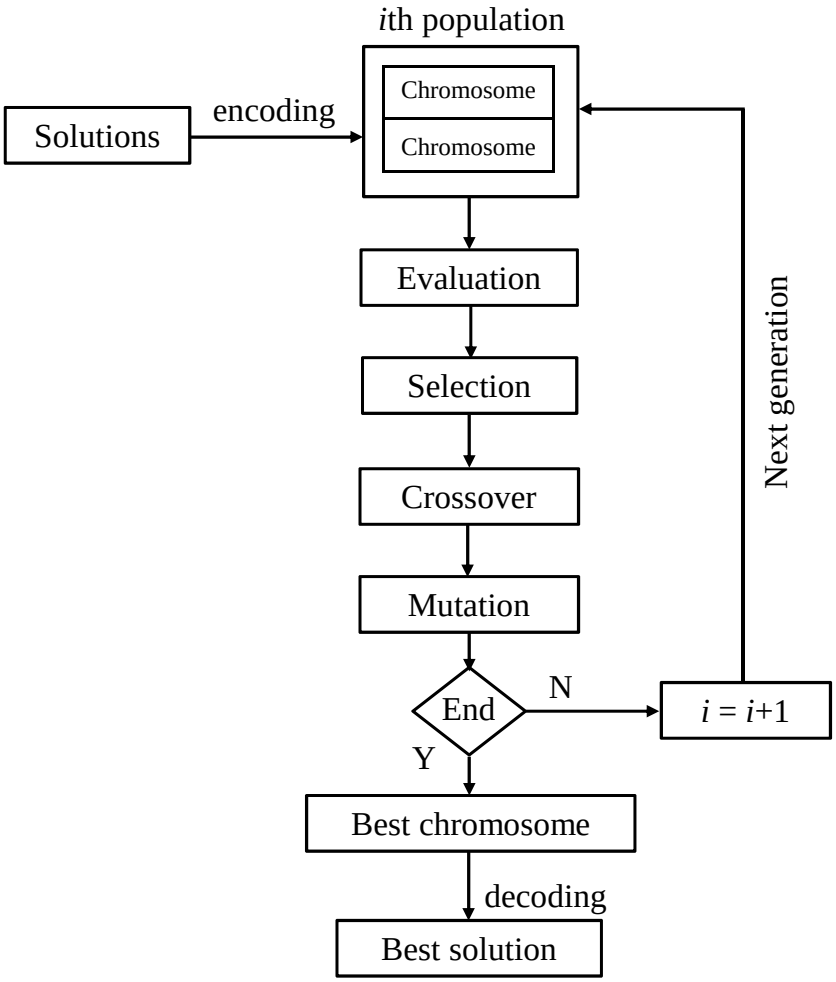


Fig. 2.8 Genetic algorithm flowchart

of the search space which is usually not available for a human designer. The flowchart of the GA can be seen in Fig. 2.8.

Some self-tuning methods using genetic algorithms have been proposed to reduce the required tuning efforts by human operators in [78], [79], [80], and [81]. There are three primary approaches to optimize the fuzzy logic controller using a genetic algorithm. In the first approach, a suitable rule base was formulated by a human designer in advance and the GA was used to tune the parameters of the knowledge base, such as scaling factors or the shapes of the membership functions. In the second approach, the GA was used to generate the entire rule base without using prior knowledge. In the third approach, the GA learned the fuzzy rules as well as the parameters of the membership functions at the same time [82], [83]. To optimize the input and output of the membership functions of the fuzzy controller, the GA was applied in [84]. A new technique for the path planning problem of mobile robot in

static environment was developed by using the combination of fuzzy logic, GA and Neural Networks (NN) in [85].

In order to improve the automatic parallel parking performance, Azadi and Taherkhani [86] optimized the fuzzy membership functions using a GA based on heuristic rules. In [87], the GA was used to optimize the parameters of the membership functions and scaling factors of fuzzy systems for parallel parking. To improve the conventional GAs in designing fuzzy controllers, a new context-dependent coding technique, chromosome reordering operators, and the coevolution of controller testing sets, were developed in [88]. The algorithm was applied to the parallel parking control of a mobile robot.

2.5 Summary

This chapter presented a thorough investigation in the areas of controlling of the car-like mobile robot. Several research gaps in the areas of designing parking controllers were identified through detailed examination.

Chapter 3

Invariant Manifold-based Stabilizing Controllers for Parking Problems

The car-like mobile robots are highly nonlinear, and especially nonholonomic constraints have motivated the development of highly nonlinear control techniques. A system with nonholonomic constraints attracts its attention from the viewpoint of control theory because no conventional control can be applied directly to such a system. Since it cannot be stabilized by a static continuous feedback with gains, there are several control methods by using a chained form up to now. The point-stabilizing controllers for various nonholonomic systems have been proposed in many recent works. However, their research studies are all about the stabilizing problem at the origin.

This chapter presents the new stabilizing controllers based on invariant manifold theory for point-to-point control (parking problems), which allow the car-like mobile robot to reach the desired pose starting from any initial pose. Note here that the combination of the position and orientation is referred to as the pose of the robot. The stability of the proposed control system is analyzed using Lyapunov theory. The car-like mobile robot which is an underactuated system with two inputs is considered as a controlled object. The switching and non-switching control methods based on an invariant manifold theory are proposed for stabilizing it in the desired pose, where a chained form model is assumed to be used as a canonical model.

The rest of this chapter is organized as follows. Section 3.1 describes the problem setting we are concerned with and Section 3.2 provides the construction of invariant manifold theory. Section 3.3 provides the attractive control based on this as well as the switching and non-switching control laws. The several simulation results are presented to demonstrate our theoretical results in Section 3.4. Section 3.5 summarizes the chapter.

3.1 Problem Setting

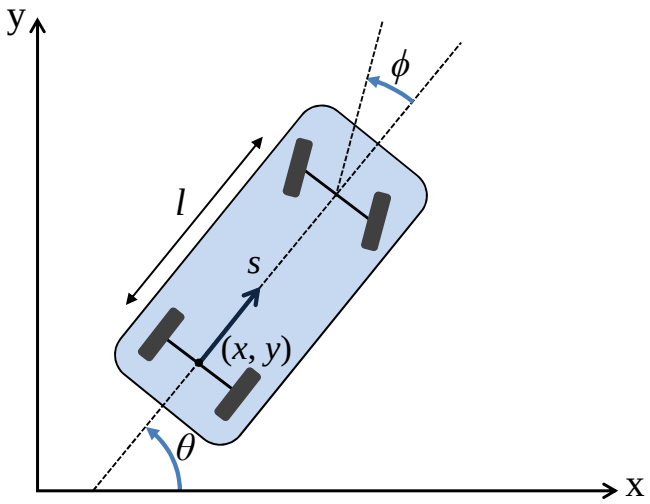


Fig. 3.1 Configuration of a car-like mobile robot

The goal of this chapter is to build the invariant manifold-based stabilizing controllers for parking problems of the car-like mobile robot. Firstly, consider a kinematic model for the car-like mobile robot shown in Fig. 1.

The motion control of this robot can be achieved by dealing with the linear velocity of the point (x, y) denoted as s and the steering velocity of the front wheel denoted as ω. The kinematic model is derived under the assumption of rolling without slippage, on horizontal ground. The configuration of the mobile robot is described by the vector $(x,\ y,\ \theta,\ \phi)^T$, where (x, y) are the coordinates, located at mid-distance of the rear-wheels, θ is the orientation of the vehicle with respect to the x-axis and ϕ is the steering angle, i.e., $\omega = \dot{\phi}$. The kinematic model is represented by:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \frac{1}{l}\tan\phi \\ 0 \end{bmatrix} s + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \omega \tag{3.1}$$

where l is the distance between the front and rear wheels.

The existence of canonical forms for kinematic models of nonholonomic robots is essential for the systematic development of both open-loop and close-loop control strategies. The most useful canonical structure is the chained form. Therefore, the system (3.1) can be transformed into a four-order chained system as discussed in [89]. To obtain the chained form model, taking a transformation form $(x,\ y,\ \theta,\ \phi)$ to the new variables (x_1, x_2, x_3, x_4)

through

$$\begin{aligned} x_1 &= x \\ x_2 &= \frac{\tan\phi}{l\cos^3\theta} \\ x_3 &= \tan\theta \\ x_4 &= y \end{aligned} \tag{3.2}$$

together with the input transformation.

$$\begin{aligned} s &= \frac{u_1}{\cos\theta} \\ \omega &= -\frac{3\cos^2\phi\tan\theta\tan^2\phi}{l\cos\theta}u_1 + l\cos^2\phi\cos^3\theta u_2 \end{aligned} \tag{3.3}$$

The $(2,4)$ chained model becomes

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= x_2 u_1 \\ \dot{x}_4 &= x_3 u_1 \end{aligned} \tag{3.4}$$

where u_1 and u_2 are the control inputs for the canonical model.

Since our aim of this chapter is to derive the controllers for stabilizing the car-like mobile robot in the desired pose, the robot is assumed that it can move from any given initial pose to the desired pose. The desired variables $(x_{1d}, x_{2d}, x_{3d}, x_{4d})$ in the chained form are defined as follows:

$$\begin{aligned} x_{1d} &= x_d \\ x_{2d} &= \frac{\tan\phi_d}{l\cos^3\theta_d} \\ x_{3d} &= \tan\theta_d \\ x_{4d} &= y_d \end{aligned} \tag{3.5}$$

where $(x_d, y_d, \theta_d, \phi_d)^T$ denotes the desired state vector in the original model. Firstly, define the error signal e such as:

$$\begin{aligned} e_1 &\triangleq x_1 - x_{1d} \\ e_2 &\triangleq x_2 - x_{2d} \\ e_3 &\triangleq x_3 - x_{3d} + \alpha \\ e_4 &\triangleq x_4 - x_{4d} + \beta \end{aligned} \tag{3.6}$$

Then, it is easy to note that the time derivatives of the error signals e_1 and e_2 are given by

$$\begin{aligned} \dot{e}_1 &= \dot{x}_1 = u_1 \\ \dot{e}_2 &= \dot{x}_2 = u_2 \end{aligned} \tag{3.7}$$

and the derivative of e_3 becomes

$$\begin{aligned} \dot{e}_3 &= \dot{x}_3 + \dot{\alpha} \\ &= x_2 u_1 + \dot{\alpha} = x_2 u_1 - x_{2d} u_1 + x_{2d} u_1 + \dot{\alpha} \\ &= e_2 u_1 + x_{2d} u_1 + \dot{\alpha} \end{aligned} \tag{3.8}$$

We want to satify $\dot{\alpha} = -x_{2d} u_1$ to obtain $\dot{e}_3 = e_2 u_1$. Then, $u_1 = \dot{e}_1$ and $\dot{\alpha} = -x_{2d}\dot{e}_1$, so that we choose $\alpha = -x_{2d} e_1$ and then

$$e_3 = x_3 - x_{3d} - x_{2d} e_1 \tag{3.9}$$

Along the same way as made in above, it follows that

$$\begin{aligned} \dot{e}_4 &= \dot{x}_4 + \dot{\beta} \\ &= x_3 u_1 + \dot{\beta} = u_1 (e_3 + x_{3d} - \alpha) + \dot{\beta} \\ &= e_3 u_1 + x_{3d} u_1 + x_{2d} e_1 u_1 + \dot{\beta} \end{aligned} \tag{3.10}$$

We choose $\dot{\beta} = -(x_{3d} + x_{2d} e_1) u_1$ from Eq. (3.10) to obtain $\dot{e}_4 = e_3 u_1$. Then $u_1 = \dot{e}_1$ and $\dot{\beta} = -(x_{3d} + x_{2d} e_1)\dot{e}_1$, finally we choose $\beta = -x_{3d} e_1 - \frac{1}{2} x_{2d} e_1^2$ and then

$$e_4 = x_4 - x_{4d} - x_{3d} e_1 - \frac{1}{2} x_{2d} e_1^2 \tag{3.11}$$

In this chapter, three types of switching controllers which perform two steps of control by an invariant manifold, and the non-switching controller are derived for parking problems of the car-like mobile robot. In the switching control technique, attractive control to an invariant manifold is performed first in the first step, and each state on the invariant manifold is stabilized in the second step. In the non-switching approach, the manifold will be automatically attractive by means of an additional state feedback, namely, v.

3.2 Derivation of Invariant Manifold

Firstly, recall the definition of invariant manifold. According to the introduction to a mobile robot control's book [90].

DEFINITION 1. An invariant manifold is an invariant set that happens to be a differentiable manifold. More specially, let $S : R^n \rightarrow R^m$ be a smooth map. A manifold $M = \{\boldsymbol{x} \in R^n : \ S(\boldsymbol{x}) = 0\}$ is invariant for the dynamic system $\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \ \boldsymbol{u})$ if all system trajectories in M at $t \ = \ t_0$ remain in this manifold for all $t \ \geq \ t_0$. In other words, the Lie derivative of S along the vector field f is zero $(L_f \, S(\boldsymbol{x}) = 0)$ for all $\boldsymbol{x} \in M$.

DEFINITION 2. An invariant manifold $M = \{\boldsymbol{x} \in R^n : \ S(\boldsymbol{x}) = 0\}$ is said to be an attractive manifold in an open domain X of R^n, where $X \in M$, if for all such that $t_0 \geq 0$ such that $x(t_0) \in X$, then $\lim_{t \rightarrow \infty} x(t) \in M$ which implies that any $x(t_0)$ outside M is always attracted toward M.

Note, however, that invariant manifold M of $\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \ \boldsymbol{u})$ should be held for stabilizing input $\boldsymbol{u}$ which is assumed to be linear state feedback control $\boldsymbol{u} = -k\boldsymbol{x}$. Our aim is to derive a stable invariant manifold which stabilizes the following controlled object:

$$\begin{aligned} \dot{e}_1 &= u_1 \\ \dot{e}_2 &= u_2 \\ \dot{e}_3 &= e_2 u_1 \\ \dot{e}_4 &= e_3 u_1 \end{aligned} \tag{3.12}$$

and consider a stabilizing control problem such that $\boldsymbol{e}(t) = [e_1, \ e_2, \ e_3, \ e_4]^T$ become zero as $t \rightarrow \infty$. Here, all the states are assumed to be measurable.

To derive a stable invariant manifold, the following linear feedback is applied to Eq. (3.12):

$$\begin{aligned} u_1 &= -ke_1 \\ u_2 &= -ke_2 \end{aligned} \tag{3.13}$$

where $k > 0$ is a linear feedback gain.

The closed-loop time response $e_1(t)$ and $e_2(t)$ are derived as follows:

$$\begin{aligned} e_1(t) &= e_1(0) \, e^{-kt} \\ e_2(t) &= e_2(0) \, e^{-kt} \end{aligned} \tag{3.14}$$

so that

$$\begin{aligned}\dot{e}_3(t) &= -e_2(0)e^{-kt}ke_1(t) = -ke_2(0)e^{-kt}e_1(0)e^{-kt} \\ &= -ke_1(0)e_2(0)e^{-2kt}\end{aligned} \tag{3.15}$$

$$\begin{aligned}e_3(t) &= e_3(0) + \left[\frac{1}{2}e_1(0)e_2(0)e^{-2kt}\right]_0^t \\ &= e_3(0) - \frac{1}{2}e_1(0)e_2(0) + \frac{1}{2}e_1(0)e_2(0)e^{-2kt}\end{aligned} \tag{3.16}$$

and

$$\begin{aligned}\dot{e}_4 &= -\left[e_3(0) - \frac{1}{2}e_1(0)e_2(0) + \frac{1}{2}e_1(0)e_2(0)e^{-2kt}\right]ke_1 \\ &= -\left[e_3(0) - \frac{1}{2}e_1(0)e_2(0) + \frac{1}{2}e_1(0)e_2(0)\times e^{-2kt}\right]ke_1(0)e^{-kt} \\ &= -k\left[e_1(0)e_3(0)e^{-kt} - \frac{1}{2}e_1^2(0)e_2(0)e^{-kt} + \frac{1}{2}e_1^2(0)e_2(0)e^{-3kt}\right]\end{aligned} \tag{3.17}$$

$$\begin{aligned}e_4(t) &= e_4(0) + \left[e_1(0)e_3(0)e^{-kt} - \frac{1}{2}e_1^2(0)e_2(0)e^{-kt} + \frac{1}{6}e_1^2(0)e_2(0)e^{-3kt}\right]_0^t \\ &= e_4(0) - e_1(0)e_3(0) + \frac{1}{2}e_1^2(0)e_2(0) - \frac{1}{6}e_1^2(0)e_2(0) + e_1(0)e_3(0)e^{-kt} \\ &\quad -\frac{1}{2}e_1^2(0)e_2(0)e^{-kt} + \frac{1}{2}e_1^2(0)e_2(0)e^{-3kt}\end{aligned} \tag{3.18}$$

As one candidate of the invariant manifold, $S(\boldsymbol{e})$ can be selected from the constant terms of $e_4(t)$ such as

$$S(\boldsymbol{e}) = e_4(t) - e_1(t)e_3(t) + \frac{1}{3}e_1^2(t)e_2(t) \tag{3.19}$$

In fact, $\dot{S}(\boldsymbol{e})$ can be reduced, under the feedback control given by Eq. (3.12), to

$$\begin{aligned}\dot{S}(\boldsymbol{e}) &= \dot{e}_4(t) - \dot{e}_1(t)e_3(t) - e_1(t)\dot{e}_3(t) + \frac{2}{3}\dot{e}_1(t)e_1(t)e_2(t) + \frac{1}{3}e_1^2(t)\dot{e}_2(t) \\ &= e_3u_1 - u_1e_3 - e_1e_2u_1 + \frac{2}{3}e_1e_2u_1 + \frac{1}{3}e_1^2u_2 \\ &= -\frac{1}{3}e_1e_2u_1 + \frac{1}{3}e_1^2u_2 = \frac{1}{3}ke_1^2e_2 - \frac{1}{3}ke_1^2e_2 \\ &\equiv 0\end{aligned} \tag{3.20}$$

That is $S(\boldsymbol{e})$ = constant. Clearly, once error of the system state reaches a manifold with a constant = 0 (i.e., $S(\boldsymbol{e}) = 0$) then by Eq. (3.19) the entire error of state $\boldsymbol{e}(t) = [e_1,\ e_2,\ e_3,\ e_4]^T$ tends to zero exponentially. Therefore, $S(\boldsymbol{e})$ is an invariant manifold for our controlled system Eq. (3.12) under the feedback control law Eq. (3.13). Furthermore, from Eq. (3.12) and

Eq. (3.13), it follows that

$$\dot{e}_1(t) = -ke_1, \qquad \dot{e}_2(t) = -ke_2$$

so that $e_1(t)$ and $e_2(t)$ are asymptotically stable, i.e., $e_1 \to 0$ and $e_2 \to 0$ as $t \to \infty$. Then, it is true that $e_4 \to 0$ because of $S(\boldsymbol{e}) = 0$. Additionally, it is found from Eq. (3.16) that $e_3 \to 0$ if $e_2(0) = e_3(0) = 0$ for $\forall e_1(0)$.

3.3 Attractive Control to the Manifold

An attractive controller is described here to the manifold derived in the previous section. To obtain the first-step control law for realizing $S(\boldsymbol{e}) = 0$, we select the Lyapunov function of $S(\boldsymbol{e})$ as

$$V(\boldsymbol{e}) = \tfrac{1}{2}S^2(\boldsymbol{e}) \tag{3.21}$$

3.3.1 Switching controller I

First, set the control input as

$$\begin{aligned} u_1 &= fS(\boldsymbol{e}(t))\frac{e_2(t)}{e_1(t)} \\ u_2 &= -fS(\boldsymbol{e}(t)) \end{aligned} \tag{3.22}$$

where f denotes the feedback gain for the attractive control. Then, it follows that

$$\begin{aligned} \dot{V}(\boldsymbol{e}) = S\dot{S} &= S\left(-\frac{1}{3}e_1e_2u_1 + \frac{1}{3}e_1^2u_2\right) \\ &= -\frac{1}{3}S\left(fSe_2^2 + fSe_1^2\right) \\ &= -\frac{1}{3}fS^2W(\boldsymbol{e}) \le 0 \end{aligned} \tag{3.23}$$

where,

$$W(\boldsymbol{e}) \triangleq e_1^2(t) + e_2^2(t) \tag{3.24}$$

Note here that, under the control law of Eq. (3.22), this $W(\boldsymbol{e})$ becomes

$$\begin{aligned} \dot{W} &= 2e_2\dot{e}_2 + 2e_1\dot{e}_1 \\ &= 2e_2u_2 + 2e_1u_1 \equiv 0 \end{aligned} \tag{3.25}$$

Namely, it is seen that $W(\boldsymbol{e}(t)) = W(\boldsymbol{e}(0))$. Therefore, Eq. (3.23) becomes negative definite, so that $S(\boldsymbol{e}) \to 0$ as $t \to \infty$, as long as $W(\boldsymbol{e}(0)) \neq 0$.This controller is not implementable for the case of $e_1(0) = 0$ and $e_2(0) = 0$, in which the controller makes the system unstable.

3.3.2 Switching controller II

As the second candidate, set the control input as

$$\begin{aligned} u_1 &= fS(\boldsymbol{e})\, e_1(t)\, e_2(t) \\ u_2 &= -fS(\boldsymbol{e})\, e_2^2(t) \end{aligned} \tag{3.26}$$

Along the same way as made in above, it follows that

$$\begin{aligned} \dot{V}(\boldsymbol{e}) = S\dot{S} &= S\left(-\frac{1}{3}e_1e_2u_1 + \frac{1}{3}e_1^2u_2\right) \\ &= -\frac{1}{3}S\left(fSe_1^2e_2^2 + fSe_1^2e_2^2\right) \\ &= -\frac{2}{3}fS^2W(\boldsymbol{e}) \leq 0 \end{aligned} \tag{3.27}$$

where,

$$W(\boldsymbol{e}) \triangleq e_1^2(t)\, e_2^2(t) \tag{3.28}$$

Note here that, under the control law of Eq. (3.26), this $W(\boldsymbol{e})$ becomes

$$\begin{aligned} \dot{W} &= 2(e_1^2e_2\dot{e}_2 + e_2^2e_1\dot{e}_1) \\ &= 2(e_1^2e_2u_2 + e_2^2e_1u_1) \equiv 0 \end{aligned} \tag{3.29}$$

Namely, it is seen that $W(\boldsymbol{e}(t)) = W(\boldsymbol{e}(0))$. Therefore, Eq. (3.27) becomes negative definite, so that $S(\boldsymbol{e}) \to 0$ as $t \to \infty$, as long as $W(\boldsymbol{e}(0)) \neq 0$.This controller is not implementable for the case of $e_1(0) = 0$ or $e_2(0) = 0$, in which the controller makes the system unstable.

3.3.3 Switching controller III

As the third candidate, set the control input as

$$\begin{aligned} u_1 &= fS(\boldsymbol{e})\,\frac{e_1(t)}{e_2(t)} \\ u_2 &= -fS(\boldsymbol{e}), \qquad\qquad f > 0 \end{aligned} \tag{3.30}$$

It follows that

$$\begin{aligned}\dot{V}(\boldsymbol{e}) = S\dot{S} &= S\left(-\frac{1}{3}e_1e_2u_1 + \frac{1}{3}e_1^2u_2\right)\\ &= -\frac{1}{3}S\left(fSe_1^2 + fSe_1^2\right)\\ &= -\frac{2}{3}fS^2W(\boldsymbol{e}) \le 0\end{aligned} \tag{3.31}$$

where,

$$W(\boldsymbol{e}) \triangleq e_1^2(t), \qquad e_1(0) \neq 0 \tag{3.32}$$

3.3.4 Non-switching controller

In the previous section, we found that there is a limitation for vehicle orientation due to the incomplete manifold in the system. Therefore we enhance the linear state feedback controller to make the constructed manifold attractive with the method of Tayebi *et al* [7], where note that they originally dealt with the stabilization of a three-wheeled mobile robot at the origin.

