Implementation of Human-Like Driving Skills by Autonomous Fuzzy Behavior Control on an FPGA-Based Car-Like Mobile Robot

Tzuu-Hseng S. Li, *Member, IEEE*, Shih-Jie Chang, and Yi-Xiang Chen

***Abstract*—In this paper, the concepts of car maneuvers, fuzzy logic control (FLC), and sensor-based behaviors are merged to implement the human-like driving skills by an autonomous car-like mobile robot (CLMR). Four kinds of FLCs, fuzzy wall-following control, fuzzy corner control, fuzzy garage-parking control, and fuzzy parallel-parking control, are synthesized to accomplish the autonomous fuzzy behavior control (AFBC). Computer simulation results illustrate the effectiveness of the proposed control schemes. The setup of the CLMR is provided, where the implementation of the AFBC on a field-programmable gate array chip is also addressed. Finally, the real-time implementation experiments of the CLMR in the test ground demonstrate the feasibility in practical car maneuvers.**

***Index Terms*—Autonomous fuzzy behavior control (AFBC), car-like mobile robot (CLMR), field-programmable gate array (FPGA), fuzzy logic control (FLC), garage parking, parallel parking, real-time implementation.**

I. Introduction

THE parking problem of a car-like mobile robot (CLMR) has been widely investigated for decades. The parking problem of a nonholonomic vehicle is usually to find a path that connects the initial configuration to the final one with collision-free capability. Much research [1]–[16] has proposed various control strategies on this topic. Numerous studies [1]–[9] adopted the fuzzy logic control (FLC) method to solve this problem. They usually derived fuzzy rules to model the parking experiences of a skilled driver to perform the parking task. Some researchers also combined the FLC with other algorithms, for example, neural networks [5] or genetic algorithms [6], to improve the feasibility of the autonomous parking control. References [9]–[16] presented some other maneuverable methods for parking problems. These methods consist of several stages or generate several maneuvers by using the specified paths (like circles, straight-line segments, sinusoids, etc.) from a starting location to a goal position.

Generally speaking, the kinematic equations of a car with nonholonomic constraints are nonlinear and time-varying differential equations. It is not easy to find an autodriving scheme by traditional control methods in performing the parking problem. However, human drivers can smoothly and even perfectly turn a corner, drive in a lane, and park the car in a garage or parking lot by some driving skills or simple instinct rules without the knowledge of the motion kinematics of the car.

Steering a car is confined to the conditions of the car's own capability of mechanism and the environment. Due to these reasons, we do not expect to design a continuously global controller for a car to perform all the maneuvering behaviors. Alternatively, we figure out some local controllers by a few experienced rules to maneuver in the specified region. Thus, we propose the fuzzy behavior control to integrate all the local controllers.

The field-programmable gate array (FPGA) chip has become the preferred device for many researchers and engineers because it can be reconfigured virtually and instantaneously to produce a physical circuit, which can be evaluated in real time. Recently, using a reconfigurable FPGA to implement an algorithm is very attractive because the FPGA offers a compromise between application specific-integrated circuit (ASIC) hardware and general-purpose processors. Numerous instances in the literature [17]–[22] have implemented the fuzzy logic control on the FPGA or the VLSI chip for mobile robots, such as in [22], where a low-cost microcontroller and an FPGA to guide the autonomous vehicle are adopted.

In this paper, we address the concept of the autonomous fuzzy behavior control (AFBC), which is composed of several maneuvers and sensor-based behaviors by using the FLCPs. The design procedure of the human-like AFBC is introduced and the implementation on an FPGA chip is also addressed. We will also specify the hardware architecture of the FPGA and setup on the CLMR to accomplish wall-following, corner-turning, garage-parking, and parallel-parking behaviors.

This paper is organized as follows. In Section II, the kinematics of the CLMR are first introduced. The fuzzy wall-following control (FWFC), the fuzzy corner control (FCC), the fuzzy garage-parking control (FGPC), and the fuzzy parallel-parking control (FPPC) are investigated. Computer simulation results of the proposed fuzzy behavior control algorithms are also given. Section III addresses the hardware architecture of the

Manuscript received September 5, 2002; revised June 11, 2003. Abstract published on the Internet July 9, 2003. This work was supported by the National Science Council, Taiwan, R.O.C., under Grant NSC89-2213-E006-187, Grant NSC90-2213-E006-052, and Grant NSC91-2213-E006-026.

T.-H. S. Li is with the IC^{2}S Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 70101, Taiwan, R.O.C. (e-mail: thsli@mail.ncku.edu.tw).

S.-J. Chang was with the IC^{2}S Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 70101, Taiwan, R.O.C. He is now with Acer Inc., Taoyuan, Taiwan, R.O.C.

Y.-X. Chen was with the IC^{2}S Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 70101, Taiwan, R.O.C. He is now with Primax Electronics Ltd., Taipei, Taiwan, R.O.C.

Digital Object Identifier 10.1109/TIE.2003.817490

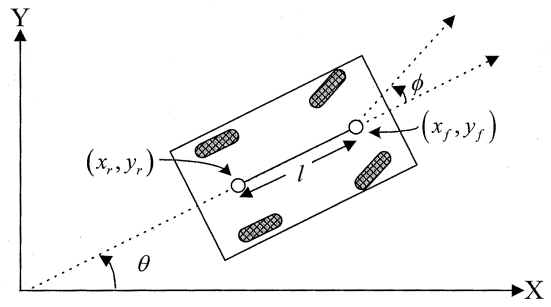


Fig. 1. Kinematic model of a CLMR.

CLMR. Implementation of the AFBC on the FPGA chip is discussed in Section IV. In Section V, the real-time implementation of the AFBC for the CLMR is presented. Section VI concludes this paper.

II. Fuzzy Behavior Control Design

The motion of the CLMR is first addressed in this section. Four kinds of control designs, wall following, corner turning, garage parking, and parallel parking, are proposed. All the driving behaviors are developed based on the left-hand driver system. Computer simulation results of each control scheme are given to show the feasibility. Finally, we provide a behavior selection mechanism to integrate these four controllers.

A. Kinematic Model of a CLMR

Consider a kinematic model of the CLMR as shown in Fig. 1, where the rear wheels are fixed parallel to the car body and allowed to roll or spin but not slip. The front wheels can turn to the left or right, but the left and right front wheels must be parallel. All the corresponding parameters of the CLMR are also defined in Fig. 1, where (x_f, y_f) is the position of the front wheel center of the CLMR, (x_r, y_r) is the position of the rear wheel center of the CLMR, ϕ is the orientation of the steering wheels with respect to the frame of the CLMR, θ is the angle between the vehicle frame orientation and X axis, and l is the wheelbase of the CLMR.

The rear-wheel kinematic equation [12] of the CLMR is described by

$$\begin{aligned} \dot{x}_r &= v\cos\theta\cos\phi \\ \dot{y}_r &= v\sin\theta\cos\phi \\ \dot{\theta} &= v\frac{\sin\phi}{l} \end{aligned} \tag{1}$$

where v is the speed of the front wheels. Equation (1) presents the backward movement of the CLMR. The front-wheel kinematic equation denoting forward motion of the CLMR is depicted as

$$\begin{aligned} \dot{x}_f &= v\cos(\theta+\phi) \\ \dot{y}_f &= v\sin(\theta+\phi) \\ \dot{\theta} &= v\frac{\sin\phi}{l}. \end{aligned} \tag{2}$$

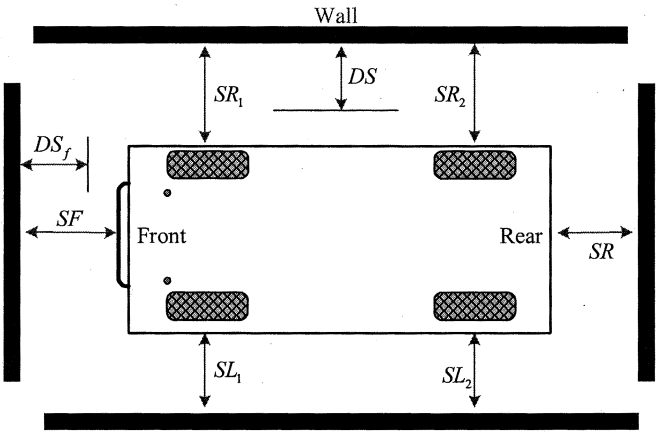


Fig. 2. Definitions of distances between the CLMR and walls.

B. FWFC Design

In the beginning, we assume that the relative distances between the CLMR and the walls can be measured. Fig. 2 illustrates the definitions of distances, where SF is the distance between the front of the CLMR and the wall, SR_1 is the distance between the front-right wheel of the CLMR and the wall, SR_2 is the distance between the rear-right wheel of the CLMR and the wall, SL_1 is the distance between the front-left wheel of the CLMR and the wall, SL_2 is the distance between the rear-left wheel of the CLMR and the wall, and SR is the distance between the rear of the CLMR and the wall. The real implementation of measuring these distances will be discussed in Section III. The steering angle control and the speed control of the CLMR are described as follows.

