Research Article

Path Planning Algorithm Based on Visual Image Feature Extraction for Mobile Robots

Qin Dong (ID)

School of Information Engineering, Yancheng Institute of Technology, Yancheng 224000, China

Correspondence should be addressed to Qin Dong; dq@ycit.edu.cn

When autonomous mobile robots plan their own movements properly, they first need to perceive the surrounding environment and then make comprehensive decisions based on the surrounding environment information, that is, path planning. Vision can provide abundant and complete environmental information for robots, and it can significantly improve the effect of path planning when it is introduced into the path planning of autonomous robots. In this article, we take the autonomous mobile robot AS-R as the research object and use the multisensors such as a gimbal camera and ultrasonic sensor attached to the robot to study the navigation line information perceived by machine vision, the obstacle information sensed by range sensor, and the fusion of multisensor information to solve the image processing, path recognition, information fusion, decision control, and other related problems of the mobile robot to realize the autonomous mobile robot.

1. Introduction

Robotic vision systems are one of the newest and hottest topics in the field of robotics research. Many countries have invested a lot of human resources and money in the research and development of robot vision systems. Machine vision system refers to the machine vision products (i.e., image acquisition devices, both CMOS and CCD) to convert the acquired target into an image signal, which is transmitted to a dedicated image processing system and converted into a digital signal based on the pixel distribution and information such as brightness and color; the image system performs various operations on these signals to extract the characteristics of the target and then controls the field equipment based on the discriminatory results movements based on the discrimination results [1].

Path planning is an important research direction that is receiving increasing attention in the development of mobile robotics research neighborhood. Navigation technology is relatively central to the study of mobile robots, and path planning is an important and integral part of navigation research. The application neighborhood of mobile robots, in many cases, has complex unknown information about the operation space. Such a robot operating in the environment needs to be able to effectively detect its operating environment in order to construct an operating path in the environment. The design of navigation, path planning, obstacle avoidance strategies, and other operations is only possible based on an understanding of the environment. Vision is usually the most intuitive and accurate reflection of environmental information. The autonomous mobile robot can follow the predetermined task instructions and do planning according to the image information it obtains, and in the process of travel, it constantly senses the information of the surrounding local environment, makes autonomous decisions, guides itself around the obstacles to safely drive to the designated target, and performs the predetermined actions and operations. It has a wide range of application prospects in industrial, civil, and military neighborhoods [2].

The development of autonomous mobile robotics is of great significance to accelerate the modernization of national defense, industry, and agriculture in China and improve the living standards of people. In addition, the research of autonomous mobile robotics is a multidisciplinary intersection of high-tech neighborhoods, which puts forward high requirements on several disciplines such as artificial intelligence, pattern recognition, automatic control, electrical and electronic, and mechanical design. Therefore, the

research on autonomous mobile robotics has an important role in promoting the development of the above-mentioned related neighborhoods.

2. Materials and Methods

2.1. Description of the Experimental Environment. This article studies visual image-based path planning for the characteristics of the image target region of interest, the experimental environment is an indoor environment, and the path to be followed by the mobile robot is the red circular navigation path in Figure 1 below. Obstacle avoidance is mainly for the common objects rectangular-shaped obstacles in typical environments, such as walls, doors, tables, and chairs. The horizontal and vertical line features are more obvious in this environment. This section studies vision-based path recognition, considering the influence of the environment, such as lighting on the image.

2.2. Color Feature Analysis. The color space of the original image captured by the CCD vision sensor is the RGB space, which has the advantage of being simple and intuitive. The RGB model can be built in a Cartesian coordinate system, where the space of the model is a square cube with three axes R, G, and B. The origin corresponds to black, and the farthest vertex from the origin corresponds to white. In this model, the grayscale values from black to white are distributed on the line from the origin to the farthest vertex from the origin, while the rest of the points in the cube correspond to different colors, which can be expressed as a vector from the origin to that point. Generally, for convenience, the cube can be normalized to a unit cube, with all values of R, G, and B in the interval [0, 1].

First, grayscale the color image acquired by the CCD, the purpose of grayscale is to make the values of the three color components R, G, and B equal, using the algorithm of the average of the three components R, G, and B; namely,

$$R = G = B = \frac{R + G + B}{3}. \tag{1}$$

The grayscale results are shown in Figure 2 above.

From the image preprocessing study in the previous section, we know that due to the influence of burr voltage, electromagnetic interference, etc., the image information only grayed out still contains a variety of noise and distortion, so the image must be smoothed and filtered before the analysis of the image. The theoretical knowledge of filtering has been mentioned in the previous section, and methods such as mean filtering, median filtering, Gaussian filtering, and edge-holding filtering can be used. In practical applications, simple smoothing, Gaussian smoothing, or median filtering of images should be used flexibly for image enhancement according to the image characteristics and processing requirements [3–5], and there is no superiority or inferiority between these algorithms, only that the application is not the same.