In this subsection, we design the stabilizing controller for parking problems which allow the car-like mobile robot to reach a desired pose starting from any initial pose. Here, two manifolds are introduced: one is coming from the non-zero constants of the solution to $e_3(t)$, whereas the other is coming from those of the solution to $e_4(t)$. That is, the following two manifolds are defined:

$$\begin{aligned}S_1(t) &= e_3(t) - \frac{1}{2}e_1(t)e_2(t)\\ S_2(t) &= e_4(t) - e_1(t)e_3(t) + \frac{1}{3}e_1^2(t)e_2(t)\end{aligned} \tag{3.33}$$

where $S_2(t) \equiv S(t)$. It is easy to see that $S_1(t)$ is an invariant manifold under the control law given in Eq. (3.13).

The transformation from (e_1, e_2, e_3, e_4) to (e_1, e_2, S_1, S_2) is a diffeomorphism, so that the stabilization of (e_1, e_2, e_3, e_4) is equivalent to the stabilization of (e_1, e_2, S_1, S_2). Therefore, to stabilize the system (3.12), it is sufficient to lend $(e_1, e_2, e_3,\ e_4)$ into a manifold by an additional state feedback, that is, to make the manifold attractive. In the sequel, a new

controlled objective is reconsidered by

$$\begin{aligned}
\dot{e}_1 &= u_1 \\
\dot{e}_2 &= u_2 \\
\dot{S}_1 &= \frac{1}{2} e_2 u_1 - \frac{1}{2} e_1 u_2 \\
\dot{S}_2 &= -\frac{1}{3} e_1 e_2 u_1 + \frac{1}{3} e_1^2 u_2
\end{aligned} \tag{3.34}$$

The enhanced controller to be used in Eq. (3.34) is assumed to be given by

$$u_1 = -k e_1, \qquad u_2 = -k e_2 + v \tag{3.35}$$

where v is a control term added to the u_2 channel. Then, the closed-loop system is reduced to

$$\begin{aligned}
\dot{e}_1 &= -k e_1 \\
\dot{e}_2 &= -k e_2 + v \\
\dot{S}_1 &= -\frac{1}{2} k e_1 e_2 + \frac{1}{2} k e_1 e_2 - \frac{1}{2} e_1 v \\
&= -\frac{1}{2} e_1 v \\
\dot{S}_2 &= \frac{1}{3} k e_1^2 e_2 - \frac{1}{3} k e_1^2 e_2 + \frac{1}{3} e_1^2 v \\
&= \frac{1}{3} e_1^2 v
\end{aligned} \tag{3.36}$$

The system of the last two equations in Eq. (3.36) can be written as:

$$\dot{\boldsymbol{S}} = E(e_1)\, \boldsymbol{b} v, \qquad \boldsymbol{S} = \begin{bmatrix} S_1 & S_2 \end{bmatrix}^T \tag{3.37}$$

where,

$$E(e_1) = \begin{bmatrix} e_1 & 0 \\ 0 & e_1^2 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{3} \end{bmatrix} \tag{3.38}$$

Now, introducing a transformed variable $\boldsymbol{z}$:

$$\boldsymbol{z} = E^{-1}(e_1)\, \boldsymbol{S} \tag{3.39}$$

which is valid for $e_1 \neq 0$, it is found that

$$\begin{aligned}\dot{\boldsymbol{z}} &= \frac{d}{dt}E^{-1}(e_1)\boldsymbol{S} + E^{-1}(e_1)\dot{\boldsymbol{S}} \\ &= \begin{bmatrix} -\dfrac{\dot{e}_1}{e_1^2} & 0 \\ 0 & -\dfrac{2\dot{e}_1}{e_1^3} \end{bmatrix} E(e_1)\boldsymbol{z} + \boldsymbol{b}v \\ &= \begin{bmatrix} \dfrac{k}{e_1} & 0 \\ 0 & \dfrac{2k}{e_1^2} \end{bmatrix} E(e_1)\boldsymbol{z} + \boldsymbol{b}v\end{aligned} \tag{3.40}$$

that is

$$\dot{\boldsymbol{z}} = A\boldsymbol{z} + \boldsymbol{b}v \tag{3.41}$$

where,

$$A = \begin{bmatrix} k & 0 \\ 0 & 2k \end{bmatrix} \tag{3.42}$$

The system given by Eq. (3.41) can be stabilized by using a feedback controller of the form:

$$v = -\boldsymbol{g}\boldsymbol{z}, \quad \boldsymbol{g} = \begin{bmatrix} g_1 & g_2 \end{bmatrix} \tag{3.43}$$

Using Eq. (3.43), the closed-loop system becomes

$$\dot{\boldsymbol{z}} = A_c\boldsymbol{z} \tag{3.44}$$

where

$$A_c \triangleq A - \boldsymbol{b}\boldsymbol{g} \tag{3.45}$$

In this approach, the manifold will be automatically attractive by means of an additional state feedback, namely, v.

3.4 Simulation Experiments

Simulation results are presented here to demonstrate the ability of the proposed controllers to converge the car-like mobile robot to the desired poses. All the controllers I to IV derived in the previous section were set in the first step, but the linear controllers given in Eq. (3.13) were set in the second step as a switching approach.

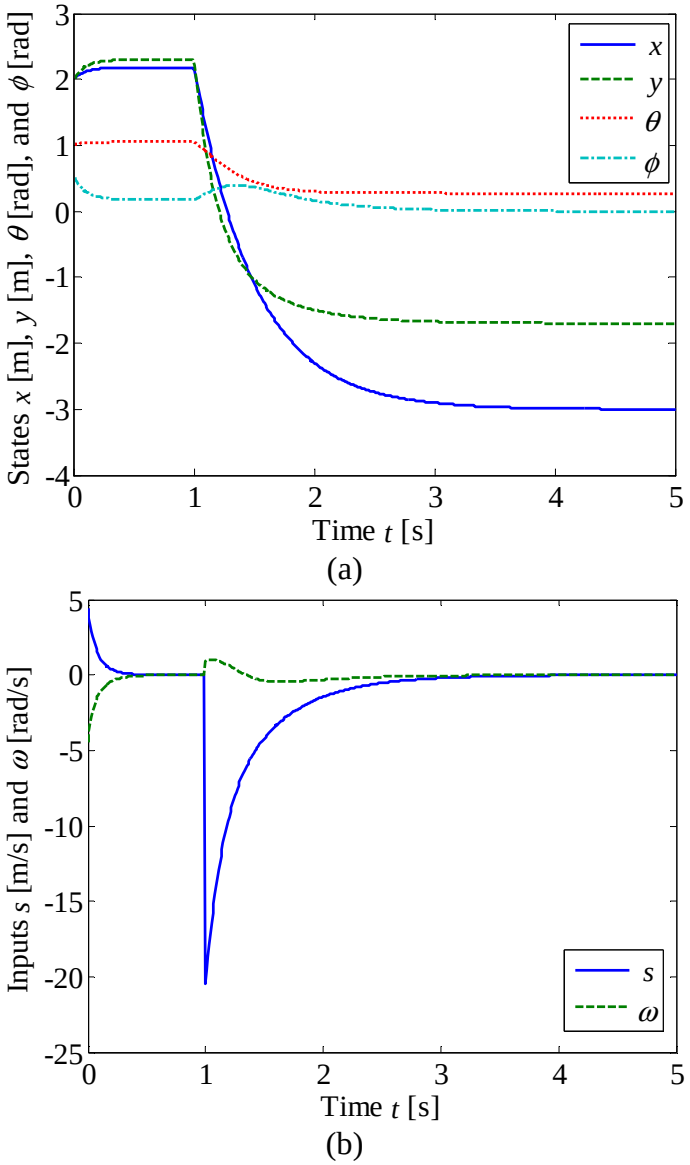


Fig. 3.2 Control results using the switching controller I: (a) state responses and (b) control inputs

As stated the above sections, our objective is to drive the car-like mobile robot from any given initial pose to the desired pose. To compare the effectiveness of each controller for backward parallel parking clearly, the same initial and desired states are set for all controllers. However, for switching controllers I, II and III, the initial states are set as $x(0) = 2$ m, $y(0) = 2$ m, $\theta(0) = 1.0$ rad, and $\phi(0) = 0.523$ rad because the vehicle will go to undesired states if the initial orientation is not equal 1 radian and steering angle equal zero. In this

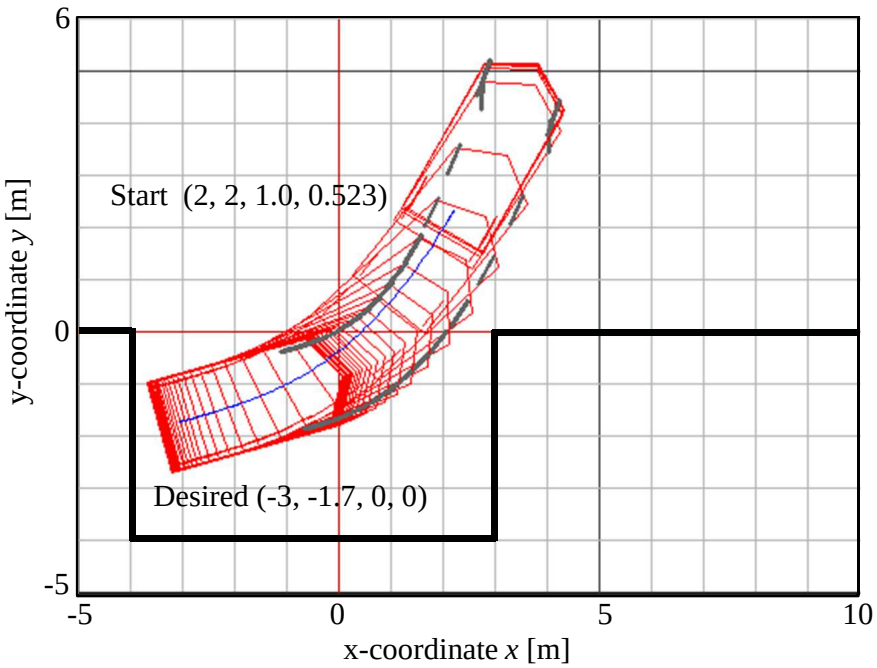


Fig. 3.3 Backward parallel parking result using the switching controller I

simulation studies, the desired states are set as $x_d = -3.0$ m, $y_d = -1.7$ m, $\theta_d = 0$ rad, and $\phi_d = 0$ rad for all controllers.

Figure 3.2 (a) and (b) show the convergence of the whole states of system (3.1) and the time evolution of the control variables for the controllers I. The simulation result of backward parallel parking using the switching controller I is illustrated in Fig. 3.3, where the vehicle orientation converged to a non-desired value about 0.275 rad. Here, it was assumed that the wheelbase of the car-like mobile robot was l = 2.5 [m], the sampling width was Δt = 0.01 [s], the control gains were set to k = 2 and f = 1 and the threshold for the switching ε = 1/1000.

The convergence of the whole states and the time evolution of the control variables for the switching controllers II and III, are shown in Figs. 3.4 and 3.6. The simulation results of backward parallel parking using the switching controllers II and III are illustrated in Figs. 3.5 and 3.7. It is found that all the vehicle positions converged to the desired states ideally. However, the vehicle orientations converged to undesired ones if the initial orientation was not 1 radian.

Finally, the simulation results for the backward parallel parking when using a non-switching rule given in Eqs. (3.35) and (3.43) are given in Figs. 3.8 and 3.9, under the condition of $x(0) = 2$ m, $y(0) = 2$ m, $\theta(0) = 0$ and $\phi(0) = 0$. Here, when defining k = 2, the gain vector was determined as $\boldsymbol{g}$ = [20 63] to make the eigenvalues of $(A - \boldsymbol{b}\boldsymbol{g})$ equal to -2 and -3. It is seen from Figs. 3.8 and 3.9 that this approach assures that all the states

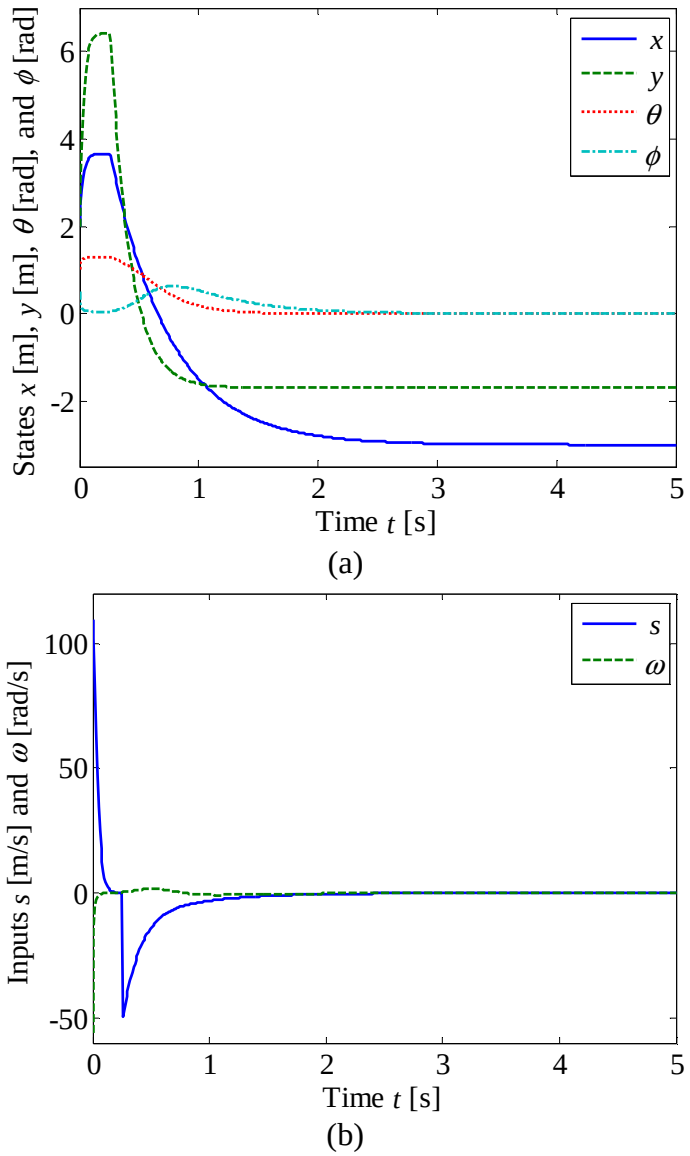


Fig. 3.4 Control results using the switching controller II: (a) state responses and (b) control inputs

converge to the desired states, even if the vehicles starts from any initial state, unlike the switching approach. Note however that this controller everytime gives very large initial ω due to a linear pole-allocation. In fact, ω was assumed to be restricted such as $|\omega| \leq 2.5\pi$ [rad/s].

Our controllers are tested for forward parallel parking too. The comparison of the vehicle orientation errors for forward parallel parking of each controller is shown in Fig. 3.10. As shown in this figure, there was no vehicle orientation error for non-switching approach.

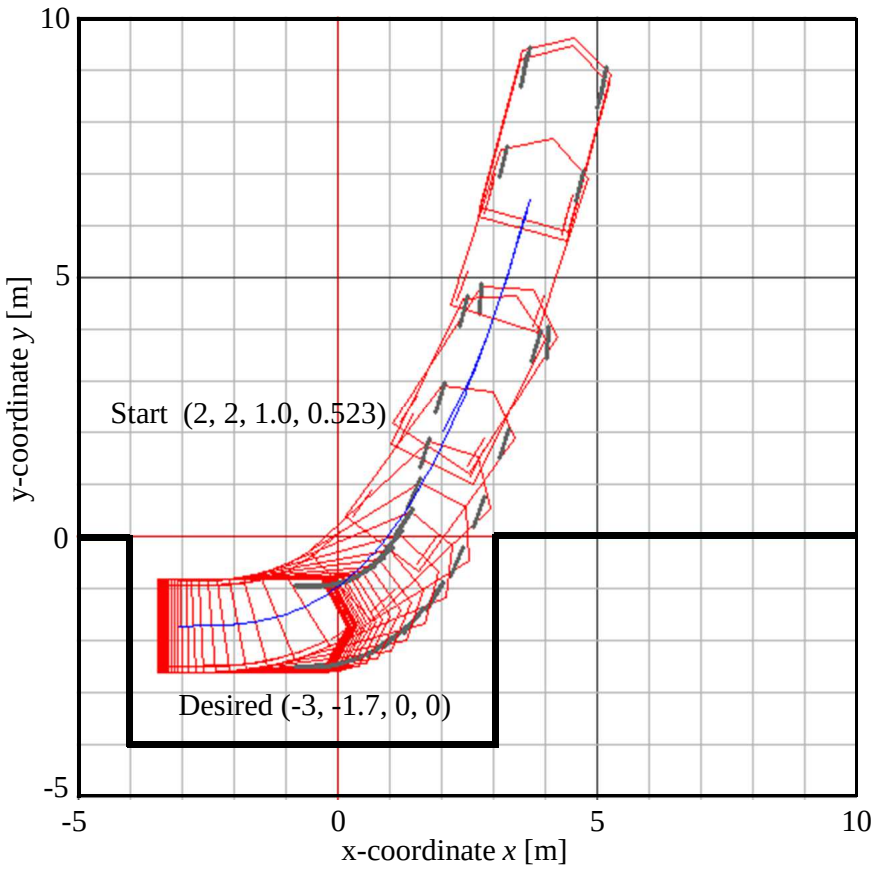


Fig. 3.5 Backward parallel parking result using the switching controller II

The orientation errors for the switching controllers II and III will become zero if the initial orientation is 1 radian. However, it still had the orientation error for the switching controller I, even if the initial orientation was 1 radian. Figure 3.11 shows the comparison of the vehicle orientation errors for backward parallel parking of each controller.

3.5 Summary

The simulation results show the effectiveness of the proposed controllers. When the initial orientation and steering angle were $\theta = 1$ radian and $\phi \neq 0$ respectively, the switching controllers II and III converged the vehicle to the desired states for both forward and backward parallel parking. If not the case its orientation converged to undesired one. It is attributed to the fact that the manifold derived in Eq. (3.19) has an undesirable limitation as shown in Section 3.3. For the switching controller I, there is a small amount of orientation error about -0.275 rad even we set the same initial orientation and steering angle values with the

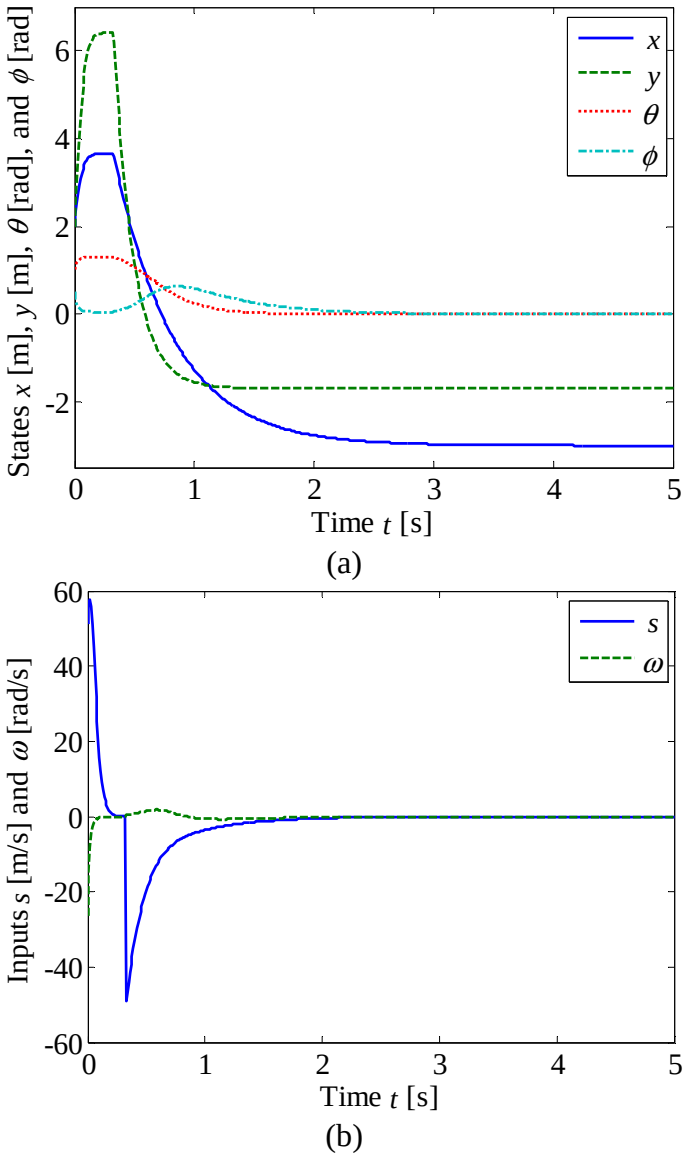


Fig. 3.6 Control results using the switching controller III: (a) state responses and (b) control inputs

switching controllers II and III. On the other hand, it has been proved that the car-like mobile robot was able to be stable and converged to a desired pose for both parking types by using a non-switching controller.

In this chapter, the several stabilizing controllers have been proposed for parking problems of the car-like mobile robot that was an under-actuated system with two inputs and four outputs. Especially, after obtaining a chained form, invariant manifolds were derived based on such a model to design the switching and non-switching controllers for stabilizing of the

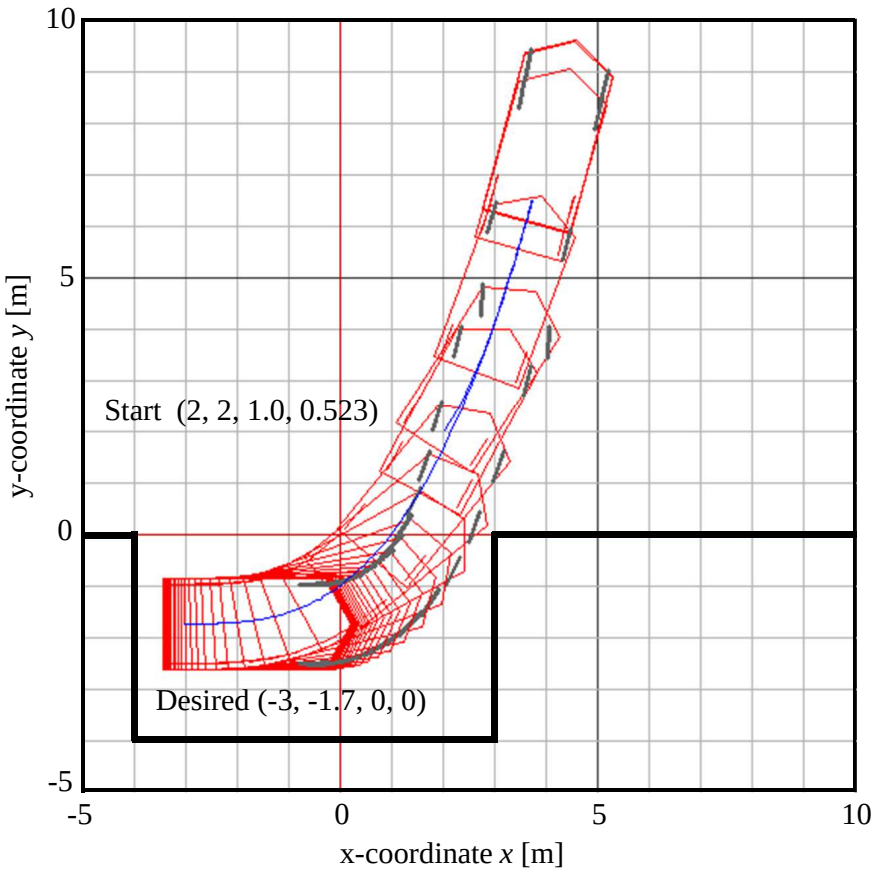


Fig. 3.7 Backward parallel parking result using the switching controller III

car-like mobile robot in the parking system. Some computer simulation results were given to illustrate the effectiveness of the proposed controllers.

