

2012 AASRI Conference on Modelling, Identification and Control

Monocular Vision Navigation and Control of Mobile Robot

Runchen Yan^{a,*}, Hong Wang^a, Yuzhi Yang^b, Huanbing Wei^b and Yonggang Wang^b

^aState Key Laboratory of Intelligent Technology and Systems

Tsinghua National Laboratory for Information Science and Technology

Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

^bThe Hongfeng Co., Hubei, Xiaogan, 432000. China

Abstract

In this paper, we implement a monocular vision navigation system on a mobile robot which can travel in known outdoor environment. The system uses 5 different sensors on the robot which are monocular color camera, laser radar, sonar, gyroscopes and optical encoder. And the robot can cruise independently in known continuous structured and unstructured road environment mainly through the vision navigation module. There are road recognition module, signs recognition module, stop line recognition module, zebra crossing recognition module, obstacle sensing module, control module and communication module. The system utilizes multi-threading technology to collaborate different modules in the same control cycle and automatic state machine to switch different control modes. Experimental results show that the robot can accomplish autonomous cruise task on the predetermined route and operate stably when it is in a good lighting conditions.

© 2012 The Authors. Published by Elsevier B.V. Open access under [CC BY-NC-ND license](#).
Selection and/or peer review under responsibility of American Applied Science Research Institute

Keywords: vision navigation, mobile robot, monocular

* Corresponding author. Tel.: +86-13141259359.
E-mail address: chen_0_1987@sina.com.

1. Introduction

Compared to intelligent vehicles, small intelligent mobile robotics has many different characteristics, such as low image stability, low control precision, different ways of turning, narrow angle of shooting due to low camera height. Therefore, it needs higher accuracy of environmental awareness, collaboration of multi-sensor and self-control technology. In application of environment exploring and cruise task, small intelligent mobile robot can play a better role.

In this paper, the autonomous vision navigation system is developed on the small mobile robot platform, Adept MobileRobots PT 3-AT. Our vision navigation robot is designed for known outdoor routes. It can travel along the edges of the unstructured roads and the lane lines of the structured roads. It will turn when recognizing a special sign or zebra crossing. Section 2 introduces the design framework of the whole system. Section 3 gives a brief introduction of vision recognition module. Section 4 elaborates the control module and strategy. Experiment results are given in section 5. And the conclusion and future work are discussed in section 6.



Fig. 1. Adept MobileRobots PT 3-AT

2. Design and Framework of Whole System

The whole system is divided into two parts, the upper machine client and lower machine host. In the paper it mainly introduces the system in lower machine host.

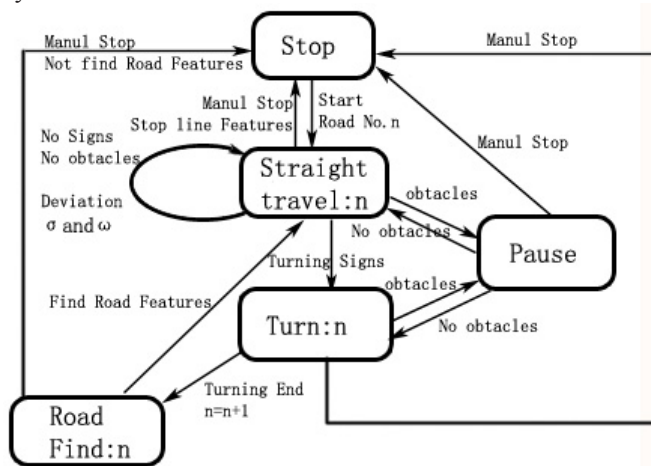


Fig. 2. The state transition simplified diagram of the FSM system

In the system, there are many states which compose a finite state machine (FSM)[1]. In state of straight traveling, turning and road finding, the system uses different well-designed strategies and rules to control the robot behavior based according to the road number. Manual control mode is automatically entered when system go into the stop state. In that state, people can send commands to control the robot how to walk through the upper machine client.