1) Steering Angle Control: As an experienced driver drives a car on a straight road, he can roughly describe his driving strategy on the straight road in terms of some linguistic rules. According to drivers' experiences, we can derive a two-input–single-output FLC scheme to command the steering angle of the front wheels for the wall-following task. First, we introduce x_d and x_e as the input linguistic variables, which can be determined by the distance sensors depicted in Fig. 2. As the robot is moving forward, variable x_d is defined as $SR_1 - SL_1$ and variable x_e is defined as $SR_1 - SR_2$; as the robot is moving backward, variable x_d is defined as $SR_2 - SL_2$ and variable x_e is defined as $SR_2 - SR_1$. Variable x_d presents whether the CLMR is on the centerline between two walls or not. Variable x_e denotes the inclination situation of the CLMR between the walls. The output of the FLC is the steering angle ϕ.

Input variables x_d and x_e are decomposed into five fuzzy partitions with triangular membership functions, and output variable ϕ is the fuzzy singleton-type membership function with five partitions. The partitions and the shapes of the membership functions are shown in Fig. 3(a) and (b), where fuzzy term sets are denoted by NB (Negative Big), NS (Negative Small), ZE (Zero), PS (Positive Small), and PB (Positive Big). The fuzzy reasoning rules for the wall-following behavior are summarized in Table I(a). The defuzzification strategy is the weighted average method.

2) Speed Control: For speed control, the driving skill can be achieved by the following situations. If the car is very far away

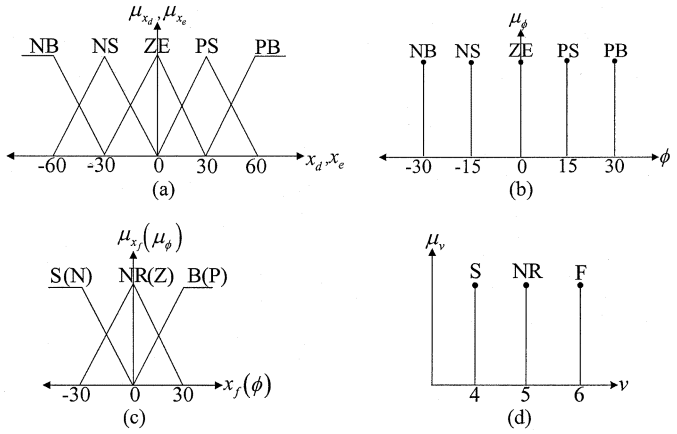


Fig. 3. (a) Membership function of the input variables x_d and x_e for the steering angle control. (b) Membership function of the output variable ϕ for the steering angle control. (c) Membership function of the input variables x_f and ϕ for the speed control. (d) Membership function of the output variable v for the speed control.

TABLE I
(a) FUZZY RULE TABLE FOR THE STEERING ANGLE CONTROL OF THE FWFC.
(b) FUZZY RULE TABLE FOR THE SPEED CONTROL OF THE FWFC

x_e \ x_d	**NB**	**NS**	**ZE**	**PS**	**PB**
NB	NB	NB	NS	NS	ZE
NS	NB	NS	NS	ZE	PS
ZE	NS	NS	ZE	PS	PS
PS	NS	ZE	PS	PS	PB
PB	NE	PS	PS	PB	PB

(a)

ϕ \ x_f	**S**	**NR**	**B**
P	S	S	NR
Z	S	NR	F
N	S	S	NR

(b)

from the front wall and the steering wheel command is ZE, we will speed up the car. If the car is also very far away from the front wall but the steering wheel command is a little turning to the right-hand side (or the left-hand side), we can just hold the car to a normal speed. If the car is just a little away from the front wall and the steering wheel command is ZE, we will also hold the car to a normal speed. If the car becomes gradually close to the front wall, we can slow down the car.

According these experiences, we can derive the two-input–single-output FLC scheme to command the speed of the CLMR. First, we introduce x_f and ϕ as the input linguistic variables. For moving forward cases, variable x_f is defined as $SF - DS_f$, where DS_f is the desired distance between the front of the CLMR and the wall. The distance sensors shown in Fig. 2 can also determine these variables. The output of the FLC is the speed v.

The partitions and the shapes of the membership functions for x_f, ϕ, and v are shown in Fig. 3(c) and (d), where x_f is divided into three term-sets, denoted by S (Small), NR (Normal), and B (Big), ϕ determined by the steering angle control is divided into three term-sets, presented by P (Positive), Z (Zero) and N (Negative), and v is also partitioned into three term-sets singletons, expressed by S (Slow), NR (Normal), and F (Fast). The fuzzy reasoning rules for the speed control of the wall following are summarized in Table I(b). The defuzzification strategy is also the weighted average method.

C. FCC Design

In our driving experiences, we always turn before meeting the corner. Thus, when a car enters a corner area, we will slow the speed and turn right as far as we possibly can. Supposing the car cannot successfully turn a corner, we will back the car a certain distance and then go forward to turn the corner again to avoid collision. This process will be applied to the corner behavior in our experiment. Fig. 4(a) shows the flowchart of the FCC design procedure, where the forward and backward FWFC is the same as that designed in Section II-B.

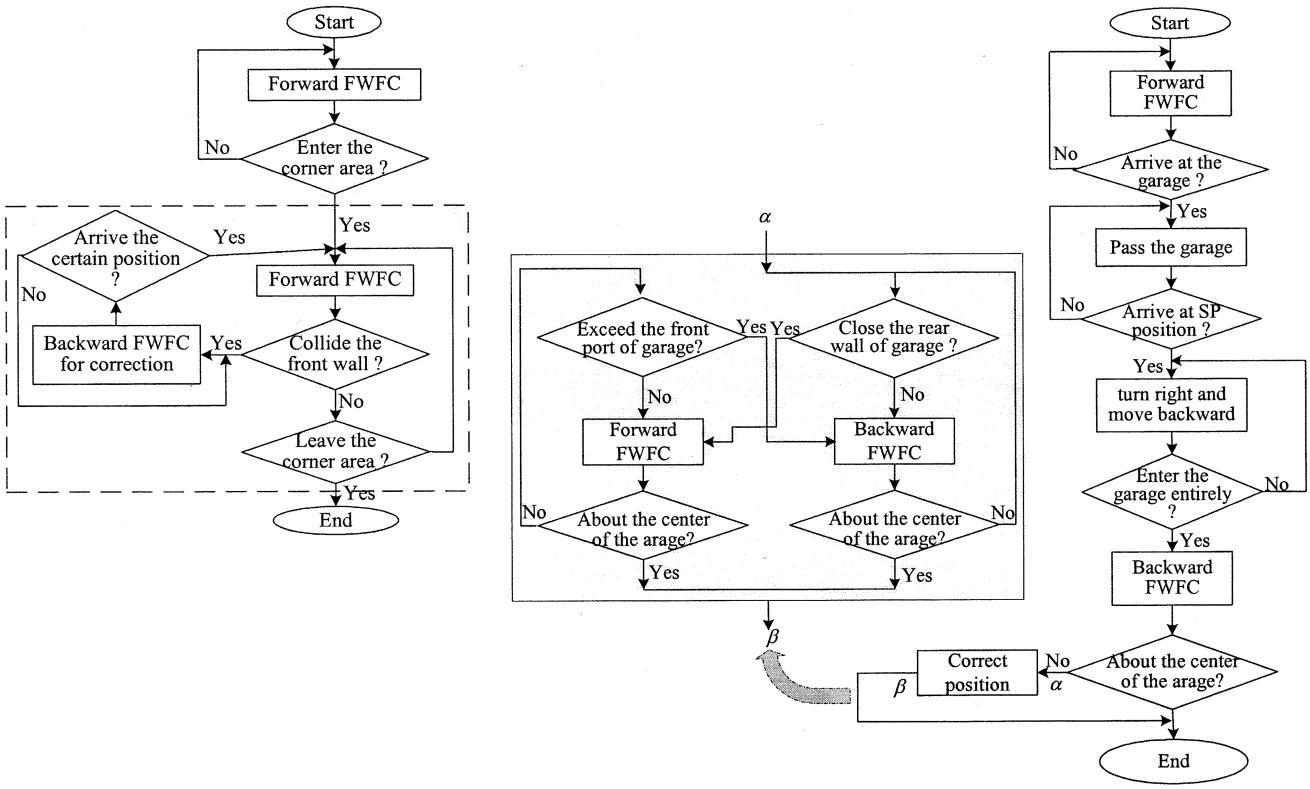
D. FGPC Design

For the garage parking problem, a driver always passes the garage and then backs into it. In fact, backward-garage parking needs a less narrow width of the garage than that in forward-garage parking. According to our driving experience, we will first pass the garage by a little distance. This action will allow us to easily back the car into garage. Then, we turn the steering wheel right and move backward to the garage. When a car entirely backs into the garage, we will move the car back and forth to an appropriate position. The process of the garage-parking behavior is shown in Fig. 5(a).