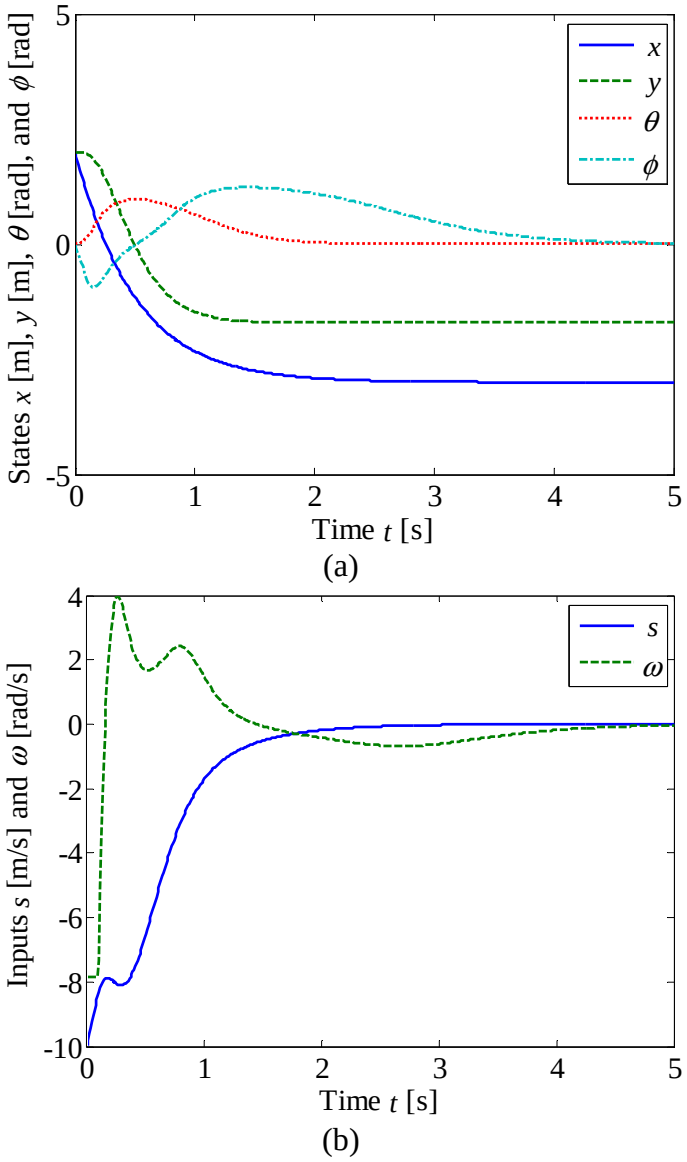


Fig. 3.8 Control results using a non-switching controller: (a) state responses and (b) control inputs

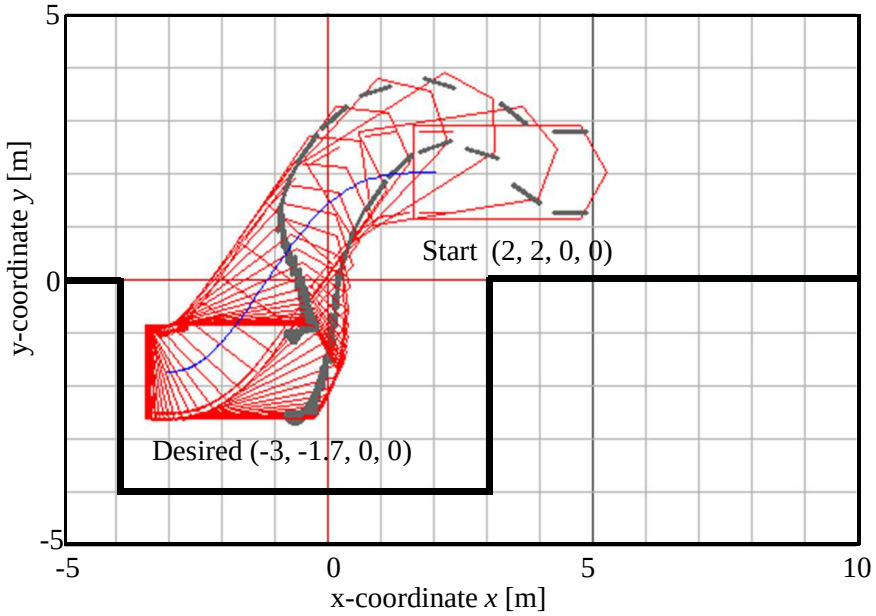


Fig. 3.9 Backward parallel parking result using a non-switching controller

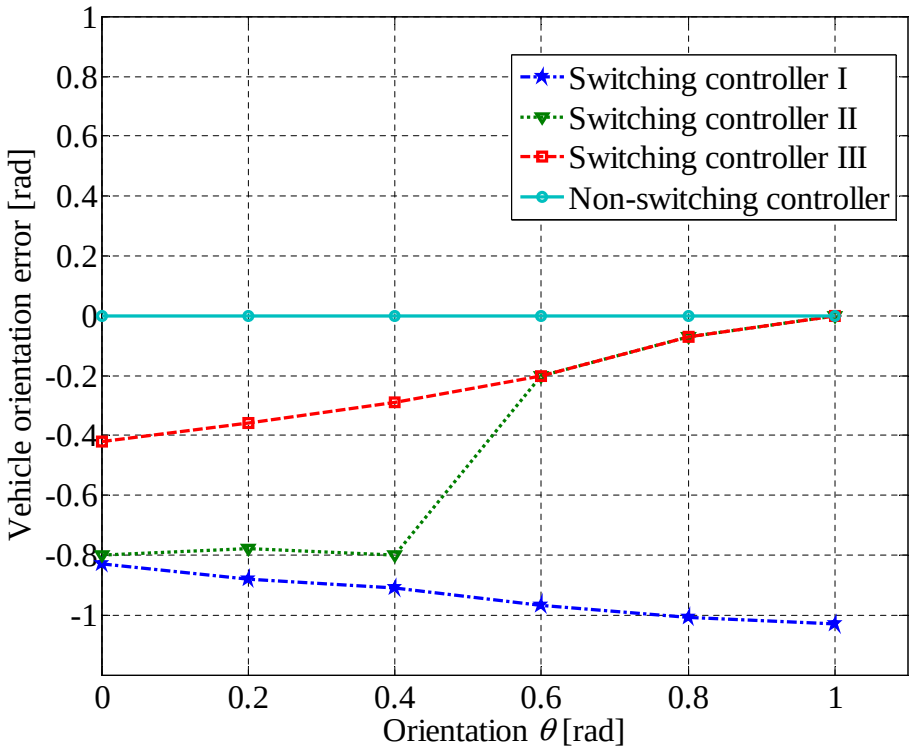


Fig. 3.10 Comparison of vehicle orientation errors for forward parallel parking

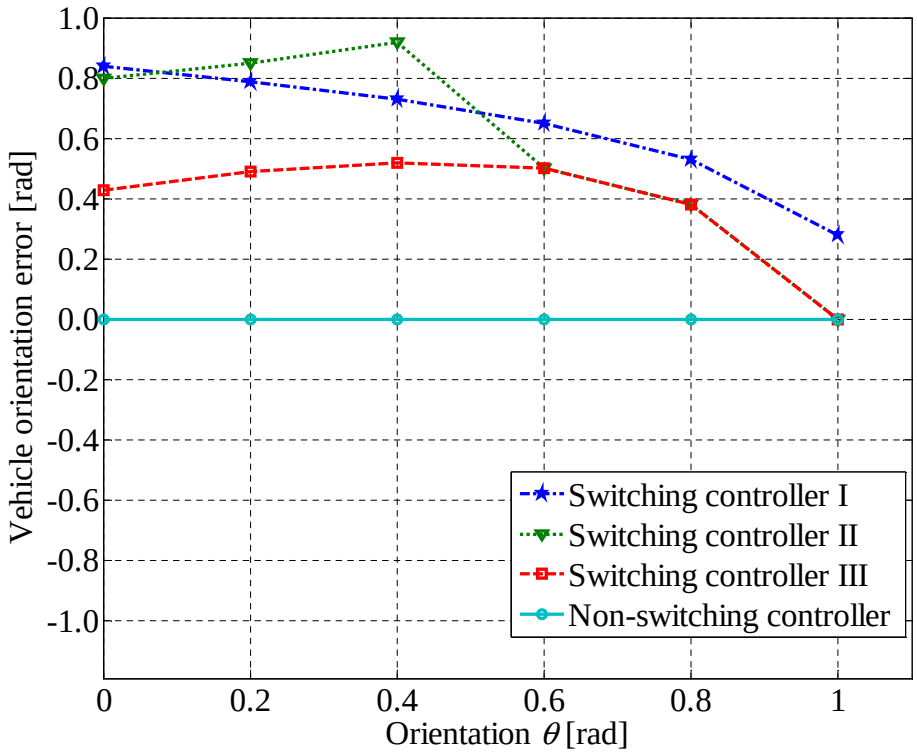


Fig. 3.11 Comparison of vehicle orientation errors for backward parallel parking

Chapter 4

Image-based Fuzzy Parking Control of a Car-like Mobile Robot

The stabilizing controllers for the parking problems of a car-like mobile robot were considered in the previous chapter. However, it was assumed that the robot states can be exactly obtained for the feedback control purpose. In most real applications, it will not be satisfied due to the uncertainties in the kinematic model and slippage of the wheels on the ground. For this reason, we are interested in a vision-based control method in our future tasks.

Automatic parking system is one of the interesting topics in developing intelligent vehicles. However, if the parking controller is not designed properly, it may endanger the vehicle and the driver. This chapter presents a novel image-based fuzzy controller which can automatically drive the car-like mobile robot equipped with a web camera toward a parking frame which is drawn on the floor. The image-based control method can operate a mobile robot by controlling only image information acquired from camera images without using the robot states. The proposed system consists of two parts. In the first part, a robot equipped with a web camera detects a red color parking frame which are drawn on the floor and generates the desired target line to be followed by the robot using a Hough transform. A fuzzy controller is designed with two inputs, which are the slope and intercept of the target line, and one output that is the steering angle of the robot to complete a parking task, in the second part.

The rest of this chapter is organized as follows. In Section 4.1, the problem setting is assigned. In Section 4.2, the parking lines on the floor are detected from a robot equipped with a web camera and a target line for an automatic parking system is generated from a captured image by using an image processing technique with a Progressive Probabilistic Hough Transform (PPHT). Section 4.3 provides the fuzzy controller to follow such a target

line. Section 4.4 shows some simulation and real robot experiments to prove its usability. Section 4.5 summarizes the chapter.

4.1 Problem Setting

The objective of this chapter is to design and implement an image-based fuzzy controller for an automatic parking system of the car-like mobile robot. A red color parking frame drawn on the floor is detected from a robot equipped with a web camera by using image processing. Figure 4.1 shows the experimental overview of the automatic parking system. The world coordinate is represented by x-y coordinate and the image coordinate for the camera is described by u-v coordinate as shown in Fig. 4.2(a) and (b). The image size is W (Width) $\times$ H (Height), in pixel. The following equations represent the center line and target

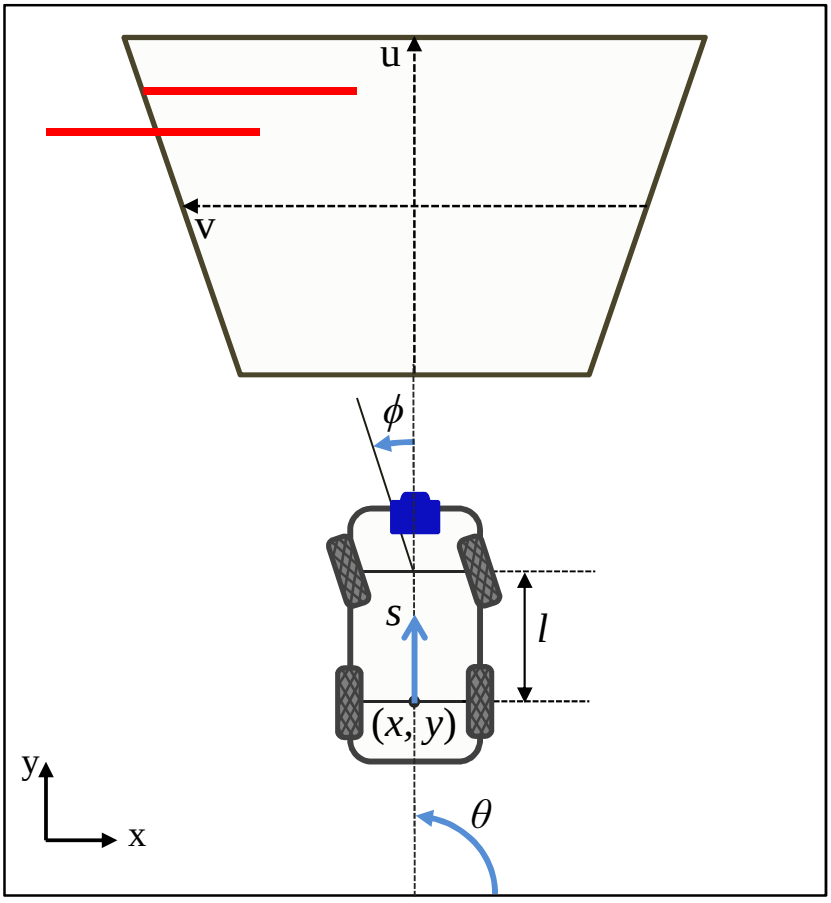


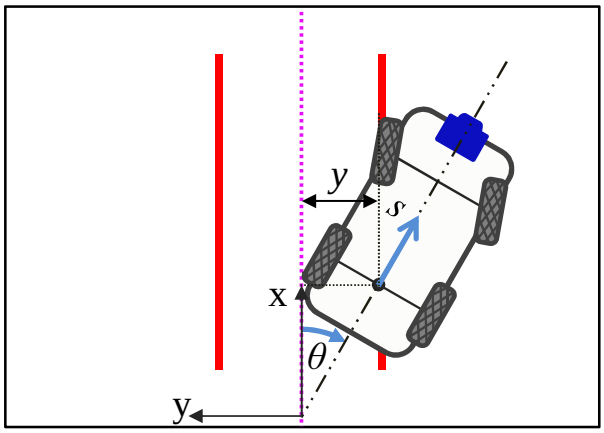
Fig. 4.1 Experimental overview

line of the parking frame on the image coordinates,

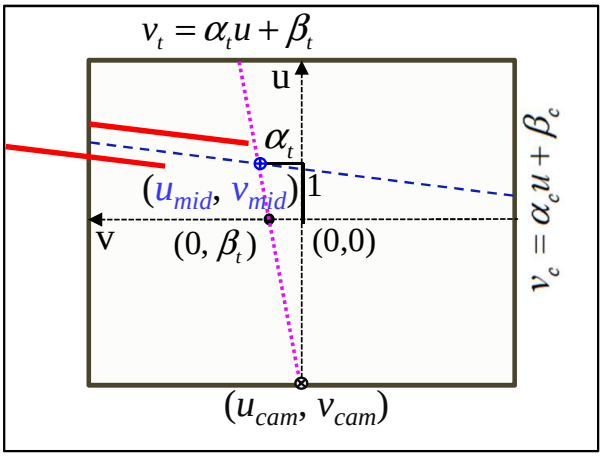
$$v_c = \alpha_c u + \beta_c \tag{4.1}$$

$$v_t = \alpha_t u + \beta_t \tag{4.2}$$

where α_c and α_t are the slopes and, β_c and β_t are the intercepts of the lines. These values can be obtained from an image processing.



(a)



(b)

Fig. 4.2 Coordinate systems used for the automatic parking system: (a) world coordinate and (b) image coordinate

The controlled object is a four-wheel steered mobile robot, whose model is expressed by:

$$\begin{aligned} \dot{x} &= s\cos\theta \\ \dot{y} &= s\sin\theta \\ \dot{\theta} &= \frac{\tan\phi}{l} \end{aligned} \tag{4.3}$$

The state of the robot is described as (x, y, θ), where (x, y) are the coordinates, located at mid-distance of the rear-wheels and θ is the orientation of the robot with respect to the x-axis. The wheelbase is denoted by l and the control inputs for our parking system are composed of the forward speed and the steering angle, (s, ϕ).

Our controller determines the control inputs (s and ϕ) from the image features (α_t and β_t) in the image coordinate to drive the vehicle in an appropriate parking position. To track the target line for our parking system, it can predict that $\alpha_t = 0$ and $\beta_t = 0$ should be accomplished on the image coordinate instead of performing $y = 0$ and $\theta = 0$ on the world coordinate. If the vehicle can follow our desired target line completely, it would say that the robot can park in the parking frame correctly. The block diagram of the image-based fuzzy parking control is shown in Fig. 4.3, where the robot is controlled without referring to its position.

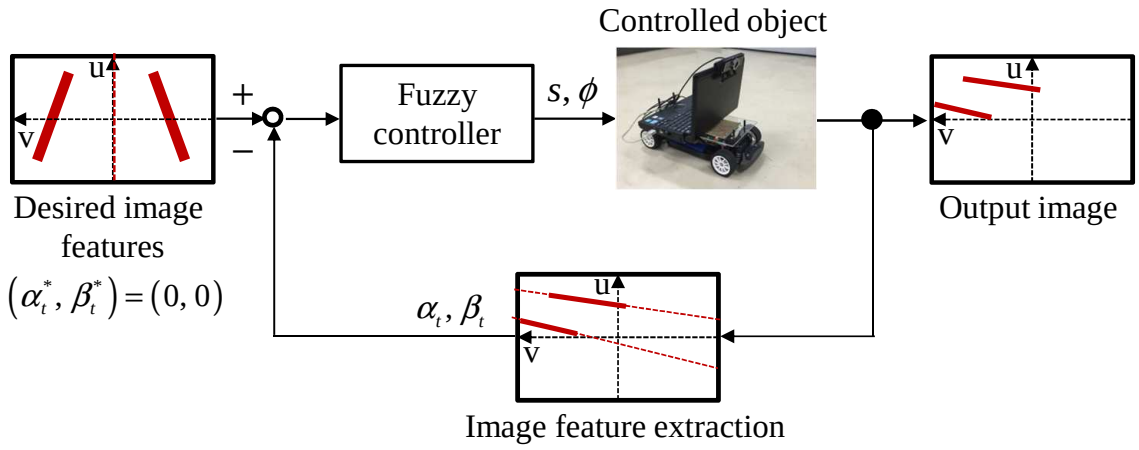


Fig. 4.3 Block diagram of the image-based fuzzy parking control

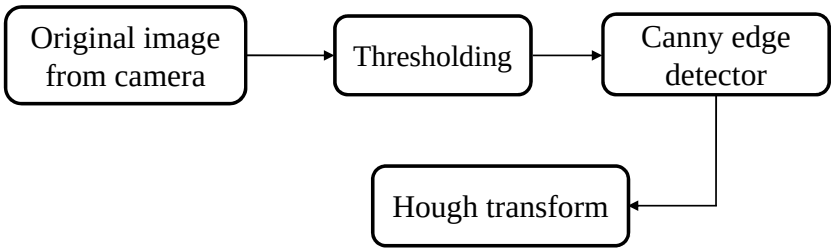


Fig. 4.4 Image processing algorithm

4.2 Generation of the Target Line Based on Image Processing

Nowadays image processing is widely used for detection in intelligent automobiles because it is one of the quick and accurate methods for the detection of parking place. In this section, the steps of image processing on a captured image are explained to detect the parking frame and generate the target line of an automatic parking system. The summarization of image processing which is used in this research is illustrated in Fig. 4.4.

4.2.1 Thresholding of the parking space

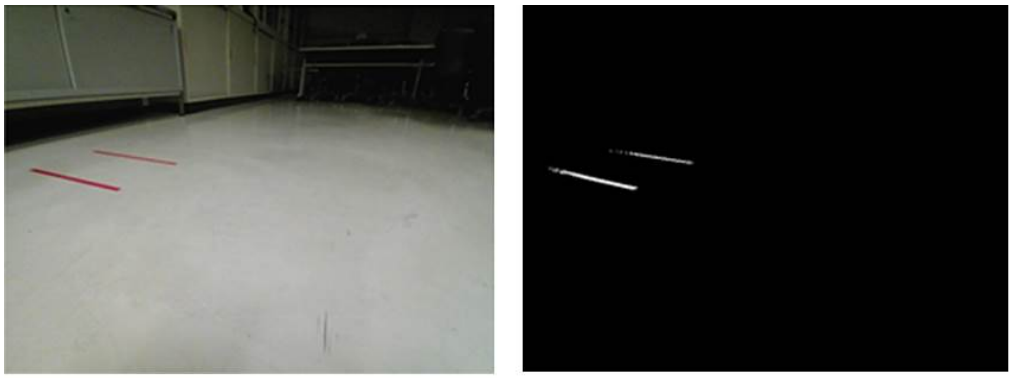
To detect the rectangular parking frame and generate a target line for an automatic parking system, at first, a preprocessing is executed on the captured image. Such a preprocessing includes thresholding. Threshold functions are used for extracting red color and removing a noise on the captured image. The original image and preprocessed image of the parking frame are shown in Fig. 4.5 (a) and (b).

4.2.2 Extraction of the parking lines using Hough transformation

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing [91]. To extract features from binary images, it is useful to be able to find simple shapes, such as straight lines, circles and ellipses in images. In this work, Hough transform is used to extract the lines in the binary image. To detect the lines, the following equations are used:

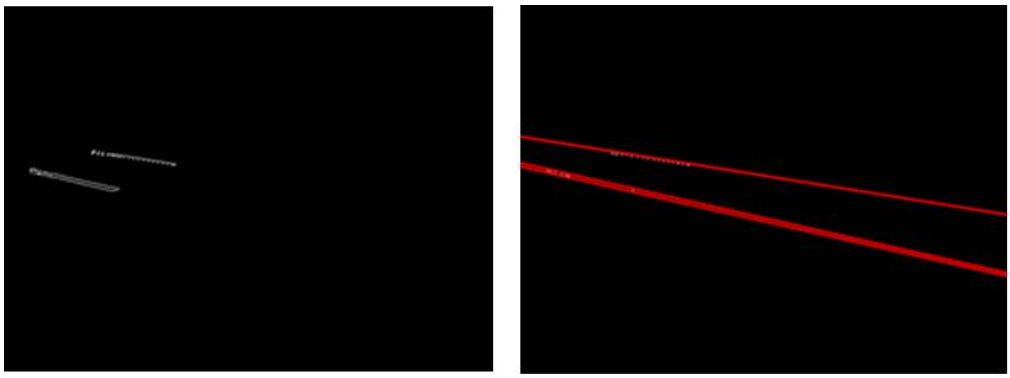
$$v = \alpha u + \beta \tag{4.4}$$

$$u = \rho \cos \delta, \quad v = \rho \sin \delta \tag{4.5}$$



(a) (b)

Fig. 4.5 Image of the parking frame: (a) the original image and (b) the preprocessed image

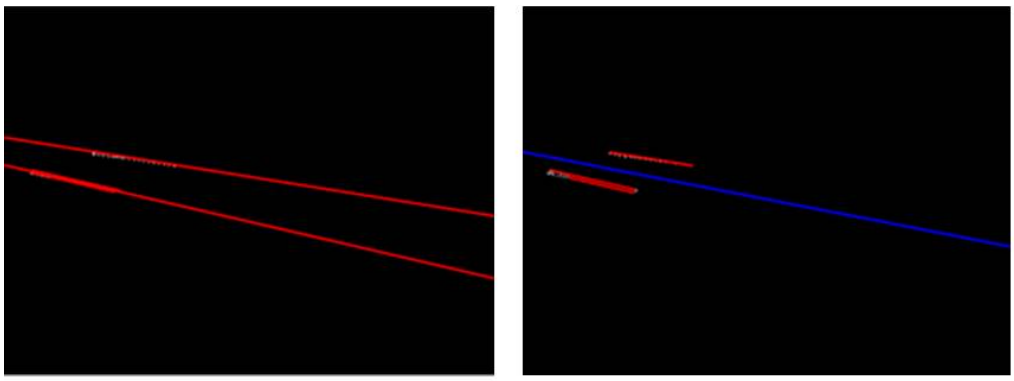


(a) (b)

Fig. 4.6 Canny edge detection: (a) binary image and (b) result image after hough transform

where ρ and δ are the distance from the origin to the line along a vector perpendicular to the line and the angle between the u-axis and this vector as shown in Fig. 4.6 (a). Two of the most efficient algorithms for line detection are the Standard Hough Transform (SHT) and Progressive Probabilistic Hough Transform (PPHT). In the SHT, every line is represented by two floating-point numbers (ρ, δ). The PPHT is more efficient in case if picture contains a few long linear segments. It returns line segments rather than the whole lines. Every segment is represented by starting and ending points, which are illustrated in Fig. 4.6 (b).