3. Vision Recognition Module

We designed two different road-edge detection algorithms for unstructured road[2]. One is color-based threshold segmentation algorithm, to deal with the situation of high color contrast between road and non-road area. The other is a grayscale-based extension algorithm of Otsu[3], to deal with the situation that the road and non-road area boundaries are not obvious. We designed another color-based threshold segmentation algorithm for structured road, to recognize the lane lines. After recognizing correct traveling path and calculating by inverse affine transformation, the algorithm above will give final results, the distance deviation and the angle deviation at the current position.

The recognition algorithms for stop lines and zebra crossings are similar to lane detection algorithm. They are used to control the behavior of the robot stopped and specific sections turn.

In order to improve the accuracy of the turning and adapt to the complex environment, we specifically designed a high-contrast sign for our vision navigation system, which is composed of yellow and black. The signs are placed in fixed positions of the turning area and maintain the same height to mark the accurate turning point. By using the HSV color threshold segmentation, edge detection, RANSAC[4] ellipse fitting and misrecognition filtering, the recognition algorithm can identify the turning signs outdoors. And it will give a turning signal to system when the detected signs in images reach a predetermined size. Then the system will go into the turning state from a straight traveling state at the predetermined turning points.

4. Control Module and Strategies

Since only using monocular camera, the vision algorithms may make slight deviations when estimating distance. It will lead continuous adjusted in corrective process of straight travelling. The robot outdoors walking and turning in differential way will make great influence on image stability because of the friction between wheels and ground. Moreover, false detection of vision recognition algorithm often appear in complex outdoor environment. All these factors present challenges of robot control strategy design to us.

4.1 The definition of distance deviation and angle deviation

The robot has its own global coordinate system. When the robot starts, it is on the origin of coordinates and face to the x-axis positive direction, 90 degrees counterclockwise is the y-axis positive direction. The range of angle is 0 to 180 degrees in the first and the second quadrant and -180 to 0 degrees in the third and fourth quadrant. In short, in the positive x direction, it is 0 degree. Clockwise is negative and counterclockwise is positive. As the same way, the angle deviation, which is defined as the angle between correct direction and current direction, will in range of 0 to 180 degrees clockwise, and range of -180 to 0 degrees counterclockwise.

The distance deviation is defined as vertical distance between the center point of the robot and the correct path line. When facing the correct direction, the distance deviation of the robot in the correct path line is 0. The left side is positive and the right side is negative.

1.1. The strategies of straight travelling and correction

In the design, we introduce a fuzzy controller to suppress the excessive response of the system to complex noise and keep the stability and accuracy of traveling straight. Fuzzy control is a control system to control the complex system which is difficult to be described by accurate mathematical model [5].

We firstly designed a group of simple linear control equations.

$$\begin{cases} \omega = \frac{-\alpha \times \theta - \beta \times \delta}{\gamma} \\ v = v_{max} - |a \times \theta| - |b \times \delta| \end{cases} \quad (1)$$

Where ω is angular velocity (degrees/s), v is linear velocity (mm/s), δ is distance deviation, θ is angle deviation, $v_{max} = 1000$, α , β , γ , a , b are adjustable parameters.

Of course, the equations above are far to meet demand of stable controlling. So we improve them in the experiments. Linear velocity remains linear. Angular velocity is controlled by a rule-based fuzzy controller in only part of the situations. Fuzzy controller structure is shown as follows:

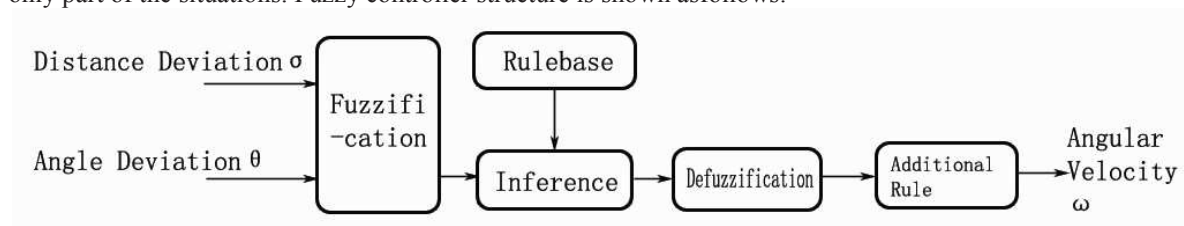


Fig. 3. The structure of fuzzy controller

We obtain the distance deviation and angle deviation advantage from the vision algorithms and pose estimation, which are referred in chapter 2. And the two parameters are used as the input fuzzy parameters. The fuzzy sets of them both are {NB, NM, NS, PS, PM, PB}.

The figure below presents the fuzzy membership of distance deviation. Horizontal axis represents the distance deviation of robot (mm). And the vertical axis represents the value of membership.

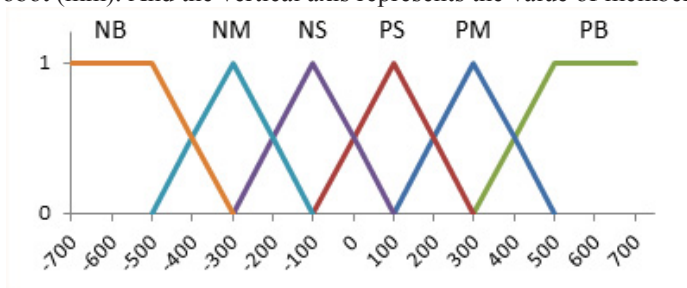


Fig. 4. The fuzzy membership of distance deviation

The fuzzy membership of angle deviation is shown below. Horizontal axis represents the angle deviation of robot (degrees). And the vertical axis represents the value of membership.

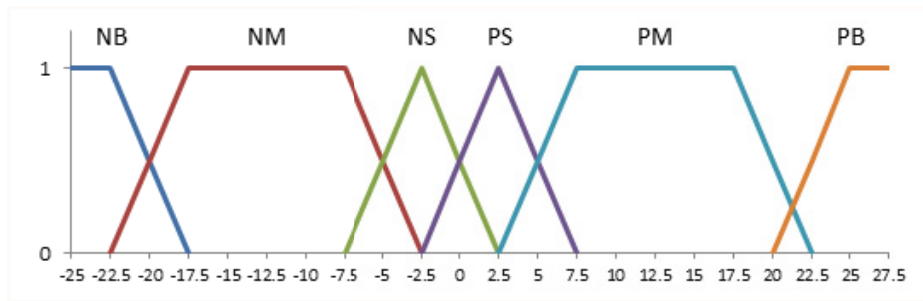


Fig. 5. The fuzzy membership of angle deviation

The fuzzy membership of angular velocity is shown as following. Horizontal axis represents the angular velocity of robot (degrees/s). The vertical axis represents the value of membership. When angular velocity is greater than 11 degrees/s or less than -11 degrees/s, the value of fuzzy membership is 0. These cases are out of the range of the fuzzy controller. Because of robot control accuracy itself, it can just utilize the discrete formulation for defuzzification calculation.

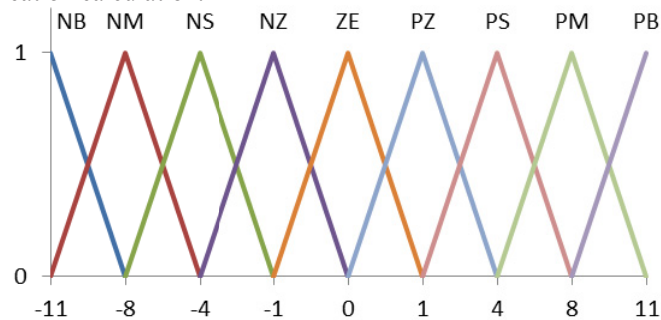


Fig. 6. The fuzzy membership of angular velocity

Considering the robot travel behaviors features when human controlling, we summarized the following fuzzy control rules. The fuzzy inference synthesis rule is maximum - minimum rule. And defuzzification directly utilizes the max criterion method (MC)[6]. The blank means that angular velocity is calculated by using the original linear equation in these situations.