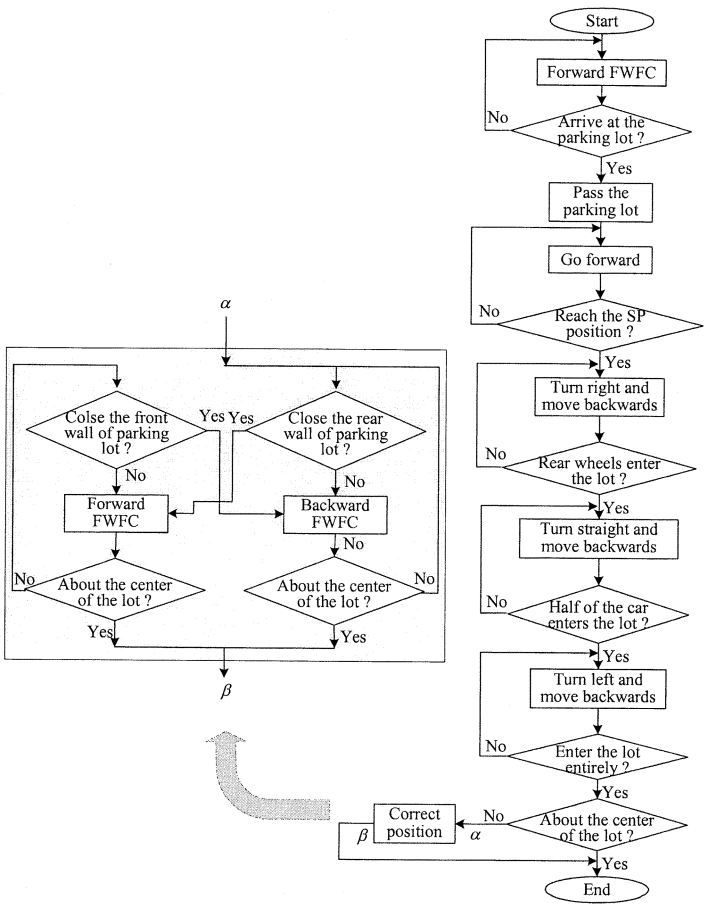
We apply these driving skills to design the FGPC. When the CLMR backs into the garage, we exploit the backward FWFC to correct the CLMR to an appropriate position. When the CLMR is very close to the rear of the garage, we let the CLMR go forward by using the forward FWFC to correct the position of the CLMR. Suppose the front wheels of the CLMR exceed the garage, then we let the CLMR move backward by using the backward FWFC to make corrections. If the CLMR still cannot park in the appropriate position, we drive the CLMR back and forth until the CLMR is at about the center of the garage. The FGPC behavior is illustrated by the flowchart depicted in Fig. 4(b).

E. FPPC Design

Usually, parallel-parking lots on the street are designed for backward parking. Therefore, a driver always passes the parking lot and then backs the car in. The process of the backward parallel parking is sketched in Fig. 5(b). The flowchart of the proposed FPPC scheme is illustrated in Fig. 4(c). The design procedure of the FPPC can be achieved by the following knowledge. We always drive the car passing the parallel-parking lot and go forward a little distance to allow us to back the car into the lot easily. Once the car reaches the prespecified SP position, we make the steering wheel turn right and begin to move backward into the lot. As the rear wheels enter the parking lot, we turn the steering wheel to straight until half of the car is entering the lot, and then we turn the steering wheel to the left until the car entirely backs into the lot. Finally, we will



(a) (b)



(c)

Fig. 4. (a) Flowchart of the FCC procedure. (b) Flowchart of the FGPC procedure. (c) Flowchart of the FPPC procedure.

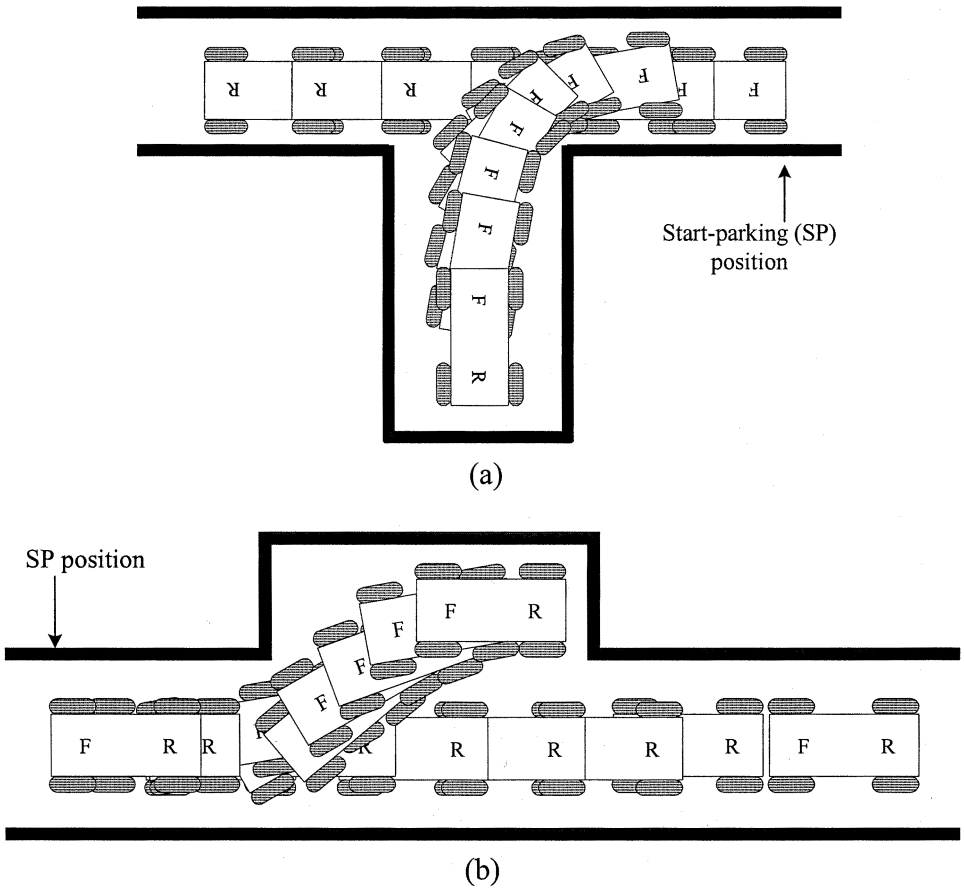


Fig. 5. (a) Process of the backward garage parking. (b) Process of the backward parallel parking.

correct the position of the car by moving it to and fro until it is about in the center of the parallel-parking lot. In fact, the correction process is the same as that in the garage-parking behavior.

F. Simulation Results

Computer simulation results are given to demonstrate the effectiveness of all the proposed control schemes. We exploit the kinematic equations (1) and (2) of the CLMR and calculate the relative distances (SF, SR, SR_1, SR_2, SL_1, and SL_2) among the CLMR, the wall, garage, and parking lot. Fig. 6(a) and (b) illustrates the wall-following capability of the CLMR by applying the FWFC scheme from two initial postures. Fig. 6(c) gives the simulation result of the FCC. Simulation results of the FGPC for two initial postures are shown in Fig. 6(d) and (e). Finally, Fig. 6(f) and (g) demonstrates the FPPC performance. It is seen very clearly from these figures that all the developed FLC methods can successfully accomplish their control goals.

In fact, simulations of these FLCs for different partitions of the membership functions are also performed. It is found that we cannot correctly drive the CLMR into the parking lot if the membership functions of x_d, x_e, and ϕ are with three partitions for steering angle control. The performance between the five and seven partitions of the membership functions are not obvious. For speed control, we find three fuzzy term sets for x_f, ϕ, and v are good enough to reach the speed control goal. Simulation results of three and seven partitions are not shown in this paper due to space limitations. The position and width of the membership functions are determined by equalizing the universe of discourses of the input and output variables.

G. Behavior Selection Mechanism

In order to evaluate the feasibility of all the control schemes, we set up a test ground for real experiments, which will be examined in Section V. The diagram of the test ground is shown in Fig. 7, where four behaviors, wall following, corner, parallel parking, and garage parking, are included. The CLMR will execute one of these behaviors when it travels in the test ground. The behavior selection mechanism should switch an appropriate behavior by the measuring data of the sensors shown in Fig. 2. Each behavior is independent of the others. Once one behavior is selected, the other behaviors will be inhibited.

The wall-following mode is set as the default behavior of the CLMR. All the distance values of the front, the front-right wheel, and the rear-right wheel of the CLMR are first compared with the prespecified decision values, and then we can decide which behavior should be executed. The decision rules of the selection mechanism are described as follows.

$$\begin{aligned}\text{Rule 1}: &\text{If } SF < SF_{sv} \text{ and } SR_1 < SR_{1,sv} \\ &\text{and } SR_2 < SR_{2,sv}, \\ &\text{then the CLMR is in the FCC.}\end{aligned} \tag{3a}$$

$$\begin{aligned}\text{Rule 2}: &\text{If } SF \gg SF_{sv} \\ &\text{and } SR_1 < SR_{1,sv} \text{ and } SR_2 < SR_{2,sv}, \\ &\text{then the CLMR is in the FPPC.}\end{aligned} \tag{3b}$$