In our research, the PPHT was used for extracting the parking lines and finding the endpoints of the parking line segments. The endpoints of the each parking line are defined as (u_0, v_0) and (u_1, v_1). The slope α and intercept β of each line are calculated using the



(a) (b)

Fig. 4.7 Center line of the parking frame: (a) result image after grouping and (b) result of the center line

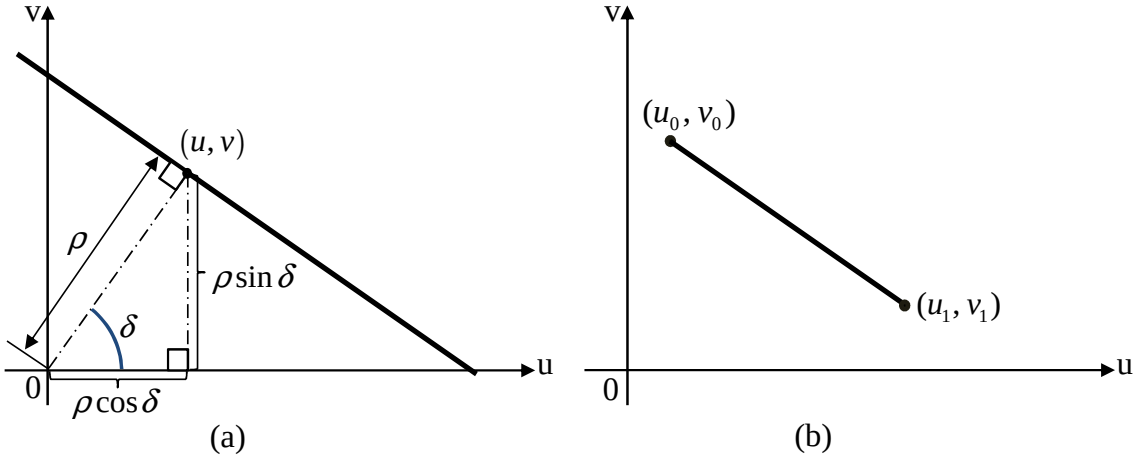


Fig. 4.8 Line representations: (a) when using parameters ρ and δ and (b) when using the starting and ending points ((u_0, v_0) and (u_1, v_1)) of a line on the image plane

following equations:

$$\alpha = \frac{v_1 - v_0}{u_1 - u_0} \tag{4.6}$$

$$\beta = \frac{u_1 v_0 - u_0 v_1}{u_1 - u_0} \tag{4.7}$$

Before Hough transformation is applied, Canny edge detector is used to obtain the edges of the binary image and the resultant image is shown in Fig. 4.7(a). Depending on the slopes and intercepts of those lines, two vertical lines are estimated by grouping them and the results are shown in Figs. 4.7 (b) and 4.8 (a), respectively. The center line of the parking frame

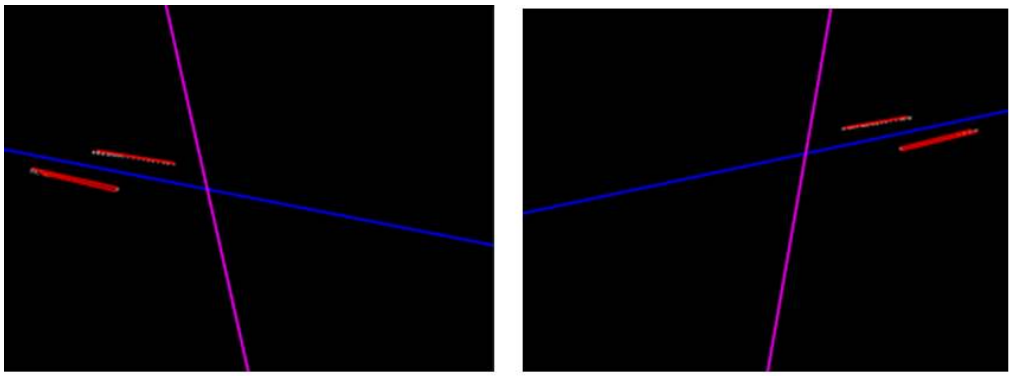


Fig. 4.9 Desired target line started from (a) left side and (b) right side, of the parking frame

denoted as v_c in Section 4.1 is estimated by averaging the slopes and intercepts of the two vertical lines as shown in Fig. 4.8 (b). Finally, the midpoint of such a line is calculated and its coordinate is denoted by (u_{mid}, v_{mid}) in Fig. 4.2(b). The target line of our parking system is constructed depending on the midpoint and a point on the captured image, which is denoted by (u_{cam}, v_{cam}) in Fig. 4.2(b). The slope α_t and intercept β_t of the target line are calculated using the following equations:

$$\alpha_t = \frac{-v_{mid}}{u_{cam} - u_{cam}} \tag{4.8}$$

$$\beta_t = \frac{u_{cam} v_{mid} - u_{mid}}{u_{cam} - u_{mid}} \tag{4.9}$$

The resultant images of the target line are shown in Fig. 4.9 (a) and (b).

4.3 Fuzzy Parking Control

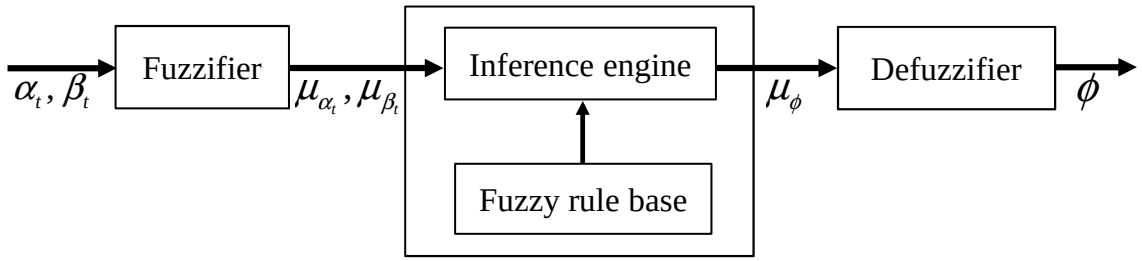


Fig. 4.10 Structure of the image-based fuzzy controller

In this section, a model free fuzzy controller is designed to combine the behavior of the robot with the image feature. It is difficult to derive the relationship function between the state variables (α_t, β_t) on the image coordinate and the control input (ϕ), because it depends on the depth of each pixel. For this reason, an image-based fuzzy controller is designed by using so called min-max centroid method [92]. To park the vehicle in an appropriate parking position, the proper steering angle should be determined according to some image features of the captured image. The structure of the image-based fuzzy controller is shown in Fig. 4.10. In this chapter, the two-input-single-output fuzzy logic control scheme is derived to command the steering angle of the robot according to the skills of an experienced human driver. The fuzzy controller determines the control inputs through mainly three steps:

(i) Fuzzification of state varables

(ii) Calculation of grade of each rule and

(iii) Defuzzification of input values

4.3.1 Fuzzification of state variables

The first step in building our image-based fuzzy control system is the fuzzification of the input and output variables. The process of transforming the crisp input values into the linguistic values is called fuzzification. To complete the fuzzification process, it needs to consider the following two steps:

(i) Step 1: Input values are translated into linguistic concepts, which are represented by fuzzy set.

(ii) Step 2: Membership functions are applied to the measurements, and the degree of membership is determined.

In this study, the state variables to be used as inputs for the controller are the slope and intercept of the target line and the output of the controller is the steering angle of the robot. The input variables α_t and β_t are decomposed into three fuzzy partitions with triangular membership functions, and the output variable ϕ is the fuzzy singleton-type membership function with three partitions. The partitions and the shapes of the membership functions are shown in Figs. 4.11(a) and (b), where fuzzy term sets are denoted by N (Negative), ZO (Zero) and P (Positive). The range of each membership function is [0, 1]. The μ_{α_t} and μ_{β_t} represent the membership functions to calculate the grades of states, α_t and β_t. The membership function to calculate the grade of the output, ϕ is represented as μ_ϕ.

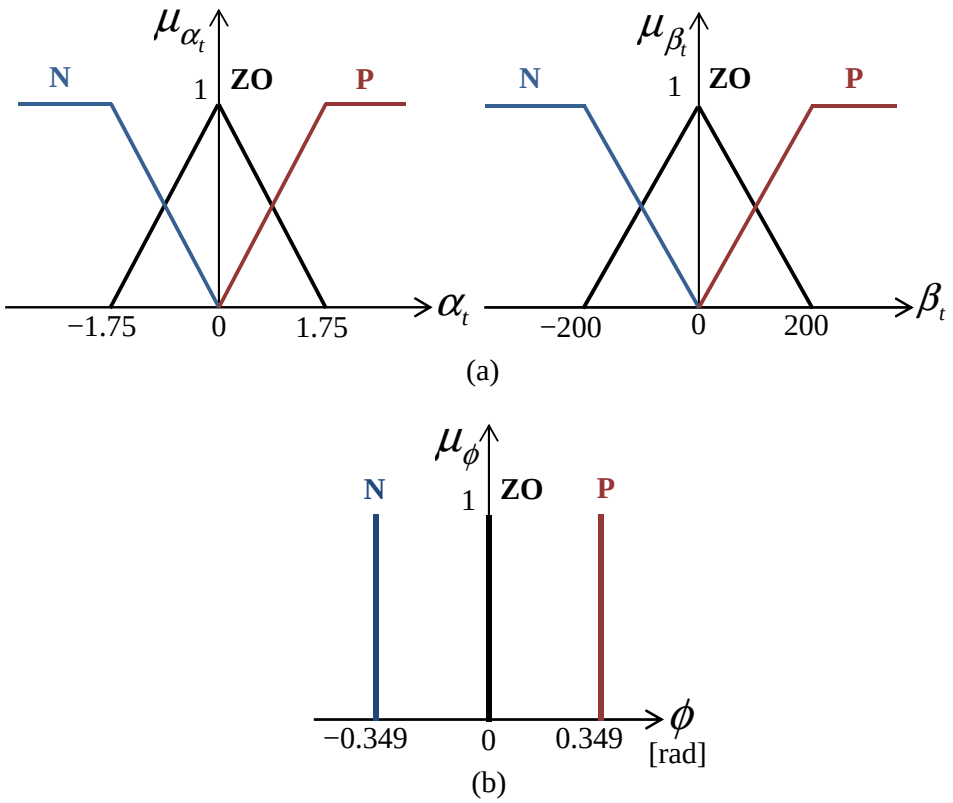


Fig. 4.11 Membership functions: (a) for the input variables α_t and β_t, and (b) for the output variable ϕ

4.3.2 Calculation of grade of each rule

The fuzzy reasoning rules for the steering angle ϕ of an automatic parking system are summarized in Table 4.1. For the control rules, the desired control inputs are set in the form of if-then rules manually, depending on the use of fuzzified state variables. For example, one rule is taken to explain. The rule: "if α_t is N and β_t is N, then ϕ is N" represents that one should turn the steering angle to the right to go along the target line. The adaptability of the control rules $h_{\mathrm{R_i}}$ (i = 1, . . . , 9) are obtained from the following equation as the minimum value among the degrees of truth of the fuzzified state variables related to the rule:

$$\begin{aligned} h_{\mathrm{R}_1} &= \min\{\mu_{\alpha_\mathrm{t},\mathrm{N}}(\alpha_\mathrm{t}), \mu_{\beta_\mathrm{t},\mathrm{N}}(\beta_\mathrm{t})\} \\ h_{\mathrm{R}_2} &= \min\{\mu_{\alpha_\mathrm{t},\mathrm{N}}(\alpha_\mathrm{t}), \mu_{\beta_\mathrm{t},\mathrm{ZO}}(\beta_\mathrm{t})\} \\ &\vdots \\ h_{\mathrm{R}_9} &= \min\{\mu_{\alpha_\mathrm{t},\mathrm{P}}(\alpha_\mathrm{t}), \mu_{\beta_\mathrm{t},\mathrm{P}}(\beta_\mathrm{t})\} \end{aligned} \tag{4.10}$$

Table 4.1 Fuzzy rules for the steering angle, ϕ

No.	α_t	β_t	ϕ
R_1	N	N	N
R_2	N	ZO	N
R_3	N	P	ZO
R_4	ZO	N	N
R_5	ZO	ZO	ZO
R_6	ZO	P	P
R_7	P	N	ZO
R_8	P	ZO	P
R_9	P	P	P

where $\mu_{\alpha_t,\mathrm{N}}(\alpha_t)$ represents the membership function to calculate the grade of state α_t when α_t = N. It should be noted that the graph of the membership functions for α_t in Fig. 4.11 includes three functions, such as $\mu_{\alpha_t,\mathrm{N}}(\alpha_t)$, $\mu_{\alpha_t,\mathrm{ZO}}(\alpha_t)$ and $\mu_{\alpha_t,\mathrm{P}}(\alpha_t)$, and other graphs include some membership functions in a similar way.

The grades for the fuzzy rules are used to calculate the grades of conformity with consequents for the control inputs. For the example of representation of the grades, the grade of conformity with the consequent "ϕ = N" is written as $\mu_{\phi,\mathrm{N}}$. The grades for the consequents are calculated based on the grades for the corresponding fuzzy rules as follows:

$$\begin{aligned}
\mu_{\phi,\mathrm{N}} &= \max\{h_{\mathrm{R}_1}, h_{\mathrm{R}_2}, h_{\mathrm{R}_4}\} \\
\mu_{\phi,\mathrm{ZO}} &= \max\{h_{\mathrm{R}_3}, h_{\mathrm{R}_5}, h_{\mathrm{R}_7}\} \\
\mu_{\phi,\mathrm{P}} &= \max\{h_{\mathrm{R}_6}, h_{\mathrm{R}_8}, h_{\mathrm{R}_9}\}
\end{aligned} \tag{4.11}$$

Figure 4.12 shows the min-max method for three rules involving the two input and single output variables.

4.3.3 Defuzzification of input values

Finally, the control inputs on the fuzzy sets are transformed into real numbers through a defuzzification. Defuzzification is the reverse process of fuzzification. In the other words, it is a transformation from the "fuzzy world" to the "real world." Some example methods of defuzzification are as follows:

(i) Max-membership method: The method chooses the elements with maximum value.

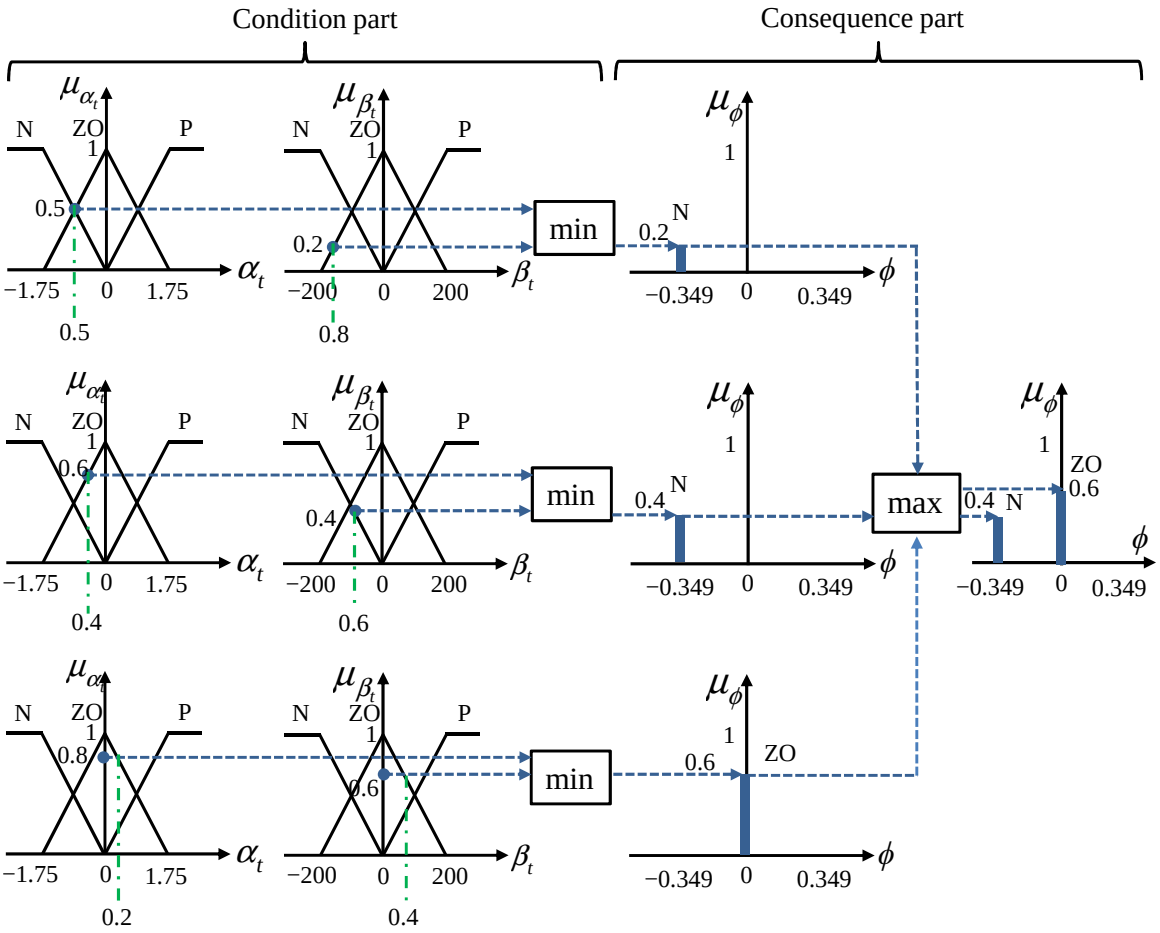


Fig. 4.12 Min-max algorithm

(ii) Centroid method: This method finds the center point of the weighted mean of the output fuzzy region

(iii) Weighted average method: Assigns weight to each membership function in the output by its respective maximum membership value.

In this chapter, the defuzzification is performed by using the weight average method such as

$$\phi = \frac{-0.349\mu_{\phi,\mathrm{N}} + 0.\mu_{\phi,\mathrm{ZO}} + 0.349\mu_{\phi,\mathrm{P}}}{\mu_{\phi,\mathrm{N}} + \mu_{\phi,\mathrm{ZO}} + \mu_{\phi,\mathrm{P}}} \tag{4.12}$$

where $\mu_{\phi,\mathrm{N}}$ denotes the grade of conformity with the consequent "ϕ = N ."

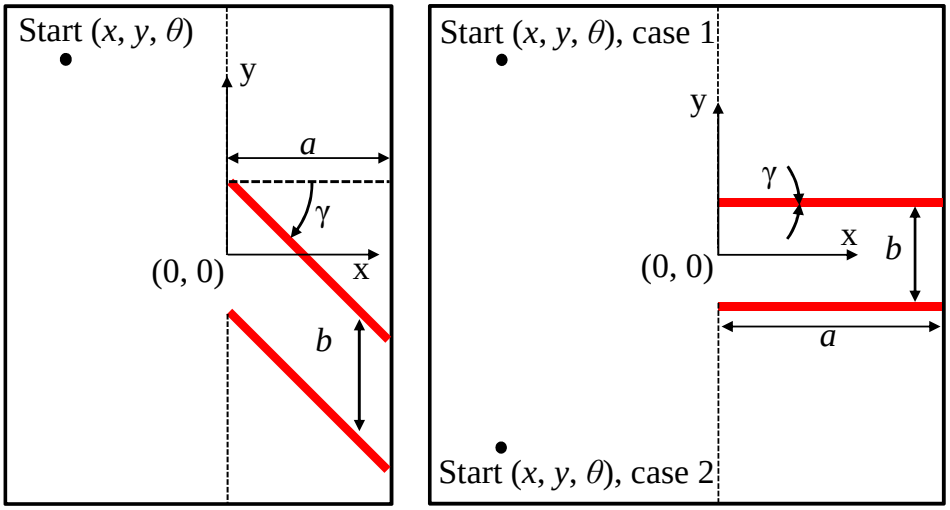


Fig. 4.13 Notations for the angle and perpendicular parking

Table 4.2 Parameters of parking frame and initial postures of the vehicle

	a [m]	b [m]	γ [deg]	x [m]	y [m]	θ [deg]
Angle parking	0.50	0.30	−45	−0.4	0.6	−90
Perpendicular parking (case 1)	0.56	0.27	0	−0.5	0.5	−90
Perpendicular parking (case 2)	0.56	0.27	0	−0.5	−0.5	90

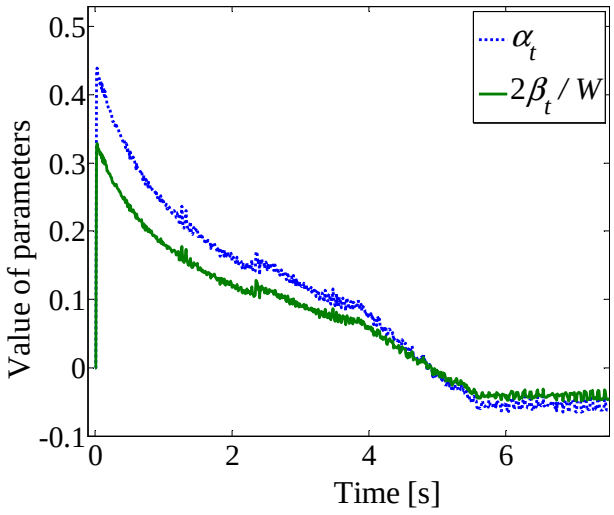


Fig. 4.14 History of parameters related to angle parking

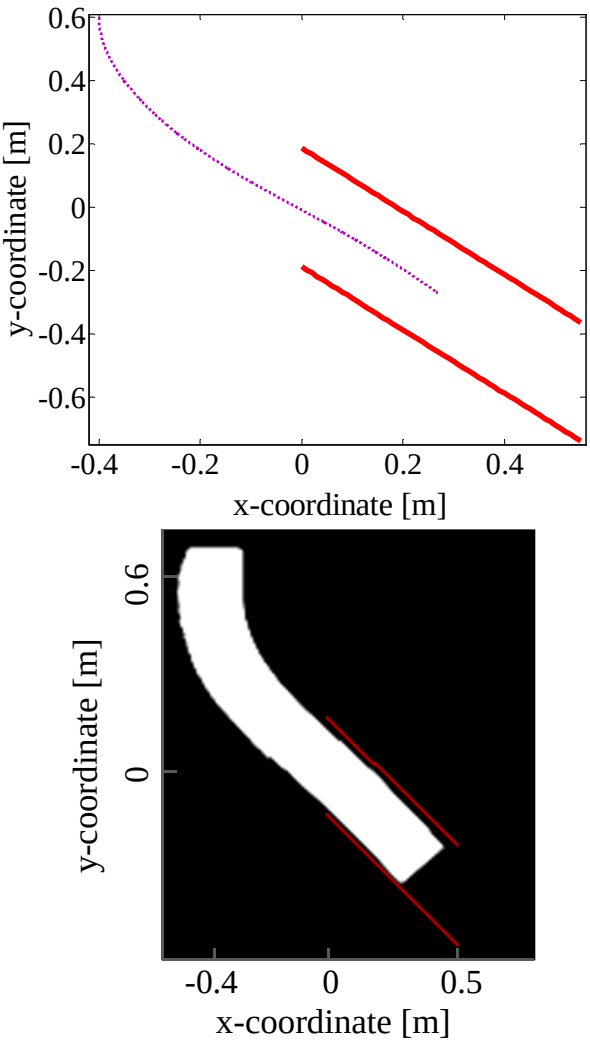


Fig. 4.15 Simulation result of angle parking

4.4 Experiments

In this section, the simulation and real robot experiments were conducted to confirm that the designed image-based fuzzy controller was able to park the vehicle to the appropriate parking position.