According to the comparative analysis of their experimental results in the existing literature, the median filtering and Gaussian filtering methods are chosen for the indoor environment images in this article.

Figure 1: Robotic indoor environment.

2.2.1. Median Filtering. Median filtering of an image is a nonlinear image processing method that determines the grayscale of the central pixel by the result of sorting pixels in the neighborhood by grayscale, and its idea is quite different from the idea of mean processing. The basic idea of median filtering is to replace the gray value of a pixel with the median of the gray value of the pixel's neighborhood. This method removes the impulse noise and pretzel noise while preserving the image edge details.

The steps of the simulation experiment are as follows: first determine window W containing $(2n + 1) \times (2n + 1)$ pixels, and after each pixel in the window is lined up according to its gray size, replace the original $f(x, y)$ with the gray value of its middle position to obtain the enhanced image $g(x, y)$, which can be expressed as follows:

$$g(x, y) = \text{media}\{f(x - k, y - l), \quad (k, l) \in W\}. \tag{2}$$

After the experiment, it can be seen that the median filter can effectively remove the noise points in the image, especially in a continuous change in the area of gentle (such as human clothes and skin) and almost 100% remove the gray mutation points (can be considered as noise points), because of this, the median filter is not suitable for use in some details, such as detail points and detail lines in the image, because the detail points may be removed as noise points. The window of the median filter can also have various shapes; the above program chose a rectangle (to facilitate the calculation); in fact, the window can also be a diamond, circle, cross, etc. Different window shapes have different filtering effects; for objects with slow and long contour lines, they are suitable for rectangular or prototype windows and for images with sharp top corner objects, they are suitable for cross-shaped windows [6, 7]. The median filtering can be combined linearly, and the filters with different window shapes can be combined linearly with each other.

2.2.2. Gaussian Smoothing Filtering. Gaussian smoothing of an image is also a method of smoothing an image using the idea of neighborhood averaging. Unlike simple smoothing of an image, in Gaussian smoothing of an image, pixels at different locations are given different weights when averaging over the image neighborhood. Gaussian filtering is a class of linear smoothing filtering methods that selects the template weights according to the shape of the Gaussian function. Gaussian smoothing filtering is more effective in

Figure 2: Grayscale of path images.

removing noise that obeys the normal distribution. The two-dimensional Gaussian function formula is as follows:

$$h(i, j) = \frac{1}{\sqrt{2\pi}\sigma}e^{-i^2+j^2/2\sigma^2}. \tag{3}$$

The width of the Gaussian filter (which determines the degree of smoothing) is evidenced by the parameter σ, and the relationship between σ and the degree of smoothing is very simple. The larger σ, the wider the band of the Gaussian filter and the better the degree of smoothing. By adjusting the smoothing parameter σ, a compromise can be made between blurring the image feature components (over-smoothing) and smoothing the image with an excessive amount of undesired mutations due to noise and fine texture (undersmoothing). Due to the separability of Gaussian functions, large Gaussian filters can be efficiently implemented. The convolution by a two-dimensional Gaussian function can be performed in two steps, first convolving the image with a one-dimensional Gaussian function and then convolving the result of the convolution with the same one-dimensional Gaussian function with perpendicular orientation. As a result, the computational effort of 2D Gaussian filtering grows linearly rather than squarely with the width of the filter template. These properties make it particularly useful in early image processing, showing that Gaussian smoothing filters are very effective low-pass filters in both the spatial and frequency domains. The essence of the air-domain Gaussian smoothing filter is a weighted mean filtering method, which can be expressed as follows:

$$g(i, j) = \sum_{m=-k}^{K} \sum_{n=-L}^{L} W(m, n) f(i+m, j+n), \tag{4}$$

where $W(m, n)$ is the weight coefficient and the Gaussian filter window is $(2K+1) \times (2L+1)$. The two filtering effects of the experimental images do not distinguish much, and both meet the requirements of the accuracy of the in-turn processing steps.

After processing Figure 3 alone, it is still not possible to determine the type (road, shadow, obstacle, or background), and it continues to be necessary to classify the original image by the difference in color. In this article, the binarization method, which is simple in the algorithm, easy to understand and implement, fast in computation, small in memory consumption, and low in computing equipment requirements, is used to segment the image by color features.

The obtained binary image is denoised: firstly, the image is corrupted, and then the noise points, scatter, and burrs are removed from the binary image without changing its size by performing a single expansion operation on the corrupted image with the same structural elements. The desired image, segmented from the binarized image, is refined [8] to reduce the amount of data processing. Refinement is the process of removing pixels from the edges of an image and extracting the "skeleton" from the image so that the image is only one pixel wide, reducing the image components and leaving only the most basic information about the region for further analysis and recognition. In the process of refining an image, two conditions should be met: first, the image should shrink regularly in the process of refinement; second, the image should be gradually reduced in the process so that the connectivity of the image remains unchanged.