Table 1. fuzzy control rules

θ	δ					
	NB	NM	NS	PS	PM	PB
NB				PM	PM	
NM			PB	PS	PS	
NS		PS	ZE	ZE	PZ	
PS		NZ	ZE	ZE	NS	
PM		NS	NS	NB		
PB		NM	NM			

After defuzzification, we add the following additional rule: $(|\theta| > \theta_m \text{ AND } \omega \times \theta > 0) \rightarrow \omega = 0$. This rule can limit angular velocity when correction of distance deviation, in order to eliminate the excessive rotation phenomenon. In experiments, we set $\theta_m = 10$.

1.2. Inertial navigation correction assistance

The vision navigation module requires certain conditions so that it can work well. However, only using the inertial navigation sensor will cause accumulated errors. So for stable travelling, it is essential that using optical encoder and gyroscope to assist the vision navigation.

As we said above, when optical encoder and gyroscope is on, our system has a global coordinate system. Through the vision information in the past and the location, orientation of robot at that time, we can calculate the equation of correct path line in global coordinate system.

According to the vision data, angle deviation θ which is accepted by system, and the orientation angle th in coordinate system at that time, we can obtain the slope of the correct path line: $k = \tan(th - \theta)$. And we can also get the angle between correct orientation and the x-axis positive direction:

$$finalth = \begin{cases} th - \theta - 360 & (th - \theta > 180) \\ th - \theta & (-180 \leq th - \theta \leq 180) \\ th - \theta + 360 & (th - \theta < -180) \end{cases} \quad (2)$$

According to the coordinates (x_1, y_1) at that time and distance deviation δ which is provided by vision navigation, we can figure out the coordinates of a point on correct path line:

$$\begin{cases} x_0 = x_1 + \delta \times \sin(finalth) \\ y_0 = y_1 - \delta \times \cos(finalth) \end{cases} \quad (3)$$

Then, we obtain the equation of the line as below:

$$kx - y - kx_0 + y_0 = 0 \quad (4)$$

According to current coordinates (x_n, y_n) , we can use the equation above to get the new distance deviation δ_n :

$$\delta_n = \begin{cases} \frac{-(kx_n - y_n - kx_0 + y_0)}{\sqrt{k^2 + 1}} & (-90 < finalth < 90) \\ \frac{kx_n - y_n - kx_0 + y_0}{\sqrt{k^2 + 1}} & (finalth \geq 90, finalth \leq -90) \end{cases} \quad (5)$$

There are two special cases: When $finalth = 90$, $\delta_n = x_1 + \delta - x_n$. When $finalth = -90$, $\delta_n = x_n - x_1 + \delta$.

The new angle deviation θ_n can be calculated according to $finalth$ which represents the angle between correct path line and the x-axis. Here $nowth$ represents the current angle of robot in global coordinate system.

$$\theta_n = \begin{cases} nowth - finalth - 360 & (nowth - finalth > 180) \\ nowth - finalth & (-180 \leq nowth - finalth \leq 180) \\ nowth - finalth + 360 & (nowth - finalth < -180) \end{cases} \quad (6)$$

The calculated results, θ_n and δ_n will be put into the control strategy as input parameters to control straight travelling of robot.

In every cycle, the system will compare the deviation data from inertial navigation with the one from vision navigation. If the difference between the two data is obvious, that is to say, distance deviation greater than 800mm or angle deviation more than 17 degrees with distance deviation less than 343mm, the system will accept the data from inertial navigation; otherwise use the vision navigation data. The above-mentioned values can be adjusted.

1.3. Turning, path-finding and collision avoidance strategies

Because of vibration caused by friction when the robot is turning, the image recognition algorithms tend to result more false detections or misses. We do not focus on designing a precise solution to turning recognition and controlling. So we just use the signs which referred in Chapter 3 to mark the fixed turning point. Then we set turning angle and distance before and after turning in every turn, which are estimated from experiments.