4.4.1 Simulation experiments

The simulation studies are implemented to confirm that the designed image-based fuzzy controller can accomplish the automatic angle and perpendicular parking of the car-like mobile robot. The corresponding notations are shown in Fig. 4.13, where the red lines

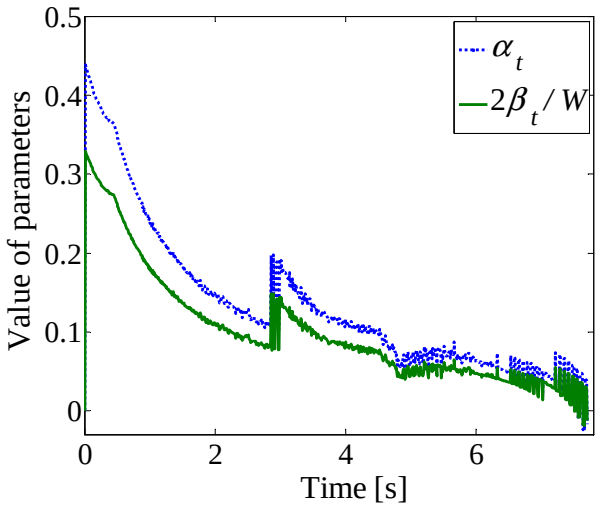


Fig. 4.16 History of parameters related to perpendicular parking, case 1

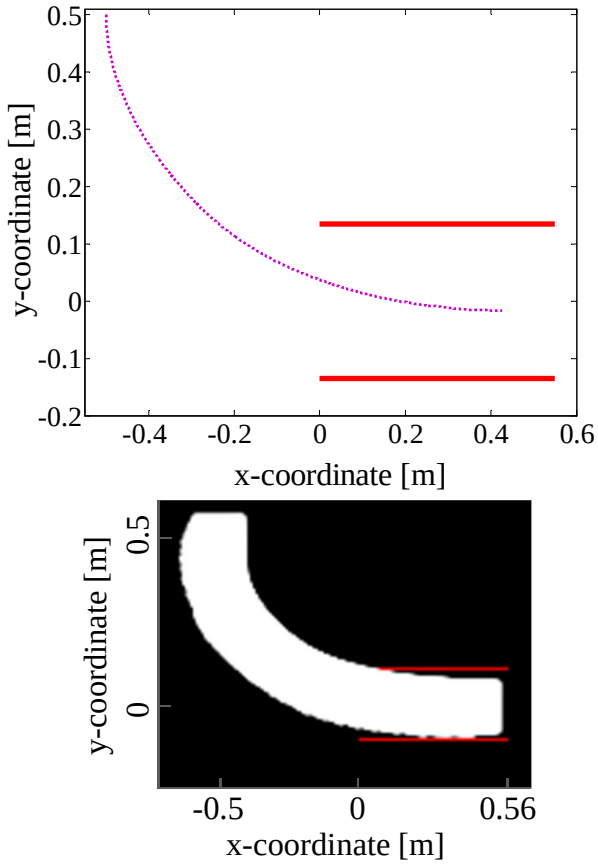


Fig. 4.17 Simulation result of perpendicular parking, case 1

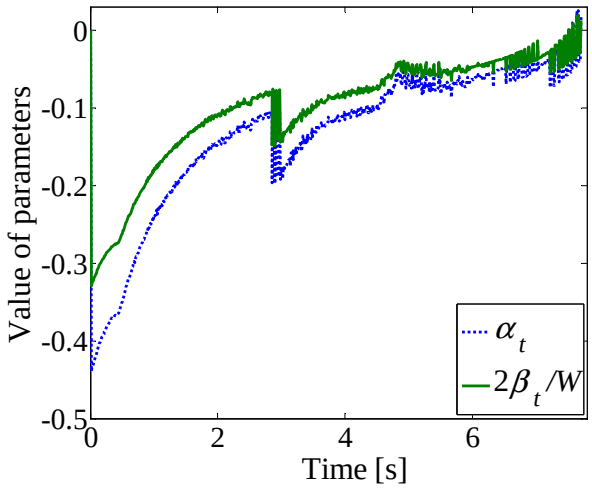


Fig. 4.18 History of parameters related to perpendicular parking, case 2

represent the parking frame. The parameters of parking frame and initial postures of the vehicle for two types of parking system, are summarized in Table 4.2, where (x, y) are the coordinates, located at the mid-distance of the rear-wheels and θ is the orientation of the robot with respect to the x-axis.

The controlled object is a front-wheel steering and rear-wheel drive type vehicle which has the following dimensions: length $l = 37$ [cm], width $W = 18$ [cm], and the wheelbase $l = 26$ [cm]. The working volume of the steering angle was -0.349 rad $< \phi < 0.349$ rad. The camera was mounted on the robot with its height of 20 [cm] and the direction of 60 [deg] to the front from the vertical downward. The camera's focal distance is 0.02 [m] and its angle of view is 2.09 [rad] in a horizontal plane. The value of the parameters on the captured images and the simulation results for the angle parking are shown in Figs. 4.14 and 4.15. The simulation results for the perpendicular parking of both cases are illustrated in Fig. 4.16 to Fig. 4.19. In Figs. 4.16 and 4.18, the parameter values of the captured image are shown for these two initial postures. According to the results, there are only the small amounts of the orientation errors about -0.001 [rad] and 0.001 [rad] for these two initial postures.

The image-based fuzzy parking controller that use the 25 fuzzy control rules, was designed in this chapter and the control rules are shown in Table 4.3. The membership functions for the inputs and output are illustrated in Fig. 4.20(a) and (b), where fuzzy sets NB, NS, ZO, PS and PB represent negative big, negative small, zero, positive small and positive big, respectively. The controller was tested for the perpendicular parking system. The initial condition was set with the same condition of perpendicular parking (case 1) in Table 4.2. The parameter value on the captured images and the simulation results for our parking system are shown in Figs. 4.21 and 4.22. According to the results, there is only the

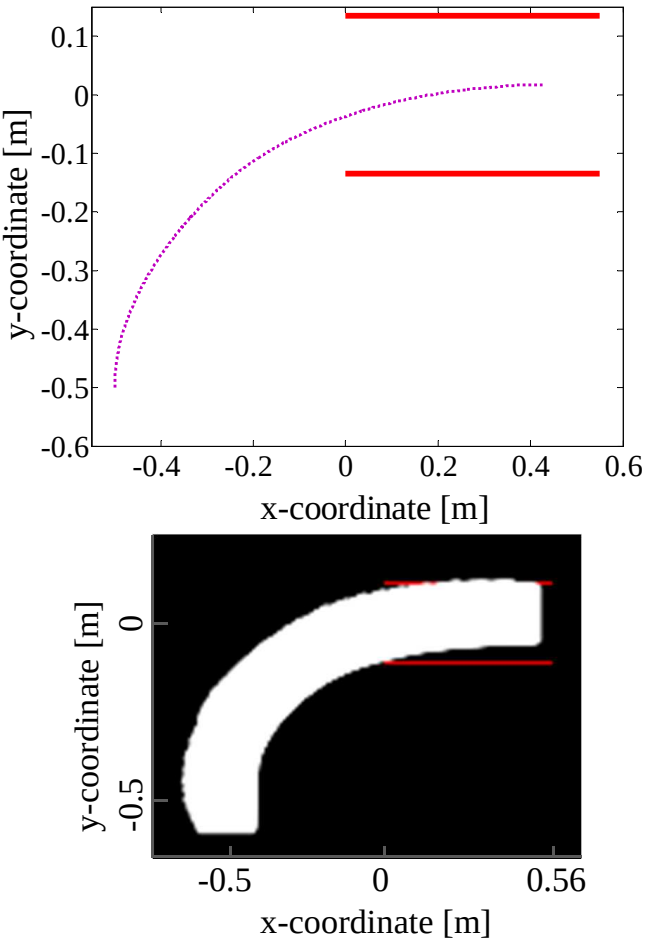


Fig. 4.19 Simulation result of perpendicular parking, case 2

small amount of the orientation error about 0.047 [rad] for our parking system. It can be clearly seen from all these results that our image-based fuzzy controller was able to park the vehicle to the appropriate parking position.

4.4.2 Real robot experiments

To verify the performance of the proposed algorithm, a real robot experiment was conducted in this study. Figure 4.23 shows the real robot of the experiment which was a front-wheel steered mobile robot. The robot was made by modifying a commercially available radio controlled car to control it by a microcontroller. A web camera is used to detect the parking frame and the captured image from the camera is sent to the Laptop PC. Each frame captured by the web camera at each sampling has 640×480 pixels and each pixel has three eight-bit-depth color channels (red, green, and blue). The Laptop PC executes the image processing to

Table 4.3 Fuzzy rules for ϕ

α_t \ β_t	**NB**	**NS**	**ZO**	**PS**	**PB**
NB	NB	NB	NS	NS	ZO
NS	NB	NS	NS	ZO	PS
ZO	NS	NS	ZO	PS	PS
PS	NS	ZO	PS	PS	PB
PB	ZO	PS	PS	PB	PB

Table 4.4 Specifications of the real robot experiments

Item	Property
CPU	2.4 [GHz]
Memory	4 [GB]
OS	Windows 7
Image size	640 $\times$ 480 [pixel]
Sampling rate	20 [fps]
Max. advanced speed	0.15 [m/s]
Range of steering angle	$\pm$ 0.349 [rad]
Wheelbase (l)	0.26 [m]

extract α_t and β_t using the PPHT from the captured image. Table 4.4 shows the specifications for the real robot experiments.

The robot has two kinds of motors: one is the DC motor used for controlling the speed, and the other is the DC servo motor used for controlling the steering angle of the robot. The web cameras that have 52 and 120 degree angle of views (AOVs) were used in the experimental studies. For the experimental studies which used the 52 degree AOV camera, the initial orientations of the vehicle were set at $\theta = -135$ [deg] and -45 [deg] due to the limitation of visibility. The results are shown in Fig. 4.24 and Fig. 4.25.

For the experimental studies used the 120 degree AOV camera, our image-based fuzzy parking controller was tested with two different initial postures of the robot, case 1 and case 2. The initial postures of the vehicle are located at (x, y, θ) = (0.55 [m], 0.55 [m], -110 [deg]) for case 1 and (-0.55 [m], 0.55 [m], -70 [deg]) for case 2. The experimental results

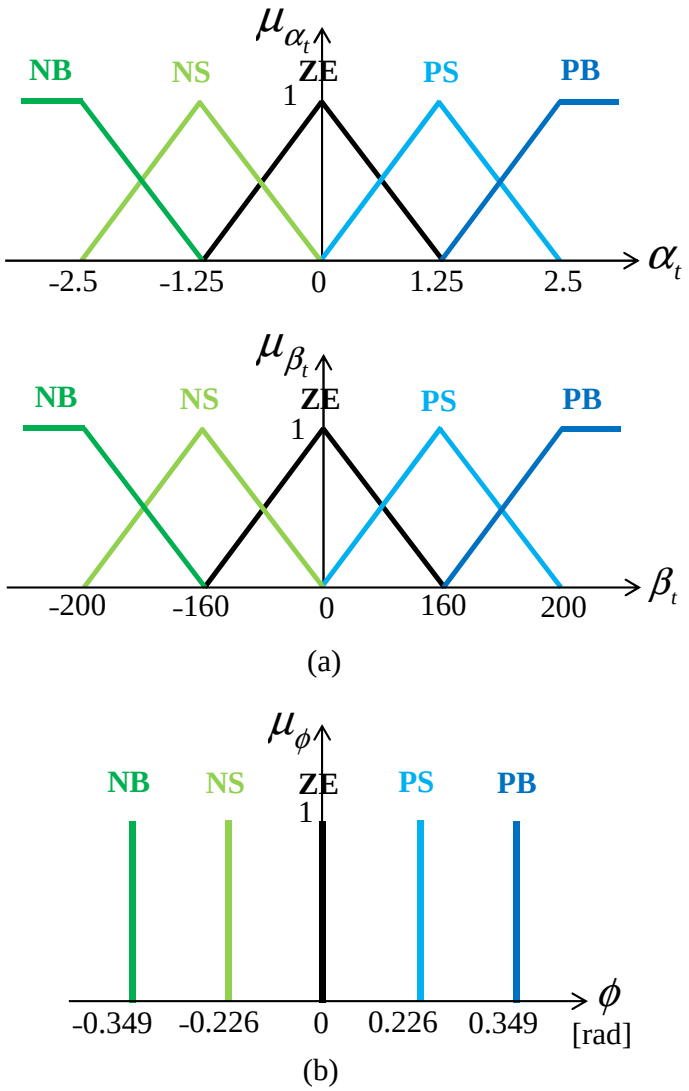


Fig. 4.20 Membership functions: (a) for the input variables α_t and β_t, and (b) for the output variable ϕ for the 25 fuzzy rules

for our automatic parking system are shown in Figs. 4.26 and 4.27. According to the results, there are only the small amounts of the orientation errors for both cases.

4.5 Summary

In this chapter, an image-based fuzzy parking control scheme which enables the car-like mobile robot to park towards the red color parking frame drawn on the floor, has been described. The image-based control method was able to reduce the amount of calculations

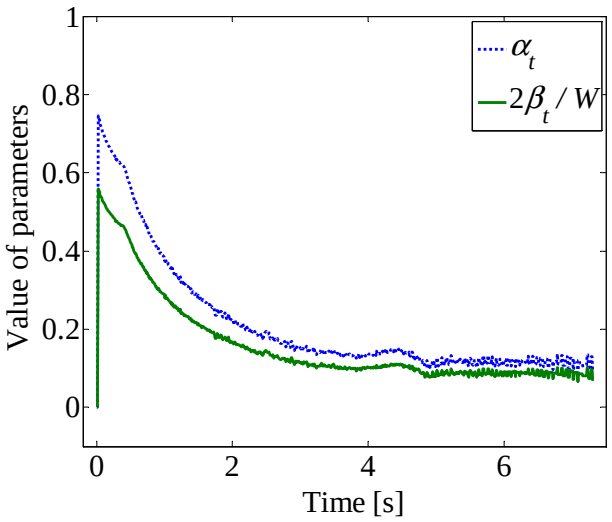


Fig. 4.21 History of the parameters related to perpendicular parking (case 1), when used the 25 fuzzy rules

because it did not need to perform the position estimation processes. Our controller was designed with two inputs, which were the slope and intercept of the target line, and one output that was the steering angle of the robot. The results of both simulation and real robot experiments showed the effectiveness of the proposed control scheme. The camera position and angle are quite effective on the target line of our parking system because it deeply depends on the image information of the captured image. Changing the camera position and angle is really effective on the results. To obtain the best results for both cases, it needs to consider two steps. The first step is to find the most suitable camera position and angle by performing the experiment over and over again. The second one is to optimize the width of the membership functions of our image-based fuzzy controller through the optimization techniques such as genetic algorithms. We have to use a genetic algorithm to optimize the width of the membership functions of our controller as future work.

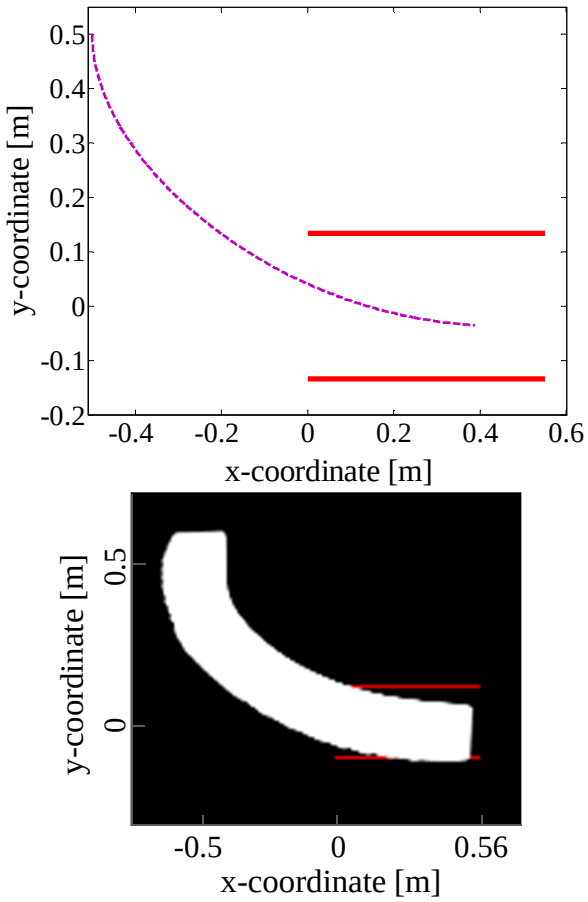


Fig. 4.22 Simulation result of perpendicular parking (case 1), when used the 25 fuzzy rules

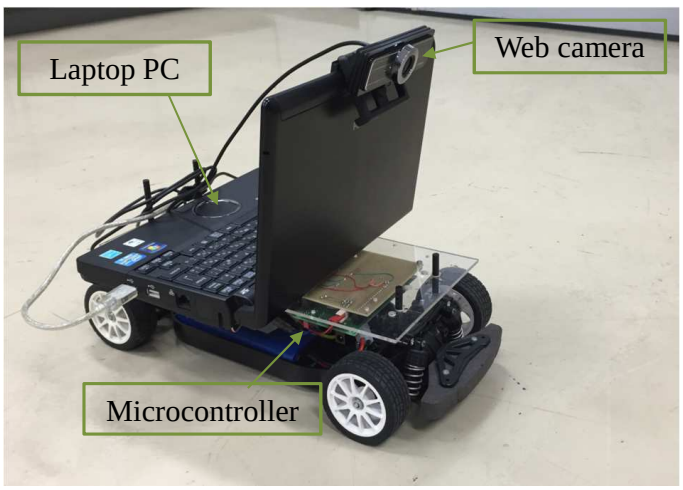


Fig. 4.23 Experimental robot

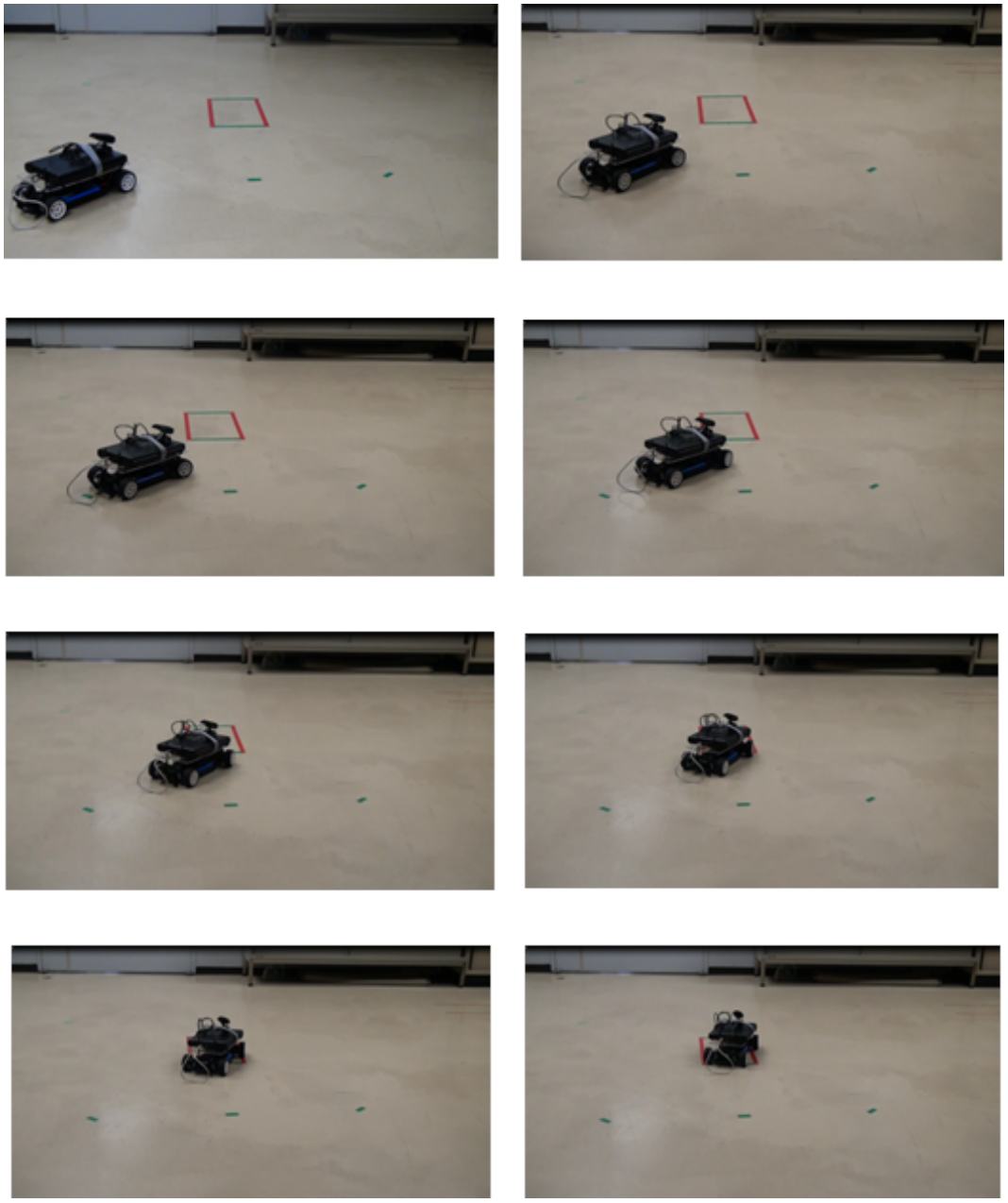


Fig. 4.24 Experimental results that used the 52 degree AOV camera, when started with initial conditions $(x, y, \theta) = (0.49\ [\mathrm{m}], 0.54\ [\mathrm{m}], -135\ [\mathrm{deg}])$

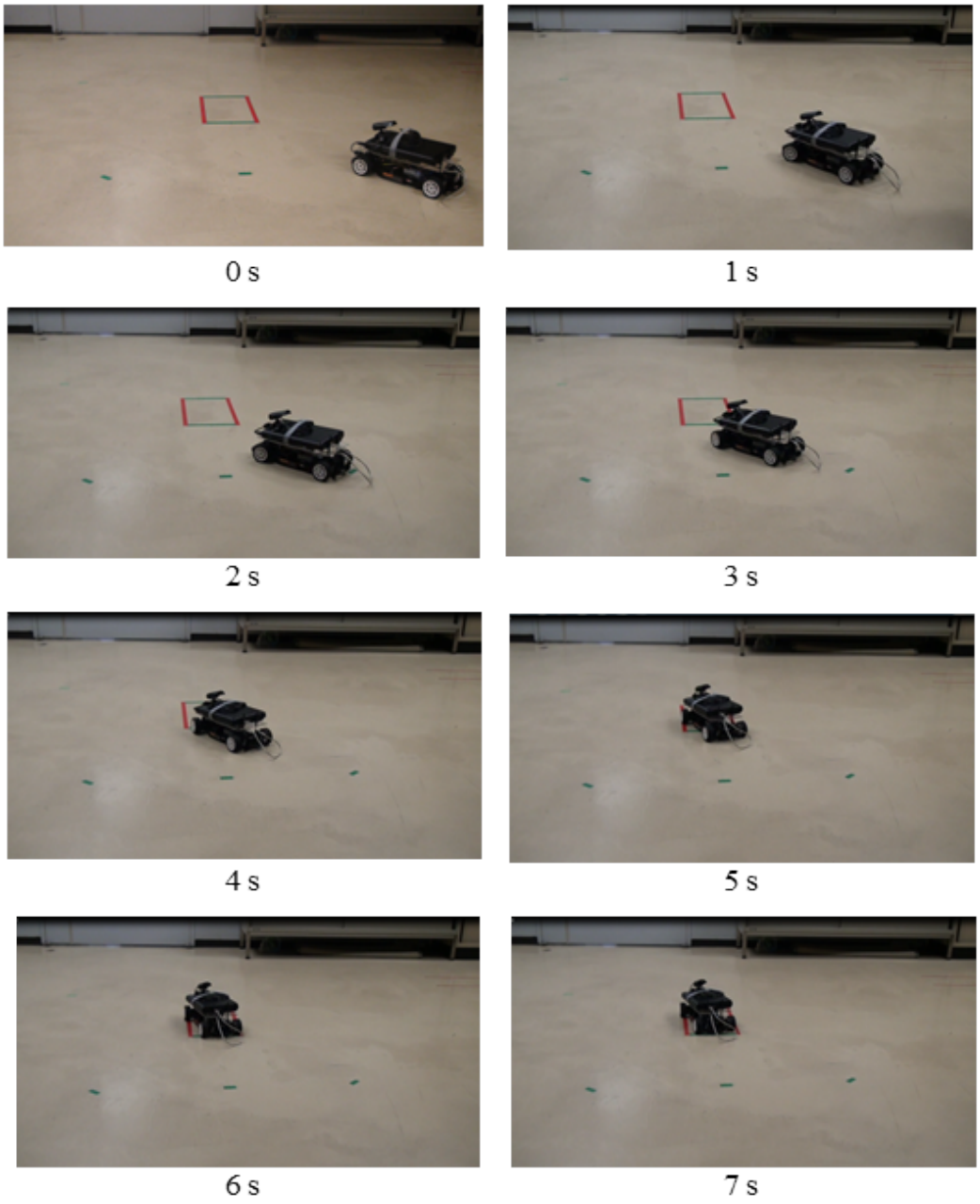


Fig. 4.25 Experimental results that used the 52 degree AOV camera, when started with initial conditions $(x, y, \theta) = (-0.49\ [\mathrm{m}], 0.54\ [\mathrm{m}], -45\ [\mathrm{deg}])$