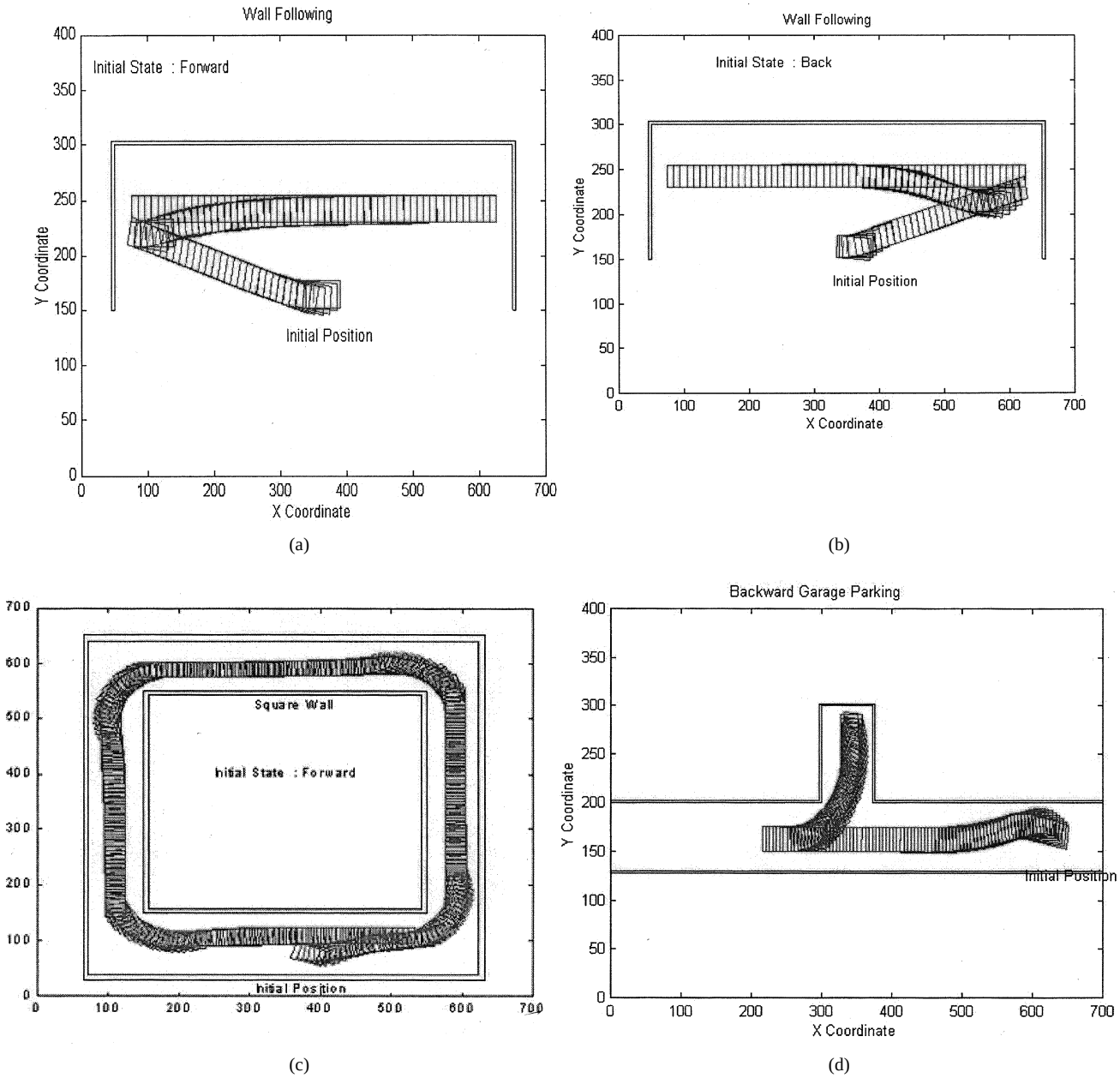


Fig. 6. Simulation results of all the proposed control schemes. (a) Simulation result of the FWFC with initial posture $(x_f, y_f, \theta) = (390°, 165°, 0°)$. (b) Simulation result of the FWFC with initial posture $(x_f, y_f, \theta) = ((390°, 165°, 0°)$. (c) simulation result of the FCC with initial posture $(x_f, y_f, \theta) = (400\text{o}, 70\text{o}, -20°)$. (d) Simulation result of the FGPC with initial posture $(x_f, y_f, \theta) = (650\text{o}, 165\text{o}, -10°)$.

$$\text{Rule 3}: \text{If } SF \gg SF_{sv} \text{ and } SR_1 > SR_{1,sv} \text{ and } SR_2 > SR_{2,sv}, \text{ then the CLMR is in the FGPC.} \quad (3c)$$

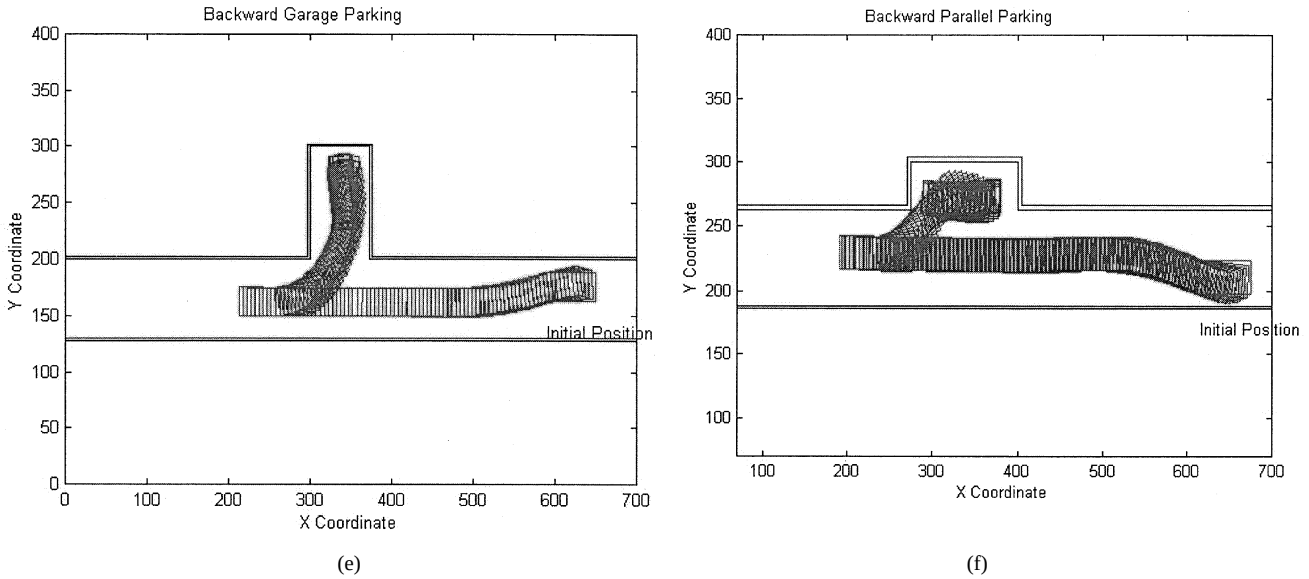
In the above, SF_{sv}, $SR_{1,sv}$, and $SR_{2,sv}$ are the prespecified distance values for SF, SR_1, and SR_2, respectively. The flowchart of the behavior selection scheme is illustrated in Fig. 8.

III. Hardware Architecture of the CLMR

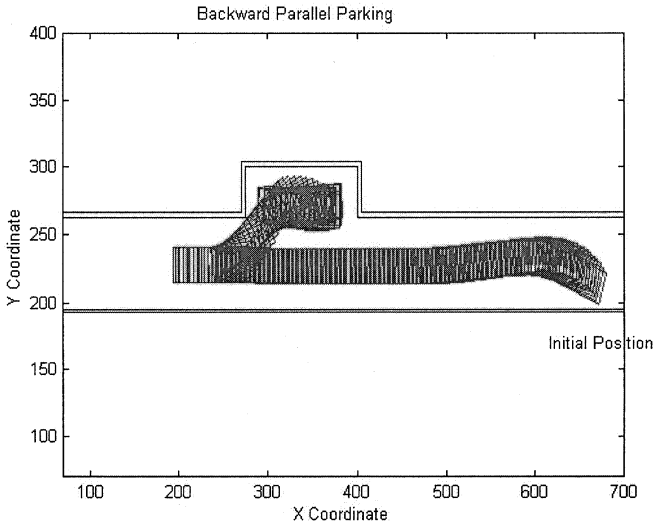
The realization of setting up a CLMR is depicted in this section. In order to simulate the behaviors of a real car, we adopt a 1/10th-scale four-wheeled model car with front-wheel drive and front steering wheels as the chassis mechanism of the CLMR. This car has two kinds of motors, one is the dc motor used for controlling the speed, and the other one is the dc servomotor used for rotating the direction. An FPGA chip, several circuit boards, and six infrared sensors are mounted on this vehicle. The appearance of the actual autonomous CLMR is illustrated in Fig. 9.

The FPGA chip of the experiment board is the EPF6024ACT144-3 that is the Flex Series of Altera chip and is manufactured by Galaxy Far East Corporation. This chip can provide 24 000 gates and 117 I/O ports. We apply the MAXPLUS II developed by Altera Company Ltd. to design and analyze the integrated circuits. The VHDL is programmed on the PC and then downloaded to the FPGA chip via a parallel port.

When the CLMR travels in the working environment, it will receive the data of the environment via six infrared sensors. The



(e) (f)



(g)

Fig. 6. *(Continued.)* Simulation results of all the proposed control schemes. (e) Simulation result of the FGPC with initial posture $(x_f, y_f, \theta) = (650\text{o}, 175, \text{o}\, 0^\circ)$. (f) Simulation result of the FPPC with initial posture $(x_f, y_f, \theta) = (675\text{o}, 210\text{o}, 0^\circ)$. (g) Simulation result of the FPPC with initial posture $(x_f, y_f, \theta) = (675\text{o}, 210\text{o}, -20^\circ)$.