The path-finding strategy is applied after turning, at the beginning at each road. Every time after turning, the robot will search along the current direction in an ± 45 degrees angle range. The correct direction should deviate from the current direction within a certain angular.

The laser radar and sonar sensors in the system are only for detecting an obstacle and avoiding collision by sending stop command when an obstacle is detected in front of the robot within ± 45 degrees angular range as well as the distance between the obstacle and the robot is less than preset safe distance.

5. Experiment results

We used the robot simulator to do the ideal test which compared the control effect before and after adding fuzzy controller. The distance deviation and angle deviation are given completely correct for each control cycle and start correction at same distance deviation. From the ideal result, we can find a decrease in the oscillation of correction and overshooting. The whole correction process became relatively steady. In the practical experiments, these advantages play a very important role to enhance the stability and accuracy and prevent the robot from significant S-shape walking.



Fig. 7. (a) linear control in correction; (b) After add fuzzy controller in correction

We hold a two-month practical adjustment and experiment on 6 predetermined continuous roads in a natural environment. The overall distance of the 6 roads is over 360m. Two of them are structural road where the robot walked along the center lane lines of the roads and turned at every zebra crossing. The other four roads are not structural roads where the robot went along the curb and turn at every sign. The monocular color camera is Sony EVI-D100P. The control cycle period is set to 1s and $\alpha = 1$, $\beta = 0.2$, $\gamma = 8$, $a = 15$, $b = 0.05$ in the linear equations. The average success rate on every road is over 95%, with over 98% in four of the six. The total success rate of continuous walking on the six roads is more than 80%, which reach the expectation for this experiment.



Fig. 8. (a) Lane recognition on structured road; (b) road edge recognition in unstructured road; (c) Some scene and results in experiments

6. Conclusion and Future Work

This research combines the technology of computer vision, pattern recognition, fuzzy control, etc. and realizes an intellectual visual navigation on a small robot platform. Based on the limited states, it manages the traveling condition of the robot, collaborating simultaneously others modules and underwent an experiment on six roads of known conditions. The experiments show that fuzzy controller and inertia navigation can well reduce the instability of the vision navigation. And mainly through well-designed vision navigation, small mobile robot can accomplish the task of automatic cruising in good lighting conditions on both structured and unstructured outdoor roads.

In the process of walking straight, the occasional corrections of the robot present continuous oscillations. Reasons are: 1) Although the fuzzy controller will stable the angular velocity at a certain value, the blurred image during turning will still cause an increase of errors in the visual navigation; 2) The ground friction of the four wheels of the robot are different. We should enhance the tolerance for this situation in the controlling strategy or develop new algorithms in terms of image stability. Also we should do more research for precise turning strategy in the future. Finally, we will gradually try to run our vision navigation and control system in unknown environment.

Reference

- [1] A. Haasch, N. Hofemann, J. Fritsch, and G. Sagere. Cavallaro, A multi-modal object attention system for a mobile robot, *Intelligent Robots and Systems*, 2005, 2712-2717.
- [2] Yunpeng Zhao, Hong Wang, Runchen Yan, Unstructured Road Edge Detection and Initial Position Approach Based on Monocular Vision, *AASRI Conference on Computational Intelligence and Bioinformatics*, 2012.
- [3] Nobuyuki Otsu . A threshold selection method from gray-level histograms. *IEEE Trans. Sys., Man., Cyber.* 1979, 9 (1): 62–66.
- [4] Fischler, M.A. and Bolles, R.C. Random Sample Consensus. A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 1981, 24(6): 381–395.
- [5] Mamdani, E. H., Application of fuzzy algorithms for the control of simple dynamic plant. *Electrical Engineers, Proceedings of the Institution*, 1974, 1585-1588.
- [6] J. L. Castro, Fuzzy logic controllers are universal approximators, *IEEE Transactions on Systems, Man and Cybernetics*, 1995, 25(4):629-635.