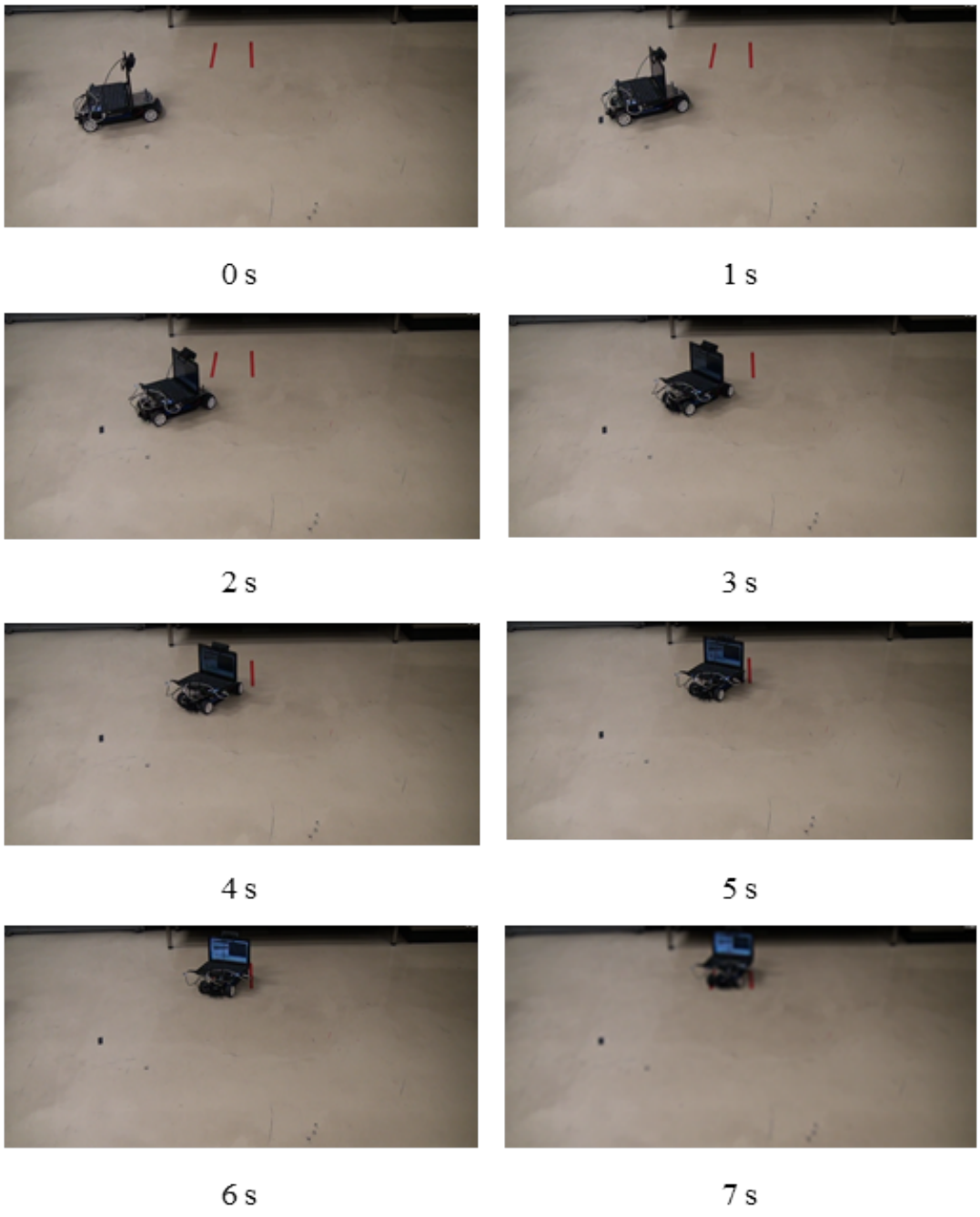


Fig. 4.26 Experimental results of case 1, where the 120 degree AOV camera was used

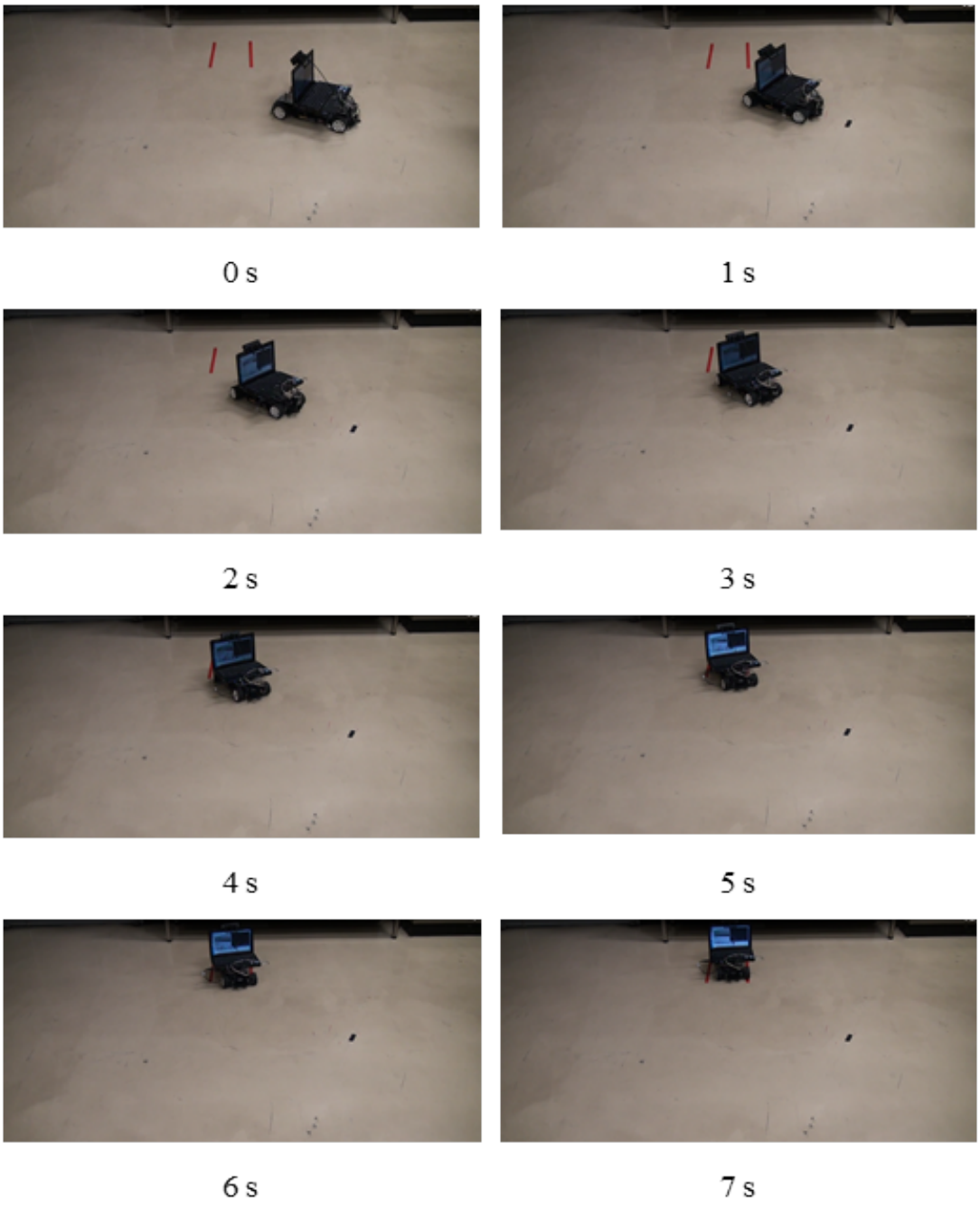


Fig. 4.27 Experimental results of case 2, where the 120 degree AOV camera was used

Chapter 5

Optimization of an Image-based Fuzzy Controller by a Genetic Algorithm

An automatic parking system of a car-like mobile robot is an important issue in commercial applications. An image-based fuzzy controller for an automatic parking system of a car-like mobile robot was presented in Chapter 4, where the membership functions were tuned by manually. However, we know usually that it is not easy to obtain suitable fuzzy membership functions and its rules of a fuzzy controller. For this reason, a genetic algorithm (GA) was used to optimize the width of the membership functions of our image-based fuzzy controller.

The goal of this chapter is to optimize the parameters of the membership functions, which were discussed in previous chapter, by using a GA against the complicated tuning of the controller. It is not used to tune the fuzzy reasoning rules because the rules are constructed based on the skills of the experienced human drivers in advance. The fuzzy rules are very standard rules and there is not much room to modify them. Adjusting these rules leads to inefficient tuning efforts.

The rest of this chapter is organized as follows. Section 5.1 describes the problem setting and Section 5.2 presents the image-based fuzzy controller to follow the desired trajectory of our parking system. A method for optimizing the fuzzy controller by GA is developed in Section 5.3. In Section 5.4, some simulation results are given to demonstrate the effectiveness of the proposed method. Section 5.5 summarizes the chapter.

5.1 Problem Setting

An image-based fuzzy parking controller of a car-like mobile robot was developed in our previous chapter. In this chapter, the parameters of the membership functions were optimized

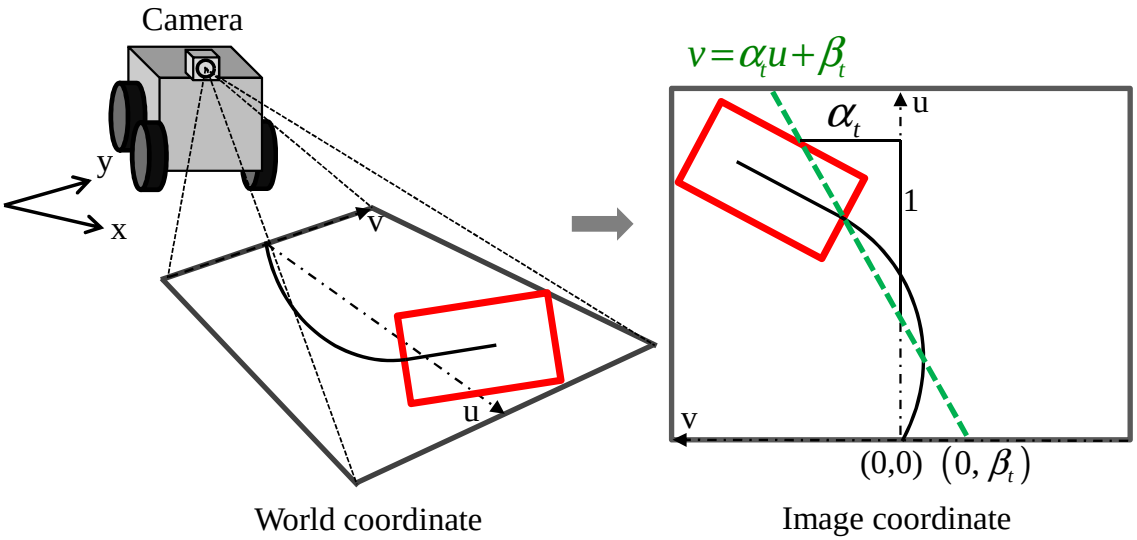


Fig. 5.1 Overview of the environment and the definition of the coordinate

by using a GA. Figure 5.1 shows the experimental overview and the definition of the coordinates. As shown in this figure, let the world coordinate be represented by x-y coordinate and the image coordinate for the camera be described by u-v coordinate. The image size is W (width) $\times$ H (height), in the pixels. The controlled object, a car-like mobile robot is shown in Fig. 5.2. The robot state is denoted by the position and the orientation of the vehicle with respect to the x-axis such as (x, y, θ) and the control input is composed of the forward speed and the steering angle, (s, ϕ). In addition, the distance between the front and rear wheels is denoted by l and the camera is directed to the forward direction of running.

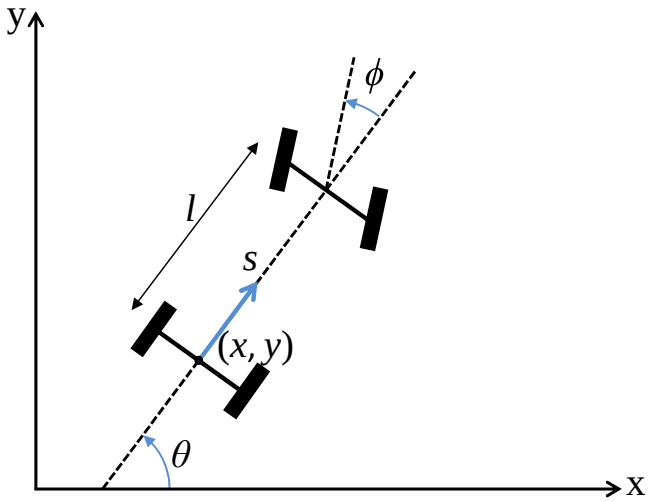


Fig. 5.2 Controlled object

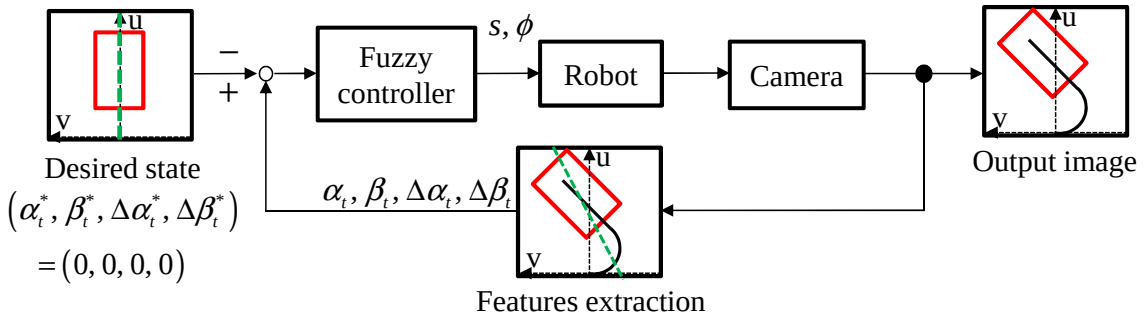


Fig. 5.3 Block diagram of an image-based fuzzy parking control method

According to our driving experiences, we must turn the steering wheel, slightly to the left, to enter the car in the garage form of a quarter circle and straight. Depending on this idea, the desired target line for perpendicular parking was designed to include a circular motion and a linear motion. The desired target line is linearized to a linear function of $v = \alpha_t u + \beta_t$ by a least squares method. α_t is the slope and β_t is the intercept of the desired target line, which are calculated by [93]:

$$\begin{aligned}
\alpha_t &= \frac{n\sum_{i=1}^{n} u_i v_i - \sum_{i=1}^{n} u_i \sum_{i=1}^{n} v_i}{n\sum_{i=1}^{n} u_i^2 - \left(\sum_{i=1}^{n} u_i\right)^2} \\
\beta_t &= \frac{\sum_{i=1}^{n} u_i^2 \sum_{i=1}^{n} v_i - \sum_{i=1}^{n} u_i v_i \sum_{i=1}^{n} u_i}{n\sum_{i=1}^{n} u_i^2 - \left(\sum_{i=1}^{n} u_i\right)^2}
\end{aligned} \tag{5.1}$$

where (u_i, v_i) represents the coordinate of i-th pixel that includes the target line on the captured image, and n is the total number of the pixels. In addition, the variable parameters, $\Delta\alpha_t$ and $\Delta\beta_t$ are the difference between the parameter values extracted from the current frame α_t, β_t and the previous frame α_{t-1}, β_{t-1}. The $\Delta\alpha_t$ and $\Delta\beta_t$ are calculated by using the following equations:

$$\begin{aligned}
\Delta\alpha_t &= \alpha_t - \alpha_{t-1} \\
\Delta\beta_t &= \beta_t - \beta_{t-1}.
\end{aligned} \tag{5.2}$$

Thus, the controller for tracking the target line for our parking system determines the control inputs (s, ϕ) from the information $(\alpha_t, \beta_t, \Delta\alpha_t, \Delta\beta_t)$ on the image coordinate. Figure 5.3 shows the block diagram of this control. This block diagram shows that the robot is controlled without referring to its position.

5.2 Image-based Fuzzy Parking Controller

Fuzzy logic control is one of the most successful areas in the application of intelligent control. The application of fuzzy logic does not need much detailed knowledge of the system and utilizes the human (expert) knowledge and intuition instead of the mathematical knowledge of the system. It includes three components which are fuzzification, fuzzy inference and defuzzification.

In this work, since the relationship between the parameters, (α_t, β_t, $\Delta\alpha_t$, $\Delta\beta_t$) on the image coordinate and the control inputs, (s, ϕ), depends on the depth of each pixel, it may not be easy to derive the model relation accurately. Therefore, it was decided to use a model-free fuzzy controller. The fuzzified state variables are denoted by (α_t, β_t, $\Delta\alpha_t$, $\Delta\beta_t$) and their elements are defined by

$$\begin{aligned} \alpha_t &= \{\mathrm{N}, \mathrm{ZO}, \mathrm{P}\} \\ \beta_t &= \{\mathrm{N}, \mathrm{ZO}, \mathrm{P}\} \\ \Delta\alpha_t &= \{\mathrm{N}, \mathrm{ZO}, \mathrm{P}\} \\ \Delta\beta_t &= \{\mathrm{N}, \mathrm{ZO}, \mathrm{P}\} \end{aligned} \tag{5.3}$$

where N, ZO and P are the labels that mean Negative, Zero and Positive, respectively.

Figure 5.4 shows the membership functions of each state variable. In the figure, a_{α_t}, a_{β_t}, $a_{\Delta\alpha_t}$, $a_{\Delta\beta_t}$, a_{ss}, a_{sb}, $a_{\phi s}$ and $a_{\phi b}$ are the variables which represent the shape of the membership functions of each state variable. It should be noted that the membership functions for β_t and $\Delta\beta_t$ refer to the values of $\frac{2\beta_t}{W}$ and $\frac{2\Delta\beta_t}{W}$ which are normalized by the width W of the respective images. The range of each membership function is [0, 1]. The μ_{α_t}, μ_{β_t}, $\mu_{\Delta\alpha_t}$, and $\mu_{\Delta\beta_t}$ represent the membership functions to calculate the grades of states, α_t, β_t, $\Delta\alpha_t$, and $\Delta\beta_t$. The membership functions to calculate the grade of the control inputs, s and ϕ, are represented as, μ_s and μ_ϕ. Figure 5.5 illustrates the procedure to determine the control inputs, s and ϕ.

Control rules are considered as 3^4 types from the four state variables that own three labels respectively, which are defined as IF-THEN rules as follows:

R_1 : IF α_t = N, β_t = N, $\Delta\alpha_t$ = N, $\Delta\beta_t$ = N THEN $s = s_1$, $\phi = \phi_1$

R_2 : IF α_t = N, β_t = N, $\Delta\alpha_t$ = N, $\Delta\beta_t$ = ZO THEN $s = s_2$, $\phi = \phi_2$

$\vdots$

R_{80} : IF α_t = P, β_t = P, $\Delta\alpha_t$ = P, $\Delta\beta_t$ = ZO THEN $s = s_{80}$, $\phi = \phi_{80}$

R_{81} : IF α_t = P, β_t = P, $\Delta\alpha_t$ = P, $\Delta\beta_t$ = P THEN $s = s_{81}$, $\phi = \phi_{81}$

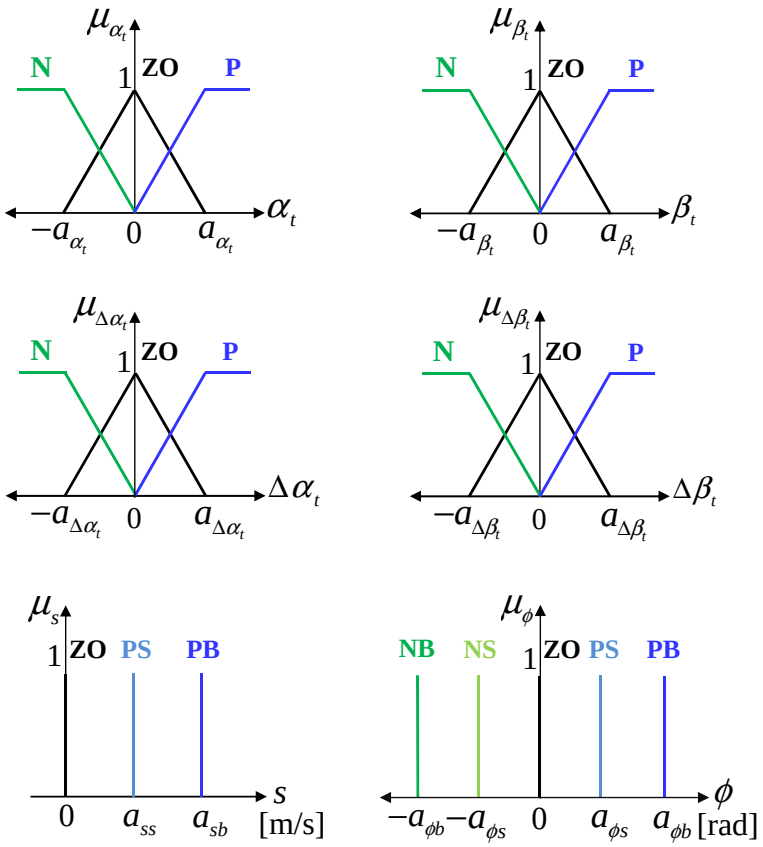


Fig. 5.4 Membership functions

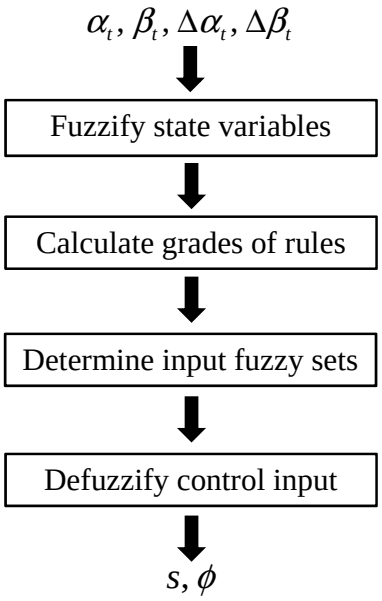


Fig. 5.5 Process to determine the control inputs, s and ϕ

The rules for the control inputs, s_i and ϕ_i (i = 1, ..., 81) are defined as:

$$s = \{\text{ZO}, \text{PS}, \text{PB}\}, \quad \phi = \{\text{NB}, \text{NS}, \text{ZO}, \text{PS}, \text{PB}\} \tag{5.4}$$

where fuzzy sets NB, NS, ZO, PS, and PB represent Negative Big, Negative Small, Zero, Position Small, and Positive Big, respectively. The reason that does not include the labels NB and NS in s, is because of $s > 0$. The adaptability of the control rules h_{R_i} (i = 1, . . . , 81) are obtained from the following equation as a product of the membership function values corresponding to the grades of each state variable:

$$\begin{aligned}
h_{R_1} &= \mu_{\alpha_t,\text{N}}(\alpha_t).\mu_{\beta_t,\text{N}}(\beta_t).\mu_{\Delta\alpha_t,\text{N}}(\Delta\alpha_t).\mu_{\Delta\beta_t,\text{N}}(\Delta\beta_t) \\
h_{R_2} &= \mu_{\alpha_t,\text{N}}(\alpha_t).\mu_{\beta_t,\text{N}}(\beta_t).\mu_{\Delta\alpha_t,\text{N}}(\Delta\alpha_t).\mu_{\Delta\beta_t,\text{ZO}}(\Delta\beta_t) \\
&\vdots \\
h_{R_{80}} &= \mu_{\alpha_t,\text{P}}(\alpha_t).\mu_{\beta_t,\text{P}}(\beta_t).\mu_{\Delta\alpha_t,\text{P}}(\Delta\alpha_t).\mu_{\Delta\beta_t,\text{ZO}}(\Delta\beta_t) \\
h_{R_{81}} &= \mu_{\alpha_t,\text{P}}(\alpha_t).\mu_{\beta_t,\text{P}}(\beta_t).\mu_{\Delta\alpha_t,\text{P}}(\Delta\alpha_t).\mu_{\Delta\beta_t,\text{P}}(\Delta\beta_t)
\end{aligned} \tag{5.5}$$

Here, $\mu_{\alpha_t,\text{N}}(\alpha_t)$ represents the membership function to calculate the grade of state α_t when α_t = N. It should be noted that the graph of the membership functions for α_t in Fig. 5.4 includes three functions, such as $\mu_{\alpha_t,\text{N}}(\alpha_t)$, $\mu_{\alpha_t,\text{ZO}}(\alpha_t)$ and $\mu_{\alpha_t,\text{P}}(\alpha_t)$, and other graphs include some membership functions in a similar way.