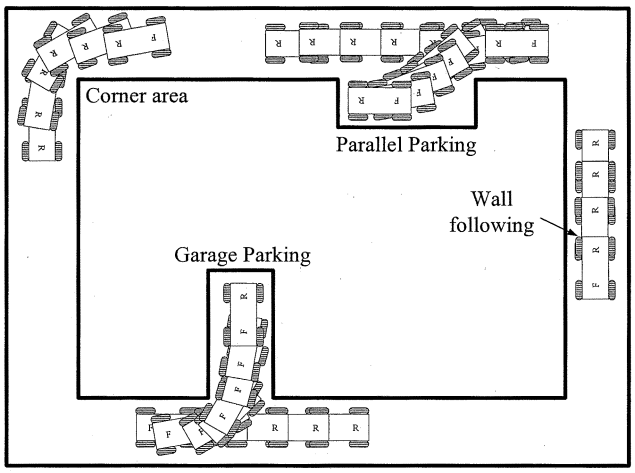


Fig. 7. Diagram of the test ground.

UF 66 MG manufactured by TELCO International Ltd. is the infrared sensor and contains both a transmitter and a receiver. This sensor gives the output voltage proportional to the reflection distance. The measurement distance is 1 m. There are six infrared sensors mounted on the CLMR. Four infrared sensors are mounted above the positions of the right wheels and left wheels of the CLMR, respectively. To avoid collisions with obstacles, we place one infrared sensor in front of the robot as well as one in the rear. By using six infrared sensors, the information of distances (SF, SR, SR_1, SR_2, SL_1, and SL_2) can be determined.

The A/D converter (ADC0804) transforms the data into digital signals. The controller implemented in the FPGA chip reads the digital data once the conversion of the A/D converter has been done and generates two control commands. One command is the pulsewidth-modulation (PWM) signal to the dc servomotor for controlling the steering angle of the CLMR. The other

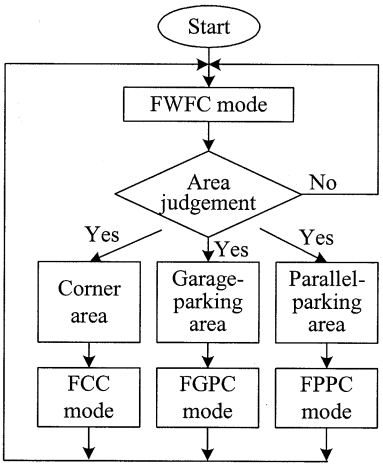


Fig. 8. Flowchart of the behavior selection scheme.

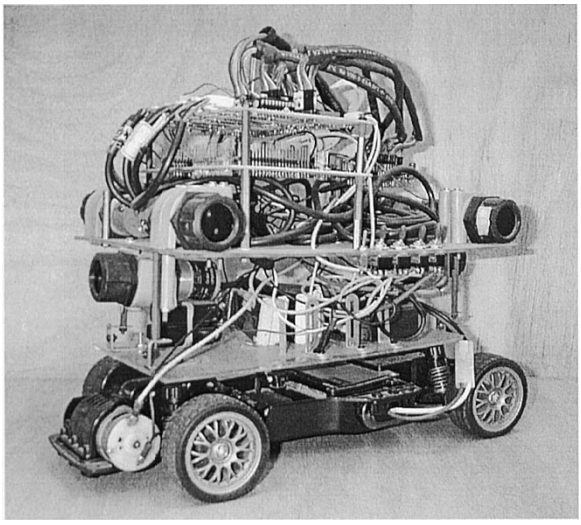


Fig. 9. Actual appearance of the autonomous CLMR.

one will be sent to the dc motor for controlling the speed of the CLMR through the motor driver IC (TA7291P) and D/A converter (DAC0800). All power used in the hardware of the CLMR will be converted from batteries by using a dc/dc converter (LM317 and LM7805).

IV. Circuit Design on FPGA Chip

After establishing the hardware architecture of the CLMR, we begin to design the AFBC on the FPGA chip to achieve the autonomous fuzzy parking control. The hardware circuit architecture of the AFBC realized on the FPGA chip is shown in Fig. 10(a). The entire I/O port for this chip includes 49 pins for the input port and 12 pins for the output port. By the input port, there are eight pins from each infrared sensor via A/D converter and one pin from the "CK" of the quartz oscillator. By the output port, there are two pins for the dc motor driver IC, one pin for the A/D converter, one pin for the dc servomotor, and eight pins for the data input of the D/A converter. The AFBC contains several modules, including the behavior selection control module, the clock divider module, A/D converter control module, the PWM fitness module, FLC module, etc. The modules of the AFBC are described as follows.

A. Behavior Selection Control Module

The behavior selection control module has three behaviors, corner behavior, garage-parking behavior, and parallel-parking behavior. The mobile robot determines which behavior to move on by the rule of (3). Fig. 10(b) shows the architecture of the behavior switching control module.

B. Clock Divider Module

The clock divider module is mainly used to generate the specified clock frequency to trigger the A/D converter and control the dc servomotor. A 1-MHz quartz oscillator is used here to supply the clock frequency of the FPGA chip. We use the clock divider module to reduce 1 MHz to the specified clock frequency. For example, the CLMR only runs about 500 Hz. It may result in unstable vibration of the dc servomotor if a high working clock frequency is utilized. The clock divider module is composed of an up-counter and a D flip-flop. We exploit the up-counter to count a double clock frequency, and then we can get the specified clock frequency after the D flip-flop.

C. A/D Converter Control Module

Because the infrared sensor is sensitive to the environment, we need to filter out the transient noise by a digital filter. To avoid the transient noise, the digital filter consists of a SR flip-flop and three D flip-flops to synthesize the noise-elimination circuit. The A/D converter control module consists of an up-counter, a digital filter, and a delay circuit. The up-counter generates the "WR" signal to start the A/D converter, then the analog signals of the six infrared sensors are converted to six 8-bit digital signals. After the digital filter, the distance information is figured out, and then the module produces the enable clock of the behavior selection control module to read these signals. The delay circuit is mainly used to determine how much time the CLMR should stay at the parking lot.

D. PWM Fitness Module

The PWM signal is provided to control the steering angle by controlling the dc servomotor. In digital circuit design, we utilize an up-counter whose output value is compared with a reference value. If the counted value is less than the reference value, then the output of the comparator exports "0;" otherwise, "1" is exported. We just need to set the reference value to choose a proper PWM signal.

Owing to that, the range of steering angle is $-30°$ to $+30°$, we take $5°$ as one unit to divide the range into 13 types of PWM signals. When the PWM fitness circuit gets the crisp output from the FLC module, we use the multiplexer as the PWM selector to match the corresponding crisp output, and then the steering angle of the CLMR can be controlled. Fig. 10(c) illustrates the configuration of the PWM fitness module.

E. Design of FLC Module for Steering Angle Control

The hardware architecture of the FLC module is also implemented on the FPGA chip. The hardware architecture in-