2.3. Path Extraction Based on Color Space Model. We put the algorithm through the AS-R mobile robot vision platform to perform a simple path recognition experiment. In the indoor environment, red tape is applied as the guide path, and the path is identified from the image based on color features. The path acquisition method is done in RGB color space, and the image preprocessing results are shown in Figure 4. It can be seen from Figure 4 that the path image obtained by this method is easy to observe, intuitive, and simple, but for some specific colors, RGB is difficult to extract their features. The extraction effect is not particularly good because it is easily influenced by the light intensity and the surrounding environment, which brings problems to the later acquisition of path information, so we extract the paths in HSV space instead.

The processing of light habits by the human visual system is distinguished by color (Hue), color saturation (Saturation), and brightness (Value). The parameter H represents the color information, expressed by 0° to 360°; red, green, and blue are separated by 120°; the hue is mainly
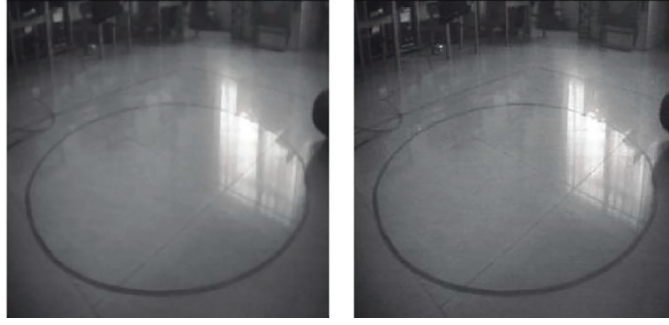
FIGURE 3: Image filtering results.

Original image   Grayscale image

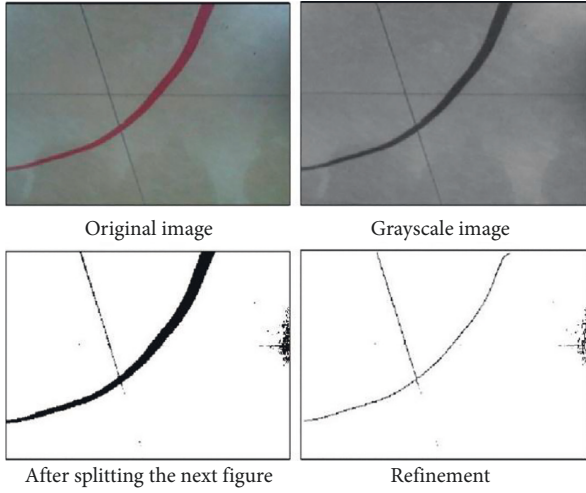After splitting the next figure   Refinement

FIGURE 4: RGB space extraction effect.

determined by the wavelength of each component of the visible spectrum, which is the basic characteristic of color light; the saturation parameter S reflects the intensity of color, which depends on the content of white light in colored light, the more white light is mixed in, the lighter the color is. S is a proportional value, ranging from 0 to 1, indicating the ratio between the selected color and the maximum purity of that color. The parameter V indicates the brightness of the color, also from 0 to 1. Brightness refers to the intensity of light stimulation caused by colored light to the human eye, which is only related to the energy of light, and the color of light. h and s components are basically unaffected by the light and are used for image processing with good results.

The transformation equation from RGB space to HSV space is as follows:

$$V = \frac{1}{3}(R + G + B),$$

$$S = 1 - \frac{3}{R - G - B}\min(R, G, B),$$ (5)

$$H = \begin{cases} \theta\,(B \leq G), \\ 360 - \theta\,(B > G). \end{cases}$$

Among them,

$$\theta = \cos^{-1}\left(\frac{2R - G - B}{2\sqrt{(R - G)^2 - (R - B)(G - B)}}\right).$$ (6)

2.4. Research on Path Recognition Algorithm Based on Vision and Sensor Fusion. Road area recognition by visual features alone cannot satisfy the accuracy of the algorithm because of the large amount of uncertainty knowledge applied therein, so we fuse multisensor information to synthesize the indoor environment. Multisensor fusion requires first describing the sensor information in a concrete mathematical form and then using the corresponding mathematical tools to process it. The internal odometer, external ultrasound (sonar and PSD), and laser rangefinders [9, 10], with which the AS-R mobile robot is equipped, are modeled below.