The control inputs, s and ϕ are determined by the following formula as a weighted average that is based on the goodness of fit of the control rules:

$$s = \frac{\sum_{i=1}^{81} h_{Ri}\hat{s}_i}{\sum_{i=1}^{81} h_{Ri}}, \quad \phi = \frac{\sum_{i=1}^{81} h_{Ri}\hat{\phi}_i}{\sum_{i=1}^{81} h_{Ri}} \tag{5.6}$$

where $\hat{s}_i$ and $\hat{\phi}_i$ are the real values corresponding to the grades of the consequent s_i and ϕ_i.

5.3 Optimization of Membership Functions by GA

The performance of the fuzzy logic controller is influenced by its knowledge base (rule set) and the membership functions. It is extremely important to adjust the rules of fuzzy controller and the membership functions of input and output variables to obtain a better performance. Manual tuning of fuzzy logic controllers is a very complex process, which

consists of choosing the type of fuzzy logic controller, the number and shape of membership functions of the inputs and outputs, and the rule base. In most fuzzy systems, membership functions, fuzzy reasoning rules and scaling factors are determined through trial and error by human operators, and it takes many iterations to converge to desirable parameters. The membership functions of each state variable discussed in Section 5.2 were created all by human experience, which will influence the performance of our image-based fuzzy controller. Therefore, it is indispensable to adjust them to obtain a better performance. The main goal of this paper is to show that tuning of the shape of membership functions of input and output variables, can be automated by using a GA to optimize the proposed fuzzy controller. In our research, GA has introduced to reduce the required tuning efforts of the 81 fuzzy control rules by human operator.

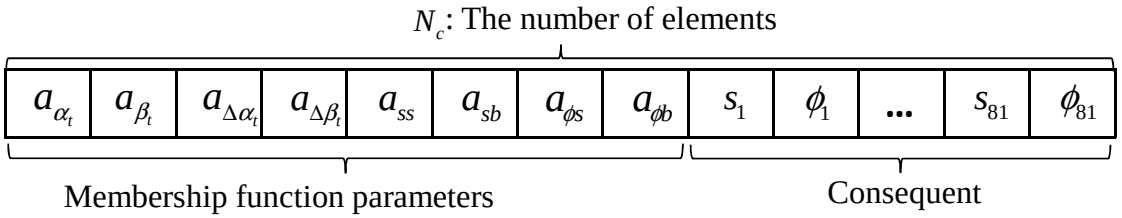
GAs are theoretically proven to provide a robust search in complex spaces, offer a valid approach to the problems which are requiring the efficient and effective search. GAs provide an alternative design approach to obtain the effective fuzzy control systems. It is expected to save the time and cost of the iterative tuning of a human designer. It is generally accepted that any GA for solving a problem must take into account the following steps [94]:
Step 1: Determine the number of chromosomes, generations, and mutation rate and crossover rate values
Step 2: Construct the random initial population of chromosomes
Step 3: Process steps 4-6 until the number of generations is met
Step 4: Evaluate the fitness value of chromosomes by calculating an objective function
Step 5: Breed new individuals through chromosomes selection, crossover and mutation operators
Step 6: Generate new Chromosomes (Offspring)
Step 7: Obtain solution (Best Chromosomes).

Therefore, the proposed optimization procedure can be achieved in accordance with the following order.

5.3.1 Chromosome and initialization

A chromosome is a solution which is generated by a GA. A chromosome is composed from genes and its value can be either numerical, binary, symbols or characters, depending on the problem what we want to be solved. The chromosome corresponds to a possible solution of the optimization problem and every chromosome is composed of several encoded genes.

In order to encode the image-based fuzzy controller, the proposed encoding procedure for the membership functions is integrated. Figure 5.6 shows the chromosome that encoded membership function parameters and the consequent. In our case, the number of individuals



$a_{\alpha_t}, a_{\beta_t}, a_{\Delta\alpha_t}, a_{\Delta\beta_t} = 0.0 \sim 2.0$

$a_{ss}, a_{sb} = 0.0 \sim 0.2$

$a_{\phi s}, a_{\phi b} = 0.0 \sim 0.436$

$s_i = \{0(\text{ZO}), 1(\text{PS}), 2(\text{PB})\}$

$\phi_i = \{-2(\text{NB}), -1(\text{NS}), 0(\text{ZO}), 1(\text{PS}), 2(\text{PB})\}$

$i = 1, 2, \ldots, 81$

Fig. 5.6 Chromosome-encoded membership function parameters and consequent

Table 5.1 Important parameters of GA

Parameters	Value
Number of generations	200
Number of individuals	80
Number of elites	10
Crossover rate	0.9
Mutation rate	0.006

of the generation is initialized as 80. The important parameters of the GA are shown in Table 5.1.

5.3.2 Evaluation function

The evaluation function used in our research is as follows:

$$V = \frac{W_{11}}{\alpha_e + \varepsilon} + \frac{W_{12}}{\beta_e + \varepsilon} + \frac{W_{22}}{t_f} + W_{31}x_f + W_{32}y_f + W_{33}\theta_f \tag{5.7}$$

where W_{11}, W_{12}, W_{22}, W_{31}, W_{32} and W_{33} are the weighting factors; α_e and β_e are the RMS errors of α_t and β_t; t_f is end time; and x_f, y_f and θ_f are the advanced distance of the states of the robot.

The constants used in the evaluation function of Eq. (5.7) are $W_{11} = 1$, $W_{12} = 1$, $W_{22} = 50$, $W_{31} = 50$, $W_{32} = 25$, $W_{33} = 200$ and $\varepsilon = 0.025$. To avoid the division by zero problem, the small constant value, ε is added in the evaluation function.

5.3.3 GA operators

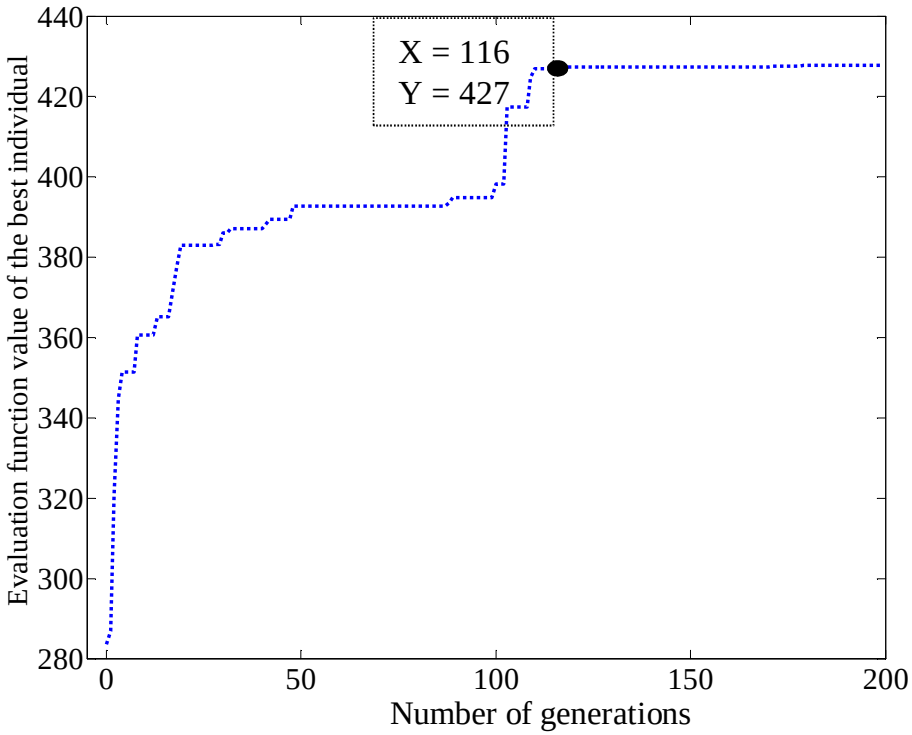


Fig. 5.7 Score of the best individual

In the proposed algorithm, three traditional operators: selection, crossover and mutation were used. Here, roulette wheel selection and uniform crossover, are used for selection operator and crossover in this work. In the mutation process, a random value replaces within the possible range that can be taken by a mutation probability of $1/N_c$, where N_c is the number of elements in the chromosome. The optimization procedure is implemented by following the above steps.

Figure 5.7 shows the transition of the evaluation value of the best individual in each generation of the GA. From this figure, it can be seen that the curve shows the maximum evaluation value of the best individual at the 116th generation. Table 5.2 shows the membership functions which are obtained by GA optimization.

5.4 Simulation Experiment

In this section, simulation results are given to demonstrate the ability of the proposed method. The desired target trajectory for our parking system was assumed to be a quarter circle and a

Table 5.2 Optimized membership function parameters

Parameters	Value
a_{α_t}	1.389
a_{β_t}	1.512
$a_{\Delta\alpha_t}$	0.62
$a_{\Delta\beta_t}$	1.153
a_{ss}	0.042
a_{sb}	0.187
$a_{\phi s}$	0.185
$a_{\phi b}$	0.429

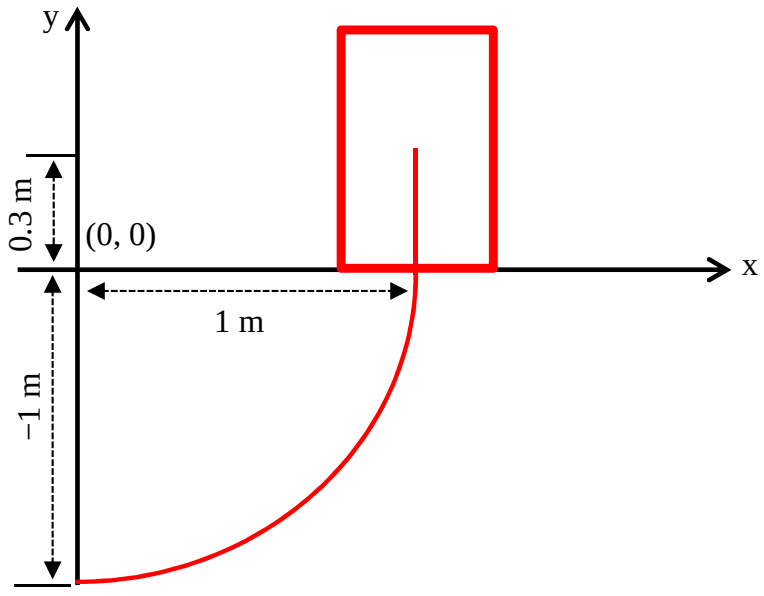


Fig. 5.8 The desired target trajectory for perpendicular parking system

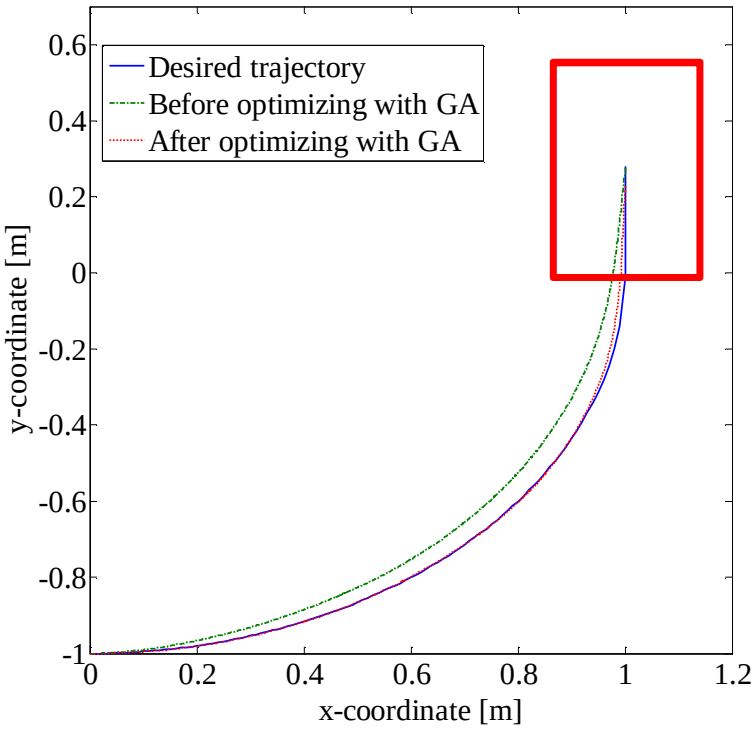


Fig. 5.9 Optimized trajectory path, when started with initial conditions $(x, y, \theta) = (0, -1, 0^\circ)$

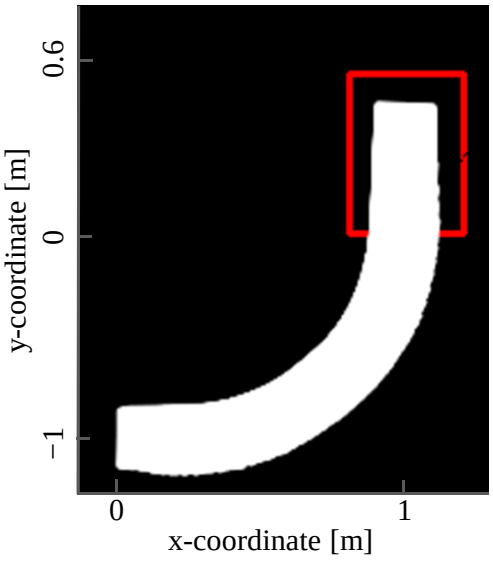


Fig. 5.10 Simulation result after optimization process

line at the end of the quarter circle to include the circular motion and linear motion during the parking mission. The desired target trajectory is represented as a function $y = f(x)$. The general form for circular motion is given by

$$y = -\sqrt{(1-x^2)}, \qquad 0 \leq x \leq 1. \tag{5.8a}$$

And the linear motion becomes

$$x = 1, \qquad 0 \leq y \leq 0.3. \tag{5.8b}$$

The desired target trajectory of perpendicular parking system is shown in Fig. 5.8. The controlled object was a car-like mobile robot and its model is expressed by:

$$\begin{aligned} \dot{x} &= s\cos\theta \\ \dot{y} &= s\sin\theta \\ \dot{\theta} &= \frac{s\tan\phi}{l} \end{aligned} \tag{5.9}$$

The camera was mounted on the robot with its height of 0.2 [m] and the direction of 60 [deg] to the front from the vertical downward. The camera horizontal and vertical view angles were 26 [deg] and 20 [deg], respectively. The size of captured image is specified as 320 [pixel] $\times$ 240 [pixel]. The initial condition of the robot was assumed to be (x, y, θ) = (0 [m], -1.0[m], 0 [rad]). Figure 5.9 shows the effectiveness of the optimization. The simulation result, which uses the optimized membership functions, is shown in Fig. 5.10. According to the result, there is only the small amount of the orientation error about 0.034 [rad]. It has sufficient fuzzy rules to achieve our control objectives, though only three membership functions of fuzzy input for each state variable are used in our research.

5.5 Summary

In this chapter, an image-based fuzzy controller has been proposed to trace the desired target trajectory for a perpendicular parking system by adding $\Delta\alpha_t$ and $\Delta\beta_t$, which were the difference between the current and previous extracted values on the image coordinate, as state variables. The shape tuning of the membership functions in the fuzzy controller became complicated with the increase of state variables, so that we proposed a method to optimize our controller by a GA. The GA provided a systematic approach to determine the shape of membership functions for the fuzzy controller. Moreover, a modification of the

image coordinate, i.e., changing of the v-axis, was able to reduce the y-direction errors. For example, even if β_t was zero on the old image coordinate in Chapter 4, β_t was positive on the new modified coordinate. Therefore, our controller generated the control inputs to approach the desired target trajectory more closely. The results confirmed that the proposed method was able to trace the desired target trajectory for our parking system.

Chapter 6

Conclusion and Future Work

6.1 Concluding Remarks

In this thesis, the switching and non-switching controllers based on an invariant manifold theory, an image-based fuzzy controller, and an optimized image-based fuzzy controller by a GA, have been described to solve the parking problem of a car-like mobile robot.

For stabilizing the car-like mobile robot in the desired posture, Chapter 3 has presented the switching and non-switching controllers based on an invariant manifold theory. It was found that our invariant manifold approach consists an incomplete manifold under a linear state feedback controller, which makes all the states tend to the desired position except the vehicle orientation. For this reason, the linear state feedback controller was enhanced by an additional state feedback to make the constructed manifold with the method of Tayebi *et al*. [7]. It has been proved from the results that the switching controllers converged the vehicle to the desired states when the initial states were $\theta = 1$ radian and $\phi \neq 0$. On the other hand, the non-switching controller was able to make the vehicle to be stable and converged to the desired states for both parking types.

In the above approach, it was assumed that the robot states can be exactly obtained for the feedback control purpose. This assumption is usually not satisfied in a real parking system due to the uncertainties in the kinematic model and slippage of the wheels on the ground. For this reason, an image-based fuzzy controller for an automatic parking system of a car-like mobile robot has been presented in Chapter 4. In this approach, the image information which was acquired from a camera image was used to control a robot without using the robot position. A model-free fuzzy controller was designed to combine the behavior of the robot with the image features without estimating the image depth. The results proved that our image-based fuzzy controller was able to park the vehicle to the appropriate parking position.

It is well known that the performance of the fuzzy logic controller is influenced by its knowledge base and the membership functions. To obtain the best performance of fuzzy logic controller, it is very important to optimize the rules and the widths of the membership functions of input and output variables of fuzzy logic controller. However, the process of manually tuning the fuzzy logic controller is a very complex task. Chapter 5 has presented the process of optimizing the widths of membership functions of our image-based fuzzy parking controller by using a GA against the complicated tuning of the controller. It was proved that the tuning of the shape of membership functions of input and output variables can be automated by using the GA to optimize the proposed image-based fuzzy controller. The simulation results were given to confirm that our optimized image-based fuzzy controller was able to trace the desired target trajectory of the perpendicular parking system.

6.2 Future Work

We have to test the GA optimized image-based fuzzy controller on some real robot experiments for parking problem as a future work.

In this dissertation, our image-based fuzzy controller was tested only for forward parking system and the backward parking system is left for future study. In the backward parking system, the first task is to find a suitable path from the initial parking position to the switching point, from where it is ensured that the vehicle is able to leave the parking spot.

The image-based fuzzy parking control system described in this thesis did not consider the avoidance of unforeseen obstacles, but focused on the parking the vehicle in a constrained parking lot. The parking system, which considers for obstacle environments should be investigated in the future study.

References

[1] I. Kolmanovsky and N. H. McClamroch, “Developments in nonholonomic control problems,” *IEEE Control Systems Magazine*, vol. 15, no. 6, pp. 20–36, 1995.

[2] R. W. Brockett, “Asympototic stability and feedback stabilization,” *Differential Geometric Control Theory*, vol. 27, no. 1, pp. 181–191, 1983.

[3] J. B. Pomet, “Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift,” *Systems and Control Letter*, vol.18, no. 2, pp. 147–158, 1992.

[4] C. Samson, “Control of chained systems application to path following and time-varying point-stabilization of mobile robots,” *IEEE Transactions on Automatic Control*, vol. 40, no. 1, pp. 64–77, 1995.

[5] A. Astolfi, “Discontinuous control of nonholonomic systems,” *System and Control Letters*, vol. 27, no. 1, pp. 37–45, 1996.

[6] A. Tayebi, M. Tadjine, and A. Rachid, "Discontinuous control design for the stabilization of nonholonomic systems in chained form using the backstepping approach: Application to mobile robots,” in *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997, pp. 3089–3090.

[7] A. Tayebi, M. Tadjine, and A. Rachid, “Invariant manifold approach for the stablization of nonholonomic chained systems: Application to a mobile robot,” *Nonlinear Dynamics*, vol. 24, no. 2, pp. 167–181, 2001.

[8] H. Khennouf and C. C. De Wit, “Quasicantinuous exponential stabilizers for nonholonomic systems,” in *Proceedings of IFAC 13th World Congress*, 1996, pp. 49–54.

[9] H. Khennouf and C. Canudas de Wit, “On the construction of stabilizing discontinuous controllers for nonholonomic system,” in *Proceedings of IFAC Nonlinear Control Systems Design Symp. (NOLCOS’ 95)*, 1995, pp. 667–672.

[10] K. Watanabe, T. Yamamoto, K. Izumi, and S. Maeyama, "Underactuated control for nonholonomic mobile robots by using double integrator model and invariant manifold theory," in *Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 2862–2867.

[11] T. Yamamoto and K. Watanabe, "A switching control method for stabilizing a nonholonomic mobile robot using invariant manifold method," in *Proceedings of SICE Annual Conference*, 2010, pp. 3278–3284.

[12] K. Watanabe, Y. Ueda, and I. Nagai, "Underactuated control for a fire truck-type mobile robot using an invariant manifold theory," in *Proceedings of the 6th International Conference on Soft Computing and Intelligent Systems and the 13th International Symposium on Advances Intelligent Systems (SCIS-ISIS)*, 2012, pp. 210–215.

[13] K. Watanabe, Y. Ueda, I. Nagai, and S. Maeyama, "Stabilization of a fire truck robot by an invariant manifold theory," *Procedia Engineering*, vol. 41, pp. 1095–1104, 2012.

[14] K. Izumi and K. Watanabe, "Switching manifold control for an extended nonholonomic double integrator," in *Proceedings of International Conference on Control Automation and Systems (ICCAS)*, 2010, pp. 27–30.

[15] M. Reyhanoglu, "On the stabilization of a class of nonholonomic systems using invariant manifold technique," in *Proceedings of the 34th IEEE Conference On Decision and Control*, 1995, pp. 2125–2126.

[16] S. Lee, M. Kim, Y. Youm, and W. Chung, "Control of a car-like mobile robot for parking problem," in *Proceedings of IEEE International Conference Robotics and Automation*, 1990, pp. 1–6.

[17] N. Marchand and M. Alamir, "Discontinuous exponential stabilization of chained form systems," *Automatica*, vol. 39, no. 2, pp. 343–348, 2003.

[18] A. M. Bloch, M. Reyhanoglu, and N. H. McClamroch, "Control and stabilization of nonholonomic dynamic systems," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1746–1757, 1992.

[19] Y. P. Tian and S. Li, "Exponential stabilization of nonholonomic dynamic systems by smooth time-varying control," *Automatica*, vol. 38, no. 7, pp. 1139–1146, 2002.