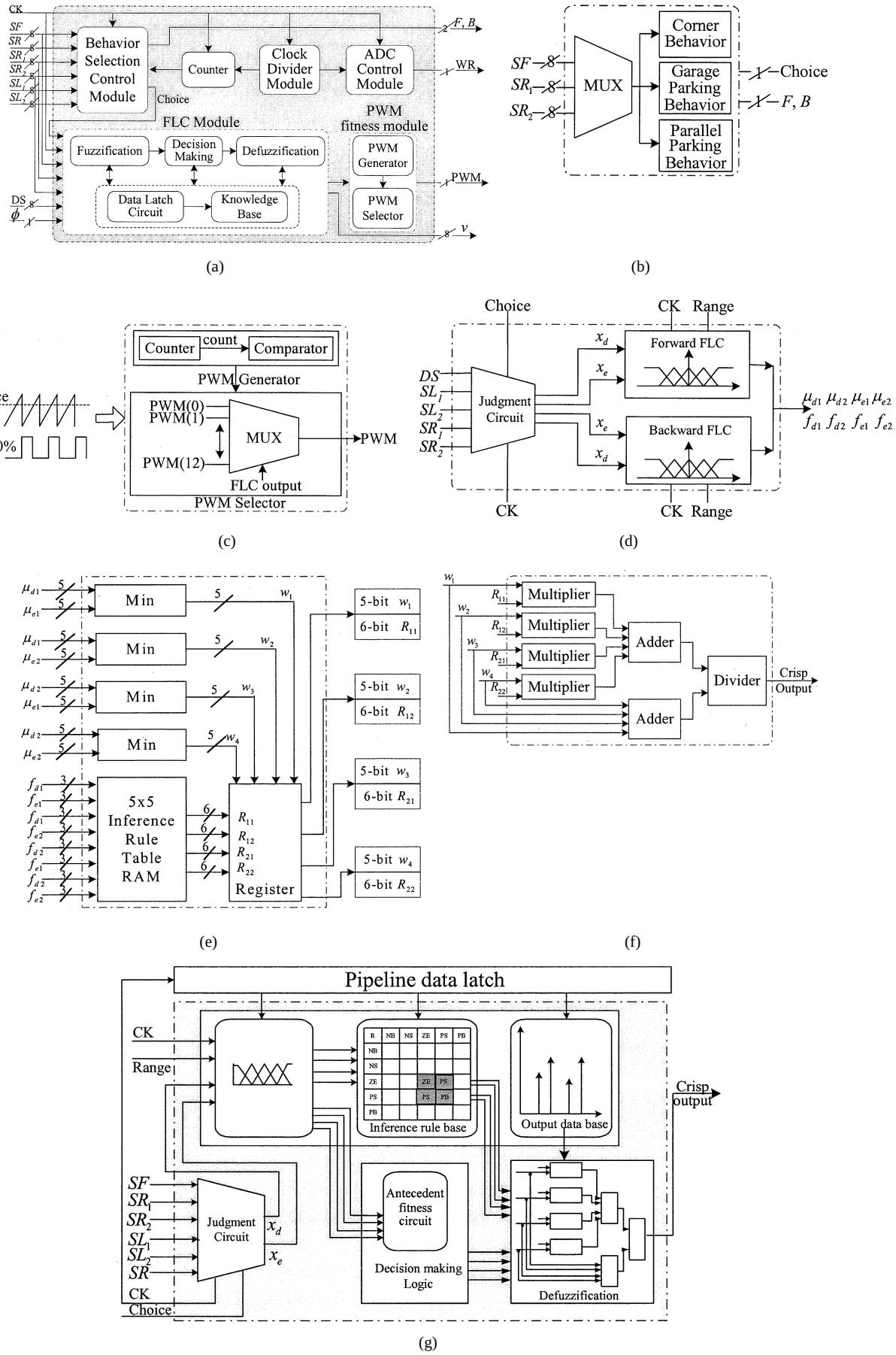


Fig. 10. (a) Hardware circuit architecture of the AFBC. (b) Architecture of the behavior selection control module. (c) PWM fitness module. (d) Fuzzification submodule. (e) DML submodule. (f) Circuit architecture of the defuzzification submodule. (g) Implementation circuit of the FLC module.

cludes fuzzification submodule, decision-making logic (DML) submodule, and defuzzification submodule. We design one FLC module for steering angle control and one FLC module for speed control, respectively. Each submodule of the FLC module for steering angle control will be described in detail in the following paragraphs.

The inputs of the FLC module are totally 26 bits, where two 8-bit inputs are the digital values from A/D converter measured by SR_1 and SR_2 sensors, one 8-bit input is the desired distance value between the wall and the CLMR, one bit is used to trigger each submodule of the FLC module for the synchronous clock, and one bit is the discriminated variable representing forward or backward movement of the CLMR. The outputs of the FLC module have 4 bits for the PWM signal, which are used for the steering angle control.

1) Fuzzification Submodule: As the FLC module catches the data measured by the infrared sensors, the data have to be transmitted into the input of the FLC module. The architecture of the fuzzification submodule is illustrated in Fig. 10(d). We design a judgment circuit to determine which statuses the input variables of the FLC scheme will be. The "Choice" signal from the behavior selection control module is used to decide that the CLMR should go forward or move backward. When "Choice" signal is low, the CLMR will go forward and use the forward FWFC and, on the contrary, the CLMR will move backward and adopt the backward FWFC.

We utilize the outside A/D converter to convert the analog voltage values measured by infrared sensors into digital data. The digital input variables x_d and x_e are chosen by the judgment circuit. Then, the digital data streams are mapped to fit the universes of discourses.

2) DML Submodule: In this stage, the antecedent fitness block and inference rule-base block are utilized to implement the DML submodule. After the fuzzification process, the fuzzification submodule will export the excited membership values and the excited fuzzy linguistic labels to the DML submodule for fuzzy inference operations. Fig. 10(e) depicts the design of the DML submodule.

Because a minimum operator is considered for realization of the composition operation of the FLC in the antecedent fitness block, the "Min" circuit is implemented to compare the minimum value. Four excited membership values μ_{d1}, μ_{d2}, μ_{e1}, and μ_{e2} from the fuzzification submodule are exported to four 2-to-1 "Min" circuits in order to obtain the antecedent values of the fuzzy rules.

In the FLC module, the antecedent and the consequence of these IF–THEN rules are associated with fuzzy conditional statements. The design concept of the inference rule base block is realized in the lookup table. The block will export the excited rules and the linguistic description of the rules can be expressed as the following rules.

R_{11}: If f_{d1} is ZE and f_{e1} is ZE, then R is ZE.
R_{12}: If f_{d1} is ZE and f_{e2} is PS, then R is PS.
R_{21}: If f_{d2} is PS and f_{e1} is ZE, then R is PS.
R_{22}: If f_{d2} is PS and f_{e2} is PS, then R is PB.

This DML block will output four control weighted values (antecedent degree) and four labels linguistic terms of the triggered rules to the next defuzzification submodule.

3) Defuzzification Submodule: The defuzzification method is based on weighted average method mentioned in previous sections. The design concept of the defuzzification submodule is to figure out one crisp output based on excited four rules. Thus, the blocks of defuzzification submodule implemented here include four multipliers, several adders, and one division divider. The circuit architecture of the defuzzification submodule is depicted in Fig. 10(f), where

$$Crisp\ output = \frac{w_1 * R_{11} + w_2 * R_{12} + w_3 * R_{21} + w_4 * R_{22}}{w_1 + w_2 + w_3 + w_4}. \tag{4}$$

Each circuit of the defuzzification submodule will be briefly described as follows. To speed up the calculation process of the multiplier, we adopt the parallel-type multiplier, which may take more the volume. The divider is the last circuit of the defuzzification submodule. It is composed of shifter, dividend register, divisor register, remainder register, quotient register, subtractor register, and logic circuit. The operating action of the divider is just a sequential divider circuit. The defuzzification submodule will be implemented by integrating all circuits after realizing the divider.

4) Latch Circuit: It is known that suppose the delay time and/or the sequence of transmission for each submodule circuit is different, then unstable output signal and/or unexpected results may occur. The latch circuit is in fact set up at the ends of fuzzification, DML, and defuzzification submodules. Its objective is that the input signals of the present stage will not vary with the outputs changes of the former stage submodule. The implementation of the digital FLC module is shown in Fig. 10(g), where all the submodules and the pipeline data latch architecture are included.

F. Design Circuit of FLC Module for Speed Control

Because the hardware architecture of the FLC for speed control is the same as that for steering angle control, the entire design procedure for the speed control process is the same as that in the previous sections and, hence, the description is omitted here.

V. Real-Time Experiments

The computer simulation results discussed in Section II show that the AFBC can successfully execute the wall-following, corner-turning, garage-parking, and parallel-parking behaviors. The hardware architecture of the CLMR and the digital circuit design on the FPGA chip have been respectively established in Sections III and IV. In this section, we want to realize the AFBC on the CLMR in a real test ground. The dimensions of the investigated CLMR are: length, 380 mm; width, 240 mm; and weight, 5 kg. The shape of the test ground depicted in Fig. 7 is rectangular and the dimensions of the real test ground are about 380 cm in length and 285 cm in width, where one garage, one parallel-parking lot, and four corners are included.

The actual experimental photographs of the AFBC on the CLMR in the test ground are shown in Fig. 11, where 72 sequential image stills captured by the handheld charge-coupled device (CCD) camera are given. We decompose all these pictures into six parts. In image stills 1–9, the CLMR begins to perform the maneuvering behavior of the upper left corner by