[20] Y. Hu, S. S. Ge, and C. Y. Su, "Stabilization of uncertain nonholonomic systems via time-varying sliding mode control," *IEEE Transactions on Automatic Control*, vol. 49, no. 5, pp. 757–763, 2004.

[21] E. Malis, "Survey of vision-based control," *ENSIETA European Naval Ship Design Short Course, Brest, France*, 2002.

[22] D. Tsakiris, C. Samson, and P. Rives, "Vision-based time-varying stabilization of a mobile manipulator," in *Proceedings of 6th International Conference on Control, Automation, Robotics and Vision (ICARV)*, 1996.

[23] I. Horswill, "Polly: A vision-based artificial agent," in *Proceedings of the 11th National Conference on Artificial Intelligence (AAAI-93)*, 1993, pp. 824–829.

[24] A. K. Das, R. Fierro, V. Kumar, B. Southall, J. Spletzer, and C. J. Taylor, "Real-time vision-based control of a nonholonomic mobile robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2001, pp.1714–1719.

[25] R. Carelli, C. M. Soria, and B. Morales, "Vision-based tracking control for mobile robots," in *Proceedings of 12th International Conference on Advanced Robotics (ICAR)*, 2005, pp. 148–152.

[26] N. R. Gans and S. A. Hutchinson, "A stable vision-based control scheme for nonholonomic vehicles to keep a landmark in the field of view," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2007, pp. 2196–2201.

[27] A. Cherubini, F. Chaumette, and G. Oriolo, "A position-based visual servoing scheme for following paths with nonholonomic robots," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 1648–1654.

[28] Y. Masutani, M. Mikawa, N. Maru, and F. Miyazaki, "Visual servoing for nonholonomic mobile robots," in *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, 1994, pp. 1133–1140.

[29] R. Pissard-Gibollet and P. Rives, "Applying visual servoing techniques to control a mobile hand-eye system," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1995, pp. 166–171.

[30] K. Hashimoto and T. Noritsugu, "Visual servoing of nonholonomic cart," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, 1997, pp. 1719–1724.

[31] N. R. Gans and S. A. Hutchinson, "A stable vision-based control scheme for nonholonomic vehicles to keep a landmark in the field of view," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2007, pp. 2196–2201.

[32] B. Thuilot, P. Martinet, L. Cordesses, and J. Gallice, "Position based visual servoing: Keeping the object in the field of vision," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, 2002, pp. 1624–1629.

[33] I. J. Ha, D. H. Park, and J. H. Kwon, "A novel position-based visual servoing approach for robust global stability with feature points kept within the field-of-view," in *Proceedings of 11th International Conference on Control, Automation, Robotics and Vision*, 2010, pp. 1458–1465.

[34] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *Proceedings of the 3rd International Symposium on Experimental Robotics III*, 1994, pp. 182–192.

[35] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.

[36] F. Conticelli, B. Allotta, and P. K. Khosla, "Image-based visual servoing of nonholonomic mobile robots," in *Proceedings of the 38th IEEE Conference on Decision Control*, 1999, pp. 3496–3501.

[37] H. Y. Wang, S. Itani, T. Fukao, and N. Adachi, "Image-based visual adaptive tracking control of nonholonomic mobile robots," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 1–6.

[38] G. L. Mariottini, G. Oriolo, and D. Prattichizzo, "Image-based visual servoing for nonholonomic mobile robots using epipolar geometry," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 87–100, 2007.

[39] G. L. Mariottini, D. Prattichizzo, and G. Oriolo, "Image-based visual servoing for nonholonomic mobile robots with central catadioptric camera," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2006, pp. 538–544.

[40] R. Rao, V. Kumar, and C. Taylor, "Visual servoing of a UGV from a UAV using differential flatness," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 743–748.

[41] Y. Ma, J. Kosecka, and S. S. Sastry, "Vision-guided navigation for a nonholonomic mobile robot," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 521–536, 1999.

[42] T. Kato, K. Watanabe, and S. Maeyama, "Image-based fuzzy trajectory tracking control for four-wheel steered mobile robots," *Artificial Life and Robotics*, vol. 17, no. 1, pp. 130–135, 2012.

[43] K. Watanabe, T. Kato, and S. Maeyama, "Obstacle avoidance for mobile robots using an image-based fuzzy controller," in *Proceedings of IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society*, 2013, pp. 6392–6397.

[44] G. M. Chen, P. Z. Lin, W. Y. Wang, T. T. Lee, and C. H. Wang, "Image-based fuzzy control system," *Electronics Letters*, vol. 44, no. 7, pp. 461–462, 2008.

[45] E. Malis, F. Chaumette, and S. Boudet, "2½D visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, 1999.

[46] P. Murrieri, D. Fontanelli, and A. Bicchi, "A hybrid-control approach to the parking problem of a wheeled vehicle using limited view-angle visual feedback," *The International Journal of Robotics Research*, vol. 23, no. 4-5, pp. 437–448, 2004.

[47] Y. Fang, W. E. Dixon, D. M. Dawson, and P. Chawda, "Homography-based visual servo regulation of mobile robots," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 5, pp. 1041–1050, 2005.

[48] C. Colombo, B. Allotta, and P. Dario, "Affine visual servoing: A framework for relative positioning with a robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1995, pp. 464–471.

[49] P. Questa, E. Grossmann, and G. Sandin, "Camera self orientation and docking maneuver using normal flow," in *Proceedings of the SPIE's 1995 Symposium on OE/Aerospace Sensing and Dual Use Photonics*, 1995, pp. 274–283.

[50] A. Cretual and F. Chaumette, "Positionning a camera parallel to a plane using dynamic visual servoing," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, vol. 1, 1997, pp. 43–48.

[51] L. A. Zadeh, "Fuzzy sets," *Information Control*, vol. 8, no. 3, pp. 338–353, 1965.

[52] S. S. Chang and L. A. Zadeh, "On fuzzy mapping and control," *IEEE Transactions on Systems, Man and Cybernetics*, no. 1, pp. 30–34, 1972.

[53] K. Tanaka and M. Sano, "Trajectory stabilization of a model car via fuzzy control," *Fuzzy Sets and Systems*, vol. 70, no. 2-3, pp. 155–170, 1995.

[54] Y. C. Chiou and L. W. Lan, "Genetic fuzzy logic controller: An iterative evolution algorithm with new encoding method," *Fuzzy Sets and Systems*, vol. 152, no. 3, pp. 617–635, 2005.

[55] A. Homaifar and E. McCormick, "Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 2, pp. 129–139, 1995.

[56] L. A. Zadeh, "Fuzzy algorithm," *Information Control*, vol. 12, pp. 94–102, 1968.

[57] C. Von Altrock, B. Krause, and H. J. Zimmermann, "Advanced fuzzy logic control technologies in automotive applications," in *Proceedings of IEEE International Conference on Fuzzy Systems*, 1992, pp. 835–842.

[58] T. Das and I. N. Kar, "Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 3, pp. 501–510, 2006.

[59] V. M. Peri and D. Simon, "Fuzzy logic controller for an autonomous robot," *in Proceedings of NAFIPS 2005-2005 Annual Meeting of the North American Fuzzy Information Processing Society*, pp. 337–342, 2005.

[60] M. K. Singh, D. R. Parhi, S. Bhowmik, and S. K. Kashyap, "Intelligent controller for mobile robot: Fuzzy logic approach," in *Proceedings of the 12th International Conference of International Association for Computer Methods and Advances in Geomechanics (IACMAG)*, 2008, pp. 1–6.

[61] J. Xu, G. Chen, and M. Xie, "Vision-guided automatic parking for smart car," in *Proceedings of IEEE Intelligent Vehicle Symposium*, 2000, pp. 725–730.

[62] Y. Zhao and E. G. Collin, "Robust automatic parallel parking in tight spaces via fuzzy logic," *Robotics and Autonomous Systems*, vol. 51, no. 2, pp. 111–127, 2005.

[63] T. H. Li and S. J. Chang, "Autonomous fuzzy parking control of a car-like mobile robot," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 33, no. 4, pp. 451–465, 2003.

[64] T. H. Li, S. J. Chang, and Y. X. Chen, "Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 5, pp. 867–880, 2003.

[65] I. Baturonel, F. J. Moreno-Velo, S. Sánchez-Solano, and A. Ollero, "Automatic design of fuzzy controllers for car-like autonomous robots," *IEEE Transacions on Fuzzy Systems*, vol. 12, no. 4, pp. 447–465, 2004.

[66] M. Sugeno, "An experimental study on fuzzy parking control using a model car," *Industrial Applications of Fuzzy Control*, pp. 105–124, 1985.

[67] A. Razinkova, H. C. Cho, and H. T. Jeon, "An intelligent auto parking system for vehicles," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 12, no. 3, pp. 226–231, 2012.

[68] F. Gómez-Bravo, F. Cuesta, and A. Ollero, "Parallel and diagonal parking in non-holonomic autonomous vehicles," *Engineering Applications of Artificial Intelligence*, vol. 14, no. 4, pp. 419–434, 2001.

[69] W. A. Daxwanger and G. K. Schmidt, "Skill-based visual parking control using neural and fuzzy networks," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, 1995, pp. 1659–1664.

[70] S. Yasunobu and Y. Murai, "Parking control based on predictive fuzzy control," in *Proceedings of 1994 IEEE International Fuzzy Systems Conference*, 1994, pp. 1338–1341.

[71] M. Ohkita, H. Miyata, M. Miura, and H. Kuono, "Traveling experiment of an autonomous mobile robot for a flush parking," in *Proceedings of 2nd IEEE International Conference on Fuzzy Systems*, 1993, pp. 327–332.

[72] M. C. Leu and T. Q. Kim, "Cell mapping based fuzzy control of car parking," in *Proceedings of 1998 IEEE International Conference on Robotics Automation*, 1998, pp. 2494–2499.

[73] K. Y. Lian, C. S. Chin, and T. S. Chiang, "Parallel parking a car-like robot using fuzzy gain scheduling," in *Proceedings of 1999 IEEE International Conference on Control Applications*, 1999, pp. 1686–1691.

[74] Q. Liu, Y. G. Lu, and C. X. Xie, "Fuzzy obstacle-avoiding controller of autonomous mobile robot optimized by genetic algorithm under multi-obstacles environment," in *Proceedings of 2006 6th World Congress on Intelligent Control and Automation*, 2006, pp. 3255–3259.

[75] C. Rekik, M. Jallouli, and N. Derbel, "Optimal trajectory of a mobile robot by a genetic design fuzzy logic controller," in *Proceedings of 2009 International Conference on Advances in Computational Tools for Engineering Applications*, 2009, pp. 107–111.

[76] J. H. Holland, *Adaptation in Natural and Artificial Systems.* MIT Press, Cambridge, MA, 1992.

[77] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison Wesley, New York, 1989.

[78] A. Homaifar and E. McCormick, "Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 2, pp. 129–139, 1995.

[79] C. C. Wong, H. Y. Wang, S. A. Li, and C. T. Cheng, "Fuzzy controller designed by GA for two-wheeled mobile robots," *International Journal of Fuzzy Systems*, vol. 9, no. 1, pp. 22–30, 2007.

[80] L. Renhou and Z. Yi, "Fuzzy logic controller based on genetic algorithms," *Fuzzy Sets and Systems*, vol. 83, no. 1, pp. 1–10, 1996.

[81] A. Arslan and M. Kaya, "Determination of fuzzy logic membership functions using genetic algorithms," *Fuzzy Sets and Systems*, vol. 118, no. 2, pp. 297–306, 2001.

[82] F. Hoffmann and G. Pfister, "Evolutionary design of a fuzzy knowledge base for a mobile robot," *International Journal of Approximate Reasoning*, vol. 17, no. 4, pp. 447–469, 1997.

[83] R. Zhao, D. H. Lee, and H. K. Lee, "Mobile robot navigation using optimized fuzzy controller by genetic algorithm," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 15, no. 1, pp.12–19, 2015.

[84] I. Hassanzadeh and S. M. Sadigh, "Path planning for a mobile robot using fuzzy logic controller tuned by GA," in *Proceedings of the 6th International Symposium on Mechatronics and its Applications (ISMA)*, 2009, pp. 1–5.

[85] T. Khelchandra, J. Huang, and S. Debnath, "Path planning of mobile robot with neuro-genetic-fuzzy technique in static environment," *International Journal of Hybrid Intelligent Systems*, vol. 11, no. 2, pp. 71–80, 2014.

[86] S. H. Azadi and Z. Taherkhani, “Autonomous parallel parking of a car based on parking space detection and fuzzy controller,” *International Journal of Automotive Engineering*, vol. 2, no. 1, pp. 30–37, 2012.

[87] Y. Zhao, E. G. Collins, and D. Dunlap, “Design of genetic fuzzy parallel parking control systems,” in *Proceedings of IEEE American Control Conference*, 2003, pp. 4107–4112.

[88] D. Leitch and P. J. Probert, “New techniques for genetic development of a class of fuzzy controllers,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 28, no. 1, pp. 112–123, 1998.

[89] T. Mita, *Introduction to Nonlinear Control*. Shokodo, Tokyo, 2000.

[90] S. G. Tzafestas, *Introduction to Mobile Robot Control.* Elsevier, Athens, Greece, 2013.

[91] Wikipedia, “Hough transform,” https://en.wikipedia.org/wiki/Houghtransform, 2016.

[92] E. H. Mamdani, “Advances in the linguistic synthesis of fuzzy controllers," *International Journal of Man-Machine Studies*, vol. 8, no. 6, pp. 669–678, 1976.

[93] Weisstein, Eric W. Least Squares Fitting. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/LeastSquaresFitting.html

[94] M. Gen, R. Cheng, *Genetic Algorithms And Engineering Design.* John Wiley and Sons, 1997.

Appendix A

Mamdani Fuzzy Inference System

The steps of Mamdani fuzzy system is as follows:

- **Determining linguistic variables and fuzzy sets:** Let the two inputs be represented as linguistic variables A and B; and the output a linguistic variable C. A_1, A_2 and A_3 are linguistic values for A; B_1, B_2 and B_3 are linguistic values for B; C_1, C_2 and C_3 are linguistic values for C with membership functions. Let us define three rules as follows:

  R_1: IF (A is A_1) AND (B is B_1) THEN (C is C_1),

  R_2: IF (A is A_2) AND (B is B_2) THEN (C is C_2),

  R_3: IF (A is A_3) AND (B is B_3) THEN (C is C_3)

- **Step 2- Fuzzification:** This is the process of generating membership values for a fuzzy variable using membership functions. The first step is to take the crisp inputs and determine the degree to which these inputs belong to each appropriate fuzzy set. This crisp input is always a numeric value limited to the universe of discourse. Once the crisp inputs are obtained, they are fuzzified against the appropriate linguistic fuzzy sets.

- **Step 3- Fuzzy Inferencing (Evaluate Rules):** This is the third step where the fuzzified inputs are applied to the antecedents of the fuzzy rules. Since the fuzzy rule has multiple antecedents, fuzzy operator (*AND* or *OR*) is used to obtain a single number that represents the result of the antecedent evaluation.

  In case of which several rules are active for the same output membership function, it is necessary that only one membership value is chosen. This process is called “fuzzy inference.” The Mamdani method is given below:

$$\mu_c(y) = \max\left[\min\left[\mu_A(input(i)), \mu_B(input(j)), \ldots, \mu_N(input(n))\right]\right] \quad \text{(A.1)}$$

This expression determines an output membership function value for each active rule. When one rule is active, an *AND* operation is applied between inputs. The smaller input value is chosen and its membership value is determined as membership value of the output for that rule. This method is repeated, so that output membership functions are determined for each rule. To sum up, graphically *AND* (min) operation are applied between inputs and *OR* (max) operations are between output.

- **Step 4- Rules Output Aggregation:** Aggregation is the process of unification of the outputs of all rules. In other words, take the membership functions of all rule consequents previously clipped or scaled and combine them into a single fuzzy set. Having evaluated all the rules, the final shape of the output is determined by combining all of the activated rule consequents.

- **Step 5- Defuzzification:** This is the last step in the fuzzy inference process, which is the process of transforming a fuzzy output of a fuzzy inference system into a crisp output. The input for the defuzzification process is the aggregate output fuzzy set and the output is a number.

Appendix B

Sugeno Fuzzy Inference System

A typical rule in a Sugeno fuzzy model has the form

R: IF $Input_1 = x$ AND $Input_2 = y$, THEN $Output$ is $z = ax + by + c$.

For a zero-order Sugeno model, the output level z is a constant ($a = b = 0$). The output level z_i of each rule is weighted by the firing strength w_i of the rule. For example, for an AND rule with $Input_1 = x$ and $Input_2 = y$, the firing strength is:

$w_i = ANDMethod\ (F_1(x), F_2(y))$,

where, $F_{1,2}(.)$ are the membership functions for $Input_1$ and $Input_2$. The final output of the system is the weighted average of all outputs, computed as:

$$\text{Final Output} = \frac{\sum_{i=1}^{N} w_i z_i}{\sum_{i=1}^{N} w_i} \tag{B.1}$$

Figure B.1 shows the fuzzy reasoning procedure for a first-order Sugeno fuzzy model. Since each rule has a crisp output, the overall output is obtained via weight average, thus avoiding the time-consuming process of defuzzification requied in a Mamdani model. In practice, the weighted average operator is sometimes replaced with the weighted sum operator (that is, $z = w_1 z_1 + w_2 z_2$ in Fig. B.1) to reduce computation further, especially in the training of a fuzzy inference system. However, this simplification could lead to the loss of MF linguistic meanings unless the sum of firing strengths is close to unity. Since the only fuzzy part of a Sugeno model is in its antecedent, it is easy to demonstrate the distinction between a set of fuzzy rules and non-fuzzy ones.

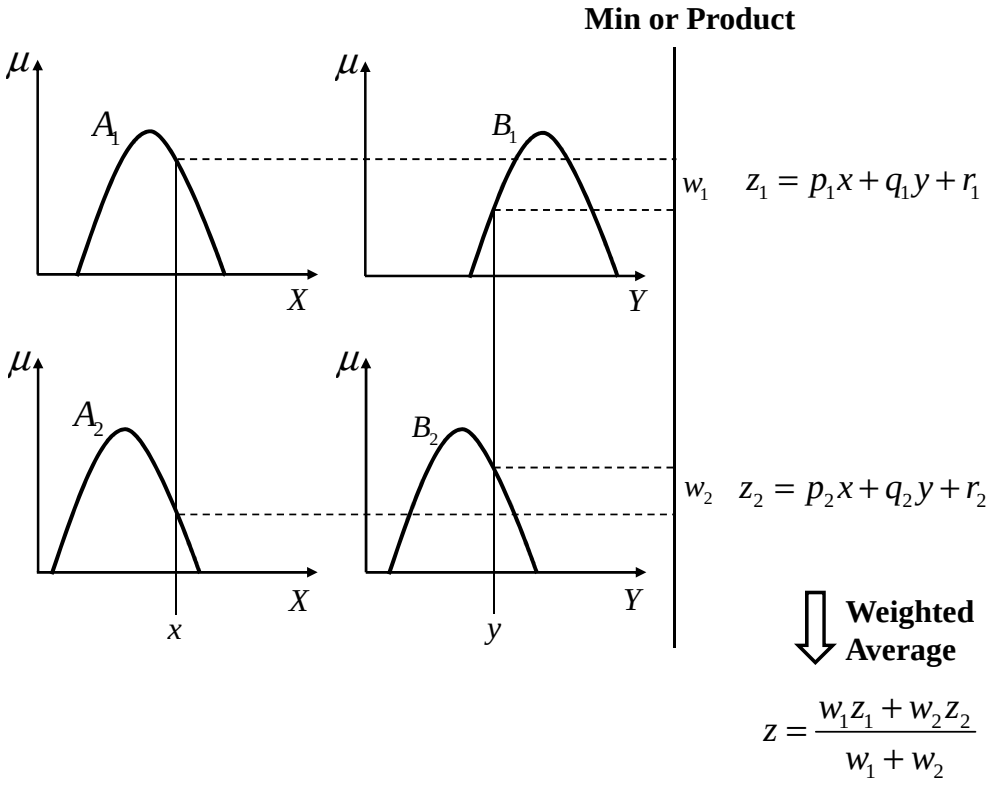


Fig. B.1 The Sugeno fuzzy model

Appendix C

Genetic Algorithms

A GA is a computational technique that simulates evolution in the search for solutions to complex problems. These are problems with large spaces of potential solutions, characterized by nonlinearities. This means there is no linear relation between the performance of solutions as classified along the dimensions used for characterizing them. If there were linear relations, the optimal solution could easily be found by a simple hill-climbing algorithm. A GA assesses the performance of different solutions in parallel and uses the information about how different solutions perform to direct the search towards promising areas of the search space. Adapting the search is done by simulating evolutionary processes.

Chromosome: A chromosome is the set of input variables for the GA. This can be considered the genetic material constituting the solution you are solving for.

Population: The genetic population is a randomly generated group of chromosomes.

Fitness function: The fitness function is a performance index such that the fittest chromosomes in the population will survive to the next generation. The fitter the parent, the greater the probability of selection for reproduction.

Fitness scaling: Fitness scaling converts the scores achieved from the fitness function to a range suitable for the function selected for reproduction.

Rank scaling: Rank scaling orders each individual according to their fitness, so that the strongest comes first and the weakest last.

Fitness function tolerance: The fitness function tolerance defines the stopping condition for the genetic algorithm. When the change in the weighted fitness function value becomes less than the fitness function tolerance the genetic algorithm stops.

Reproduction:. Reproduction is the process where a selected chromosome is combined with another chromosome (the parents) to form a new chromosome of different genetic composition (the offspring).

Stochastic uniform: The stochastic uniform function implies that a good mix of genes (weak and strong) would be selected at a uniform random interval for breeding.

Parents: The parents are the selected chromosomes for reproduction.

Offspring: The offspring are the results of reproduction between two parents (crossover), or a single parent (mutation).

Crossover: Crossover determines the amount of genetic material from each parent that contributes to the new offspring. This is illustrated in Fig. C.1.

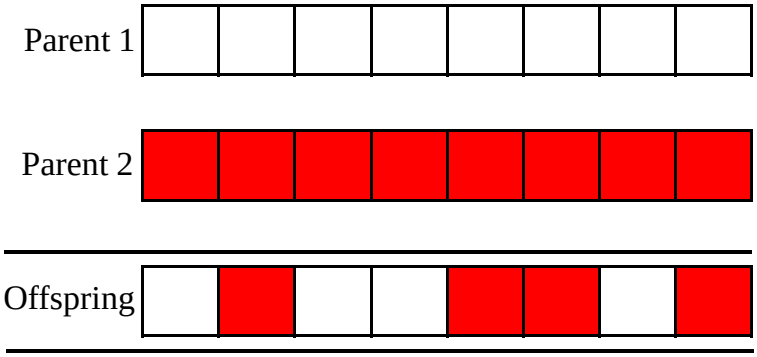


Fig. C.1 Illustration of parental crossover in genetic reproduction

Scattered function: The scattered function ensures that each of the offspring inherited a random amount of the two parents' genes thus giving rise to a diverse population.

Mutation: Mutation is the random change in a gene within the chromosome of the offspring, as illustrated in Fig. C.2. This is controlled by the mutation rate. Mutation allows for greater survey of the problem space.

Adaptive scattered function: The adaptive scattered function allowed for a randomly generated number to be added to a random chromosome to make it differ from the original chromosome.

Generations: Once the new population has be created, the genetic algorithm progresses to the next generation. This is a factor that indicates how many times the population has changed.

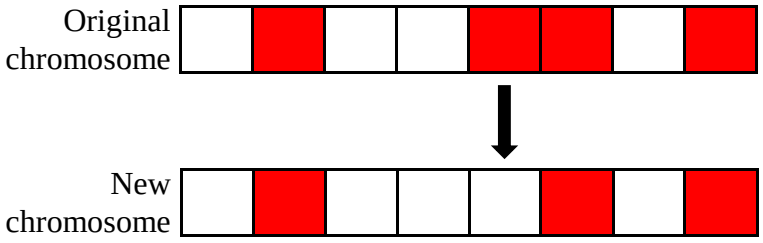


Fig. C.2 Illustration of mutation of a chromosome on the fifth gene