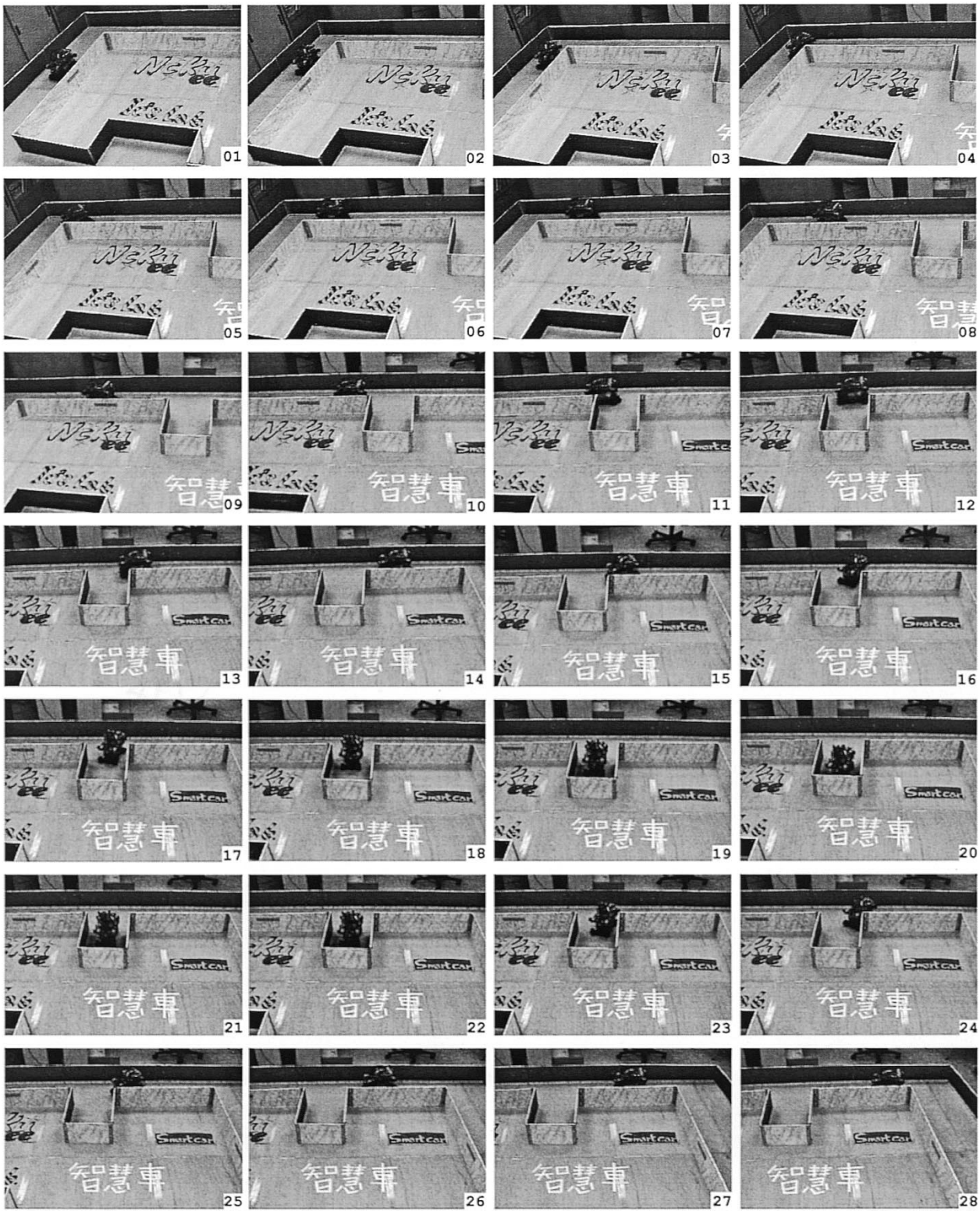


Fig. 11. Seventy-two sequential image stills (top left to bottom right, captured by handheld CCD camera) for real experiments of the AFBC on the CLMR in the test ground. Image stills 1–9: experimental results of the FCC at the upper left corner; image stills 10–21: experimental results of the FGPC; image stills 22–33: experimental results of the maneuvers for the departure of the garage and the upper right corner; image stills 34–42: experimental results of the FCC at the lower right corner; image stills 43–57: experimental results of the FPPC; image stills 58–72: experimental results of the maneuvers for the departure of the parking lot and the lower left corner.

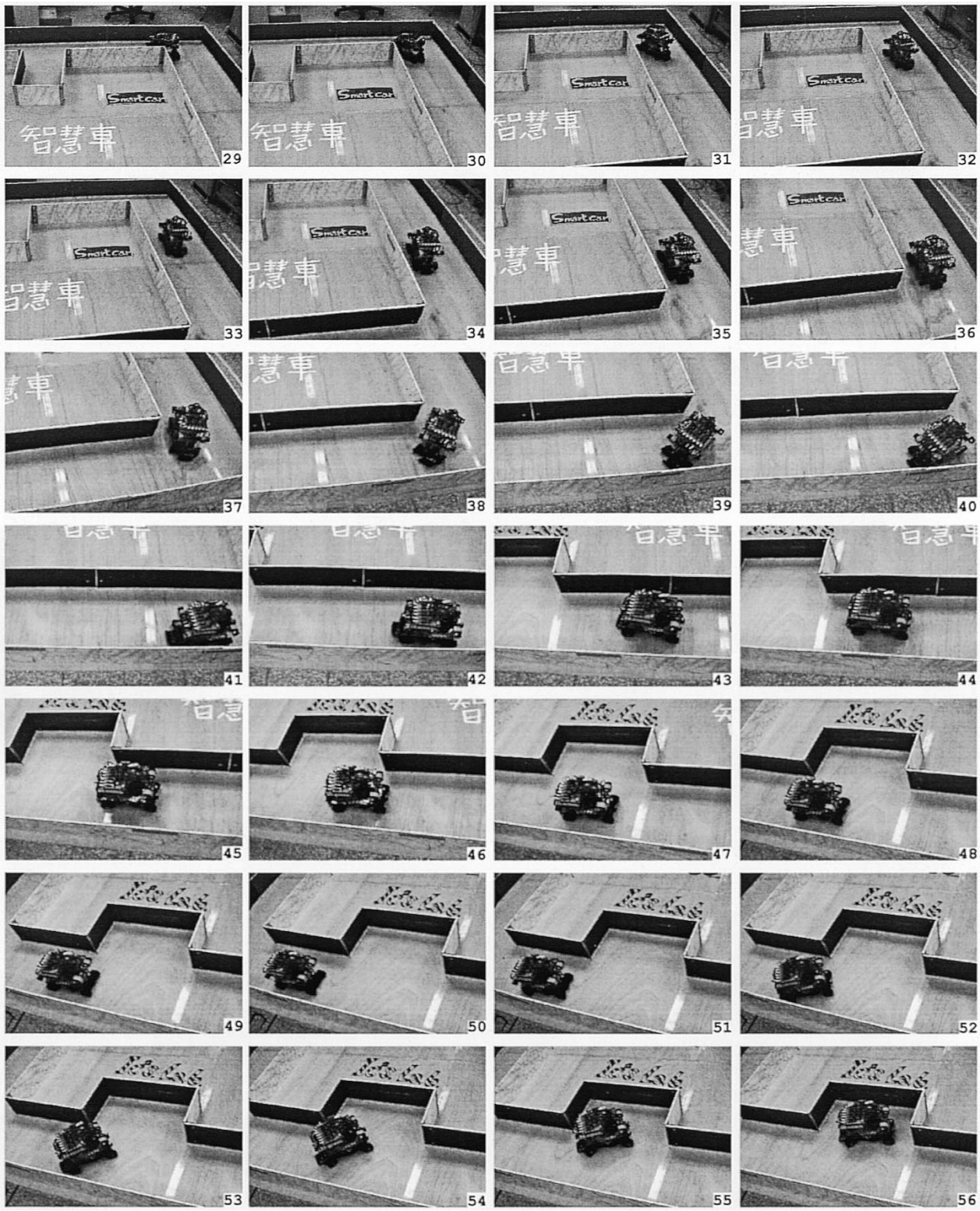


Fig. 11. *(Continued.)* Seventy-two sequential image stills (top left to bottom right, captured by handheld CCD camera) for real experiments of the AFBC on the CLMR in the test ground. Image stills 1–9: experimental results of the FCC at the upper left corner; image stills 10–21: experimental results of the FGPC; image stills 22–33: experimental results of the maneuvers for the departure of the garage and the upper right corner; image stills 34–42: experimental results of the FCC at the lower right corner; image stills 43–57: experimental results of the FPPC; image stills 58–72: experimental results of the maneuvers for the departure of the parking lot and the lower left corner.

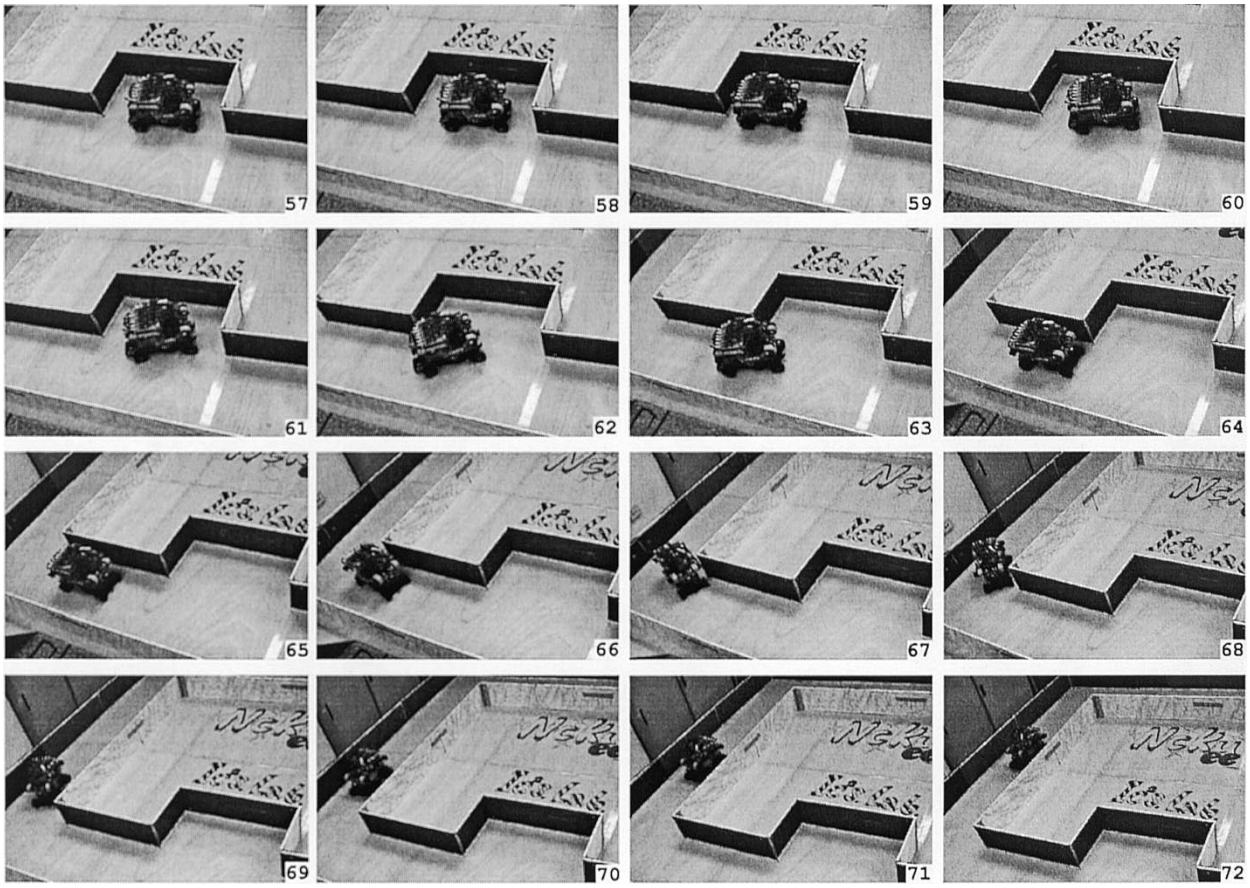


Fig. 11. *(Continued.)* Seventy-two sequential image stills (top left to bottom right, captured by handheld CCD camera) for real experiments of the AFBC on the CLMR in the test ground. Image stills 1–9: experimental results of the FCC at the upper left corner; image stills 10–21: experimental results of the FGPC; image stills 22–33: experimental results of the maneuvers for the departure of the garage and the upper right corner; image stills 34–42: experimental results of the FCC at the lower right corner; image stills 43–57: experimental results of the FPPC; image stills 58–72: experimental results of the maneuvers for the departure of the parking lot and the lower left corner.

using the FCC. Image stills 10–21 give the experimental results of the FGPC. In image stills 22–33, the CLMR first drives off from the garage and then makes a right turn at the upper right corner. Image stills 34–42 show the corner-turning capability at the lower right corner. Image stills 43–57 are the real experiments on the FPPC. In image stills 58–72, we first drive the CLMR off the parking lot and execute the corner-turning behavior of the lower right corner. One can find from these pictures that the proposed AFBC can successfully complete all these tasks.

We want to emphasize here that once the CLMR is turned on, it has the capability to autonomously maneuver in the test ground. In other words, the CLMR itself can determine which behavior should be executed according to the AFBC. In the present test ground, the CLMR will continuously perform the corner turning at the upper left corner, the garage parking, the corner turning at the upper right corner, the corner turning at the lower right corner, the parallel parking, and the corner turning at the lower left corner until we turn off the power of the CLMR.

VI. Conclusion

In this paper, an AFBC method has been presented to emulate human-like driving skills. This method is in fact developed by our driving experiences, FLC techniques, and sensor-based behaviors. It has been synthesized by the FWFC, FCC, FGPC, and FPPC. We have also designed and implemented an autonomous CLMR, where chassis mechanism, motor driver, FPGA experiment board, A/D and D/A converter, infrared sensor, and power regulator have been set up. The AFBC scheme has been realized by the FPGA chip that is set up on the CLMR. Design procedure of the AFBC on the FPGA chip has been clearly provided. All computer simulations and real-time experiments demonstrate that the propounded AFBC is indeed effective and feasible for practical application to real car maneuvers.

Acknowledgment

The authors wish to thank the anonymous reviewers for their valuable suggestions.

References

[1] M. Sugeno and K. Murakami, "An experimental study on fuzzy parking control using a model car," in *Industrial Applications of Fuzzy Control*, M. Sugeno, Ed. Amsterdam, The Netherlands: North-Holland, 1985, pp. 105–124.

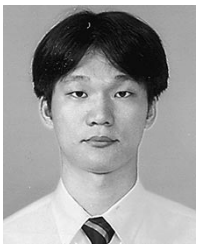
[2] M. Sugeno, T. Murofushi, T. Mori, T. Tatematsu, and J. Tanaka, "Fuzzy algorithmic control of a model car by oral instructions," *Fuzzy Sets Syst.*, vol. 32, pp. 207–219, 1989.

[3] M. Ohkita, H. Mitita, M. Miura, and H. Kuono, "Traveling experiment of an autonomous mobile robot for a flush parking," in *Proc. 2nd IEEE Conf. Fuzzy System*, vol. 2, San Francisco, CA, 1993, pp. 327–332.

[4] S. Yasunobu and Y. Murai, "Parking control based on predictive fuzzy control," in *Proc. IEEE Int. Conf. Fuzzy Systems*, vol. 2, 1994, pp. 1338–1341.
[5] W. A. Daxwanger and G. K. Schmidt, "Skill-based visual parking control using neural and fuzzy networks," in *IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 2, 1995, pp. 1659–1664.
[6] D. Leitch and P. J. Probert, "New techniques for genetic development of a class of fuzzy controllers," *IEEE Trans. Syst., Man, Cybern.*, vol. 28, pp. 112–123, Feb. 1998.
[7] M. C. Leu and T. Q. Kim, "Cell mapping based fuzzy control of car parking," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1998, pp. 2494–2499.
[8] K. Y. Lian, C. S. Chin, and T. S. Chiang, "Parallel parking a car-like robot using fuzzy gain scheduling," in *Proc. 1999 IEEE Int. Conf. Control Applications*, vol. 2, 1999, pp. 1686–1691.
[9] F. Gomez-Bravo, F. Cuesta, and A. Ollero, "Parallel and diagonal parking in nonholonomic autonomous vehicles," *Eng. Applicat. Artif. Intell.*, vol. 14, pp. 419–434, 2001.
[10] B. Shirazi and S. Yih, "Learning to control: a heterogeneous approach," in *Proc. IEEE Int. Symp. Intelligent Control*, 1989, pp. 320–325.
[11] R. M. Murray and S. S. Sastry, "Nonholonomic motion planning: steering using sinusoids," *IEEE Trans. Automat. Contr.*, vol. 38, pp. 700–716, May 1993.
[12] J. P. Laumond, P. E. Jacobs, M. Taix, and R. M. Murray, "A motion planner for nonholonomic mobile robots," *IEEE Trans. Robot. Automat.*, vol. 10, pp. 577–593, Oct. 1994.
[13] I. E. Paromtchik and C. Laugire, "Motion generation and control for parking an autonomous vehicle," in *Proc. 1996 IEEE Conf. Robotics and Automation*, vol. 4, Minneapolis, MN, 1996, pp. 3117–3122.
[14] C. Laugier, T. Fraichard, I. E. Paromtchik, and P. Garnier, "Sensor-based control architecture for a car-like vehicle," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, 1998, pp. 216–222.
[15] K. Jiang and L. D. Seneviratne, "A sensor guided autonomous parking system for nonholonomic mobile robots," in *Proc. 1999 IEEE Int. Conf. Robotics and Automation*, vol. 1, 1999, pp. 311–316.
[16] K. Jiang, "A sensor guided parallel parking system for nonholonomic vehicles," in *Proc. IEEE Conf. Intelligent Transportation Systems*, 2000, pp. 270–275.
[17] H. Watanabe and W. D. Dettloff, "VLSI fuzzy chip and inference accelerator board systems," in *Proc. IEEE 21st Int. Symp. Multiple-Valued Logic*, 1991, pp. 120–127.
[18] F. G. Pin, H. Watanabe, J. Symon, and R. S. Pattay, "Using custom-designed VLSI fuzzy inferencing chips for the autonomous navigation of a mobile robot," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, 1992, pp. 790–795.
[19] F. G. Pin and Y. Watanabe, "Using fuzzy behaviors for the outdoor navigation of a car with low-resolution sensors," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1993, pp. 548–553.
[20] A. Kongmunvattana and P. Congstitvatana, "A FPGA-based behavioral control system for a mobile robot," in *Proc. IEEE Asia-Pacific Conf. Circuits and Systems*, 1998, pp. 759–762.
[21] D. Kim, "An implementation of fuzzy logic controller on the reconfigurable FPGA system," *IEEE Trans. Ind. Electron.*, vol. 47, pp. 703–715, June 2000.
[22] J. L. Arroyabe and G. Aranguren, "Autonomous vehicle guidance with fuzzy algorithm," in *Proc. IEEE IECON*, vol. 3, 2000, pp. 1503–1508.

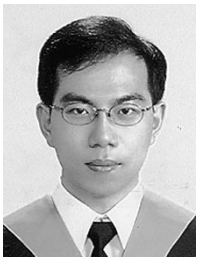
Tzuu-Hseng S. Li (S'85–M'90) received the B.S. degree from Ta-Tung Institute of Technology, Taipei, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees from National Cheng Kung University, Tainan, Taiwan, R.O.C., in 1985 and 1989, respectively, all in electrical engineering.

He has been a Professor in the Department of Electrical Engineering, National Cheng Kung University, since 1994. He has served as a Researcher for the Engineering and Technology Promotion Center of the National Science Council since 1996. From 1999 to 2002, he served as the Director of Electrical Laboratories, National Cheng Kung University. His current research interests include singular perturbation, intelligent control ICs and systems, mechatronics, mobile robots, RoboCup, and automobile active suspension systems.



Shih-Jie Chang received the B.S. degree from Chung Yuan Christian University, Chung-Li, Taiwan, R.O.C., in 1995, and the M.S. and Ph.D. degrees from National Cheng Kung University, Tainan, Taiwan, R.O.C., in 1997 and 2003, respectively, all in electrical engineering.

In October 2003, he joined Acer Inc., Taoyuan, Taiwan, R.O.C., as an R&D Engineer. His current research interests are fuzzy logic control, intelligent control, and mobile robots.



Yi-Xiang Chen received the B.S. and M.S. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, R.O.C., in 2000 and 2002, respectively.

He is currently an R&D engineer with Primax Electronics Ltd., Taipei, Taiwan, R.O.C. His current research interests are fuzzy logic control and mobile robots.