Mobile robot platforms with different physical structures are mapped to different motion models. For this article, the kinematics of the AS-R mobile robot is based on the working principle of the odometer, which is based on the photo-electric encoders mounted on the motors of the two driving wheels to detect the radian of the wheel rotation in a certain time, and thus the change of the relative position of the robot is deduced. Combining the kinematic model shown in Figure 5, the odometer kinematic model of the mobile robot can be expressed as follows:

$$X_{K+1/K} = f\left(X_{K/K}, U_K\right) + \omega_k,$$ (7)

where ω_k is the process input noise and is assumed to obey a Gaussian white noise distribution. Its covariance matrix is

$$P_{K+1/K} = \left(\frac{\partial f}{\partial X}\right)P_{K/K}\left(\frac{\partial f}{\partial X}\right)^T + \left(\frac{\partial f}{\partial U}\right)Q_K\left(\frac{\partial f}{\partial U}\right)^T.$$ (8)

3. Results and Discussion

3.1. Machine Vision-Based Path Planning for Mobile Robots. Combining the video image processing data collected from the mobile robot and data from range sensors such as lasers, information such as the position of the mobile robot, navigation lines, and the orientation of obstacles can be obtained. Mobile robots make comprehensive decisions based on this information to plan a suitable travel path [11].
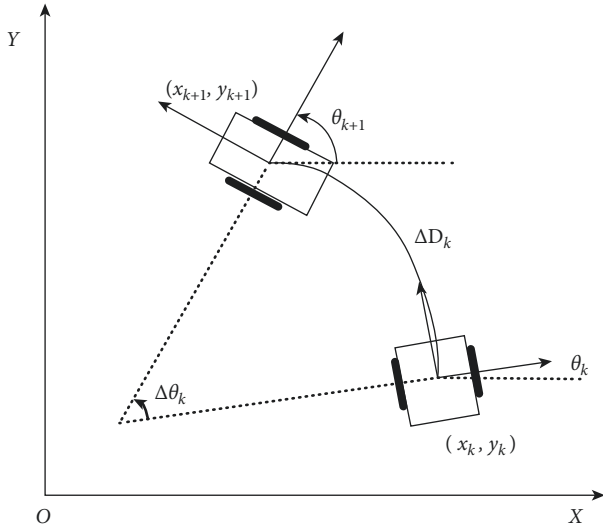
FIGURE 5: Odometer-based robot motion model.

FIGURE 6: Input variables for the fuzzy controller.

The path planning algorithm used in this article is a fuzzy logic control technique.

Fuzzy logic is a reasoning method that deals with imprecisely described information based on fuzzy set theory. It is oriented towards imprecise descriptions of things' characteristics and capabilities as opposed to uncertainty reasoning dealing with the possibility of random events occurring. The fuzzy logic method simulates the driver's driving thought, considering that artificial driving is a fuzzy control behavior. The size of the curvature of the path, the size of the position, and directional deviation are fuzzy quantities obtained by the human eye, the driver's driving experience cannot be precisely determined, and fuzzy control is precisely imitating this human brain's uncertain conceptual judgment, reasoning way of thinking. For the description system whose model is unknown or cannot be determined, the strong nonlinear, large lag control objects, the application of fuzzy sets and fuzzy rules for reasoning, expressing transitional boundaries or qualitative knowledge experience, simulating the way of the human brain, and implementing fuzzy integrated judgment is an effective way to solve such problems. Because the road environment information is generally obtained through machine vision, ultrasonic and other sensors have the problems of approximation, imperfection, and being mixed with certain noise; one advantage of fuzzy control is that it can accommodate such uncertain input information and produce smooth control out. In addition, mobile robots and vehicles are similar in that their dynamics models are more complex, whereas fuzzy control is not required to control the mathematical model of the system. Both mobile robots and vehicles are typically time-delayed, nonlinearly unstable systems, and fuzzy controllers can accomplish a nonlinear mapping from input space to output space. Thus, combining the robustness of fuzzy control with the physiology-based "perception-action" behavior avoids the disadvantages of traditional algorithms, which are sensitive to the positioning accuracy of mobile robots and highly dependent on
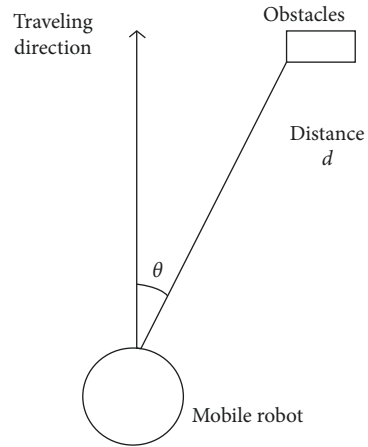
environmental information, and provides a more effective solution to the motion planning and navigation of mobile robots in unknown or uncertain environments. It provides a more effective solution to the problem of motion planning and navigation of mobile robots in unknown or uncertain environments.

3.1.1. Fuzzy Description of Input and Output Variables. Fuzzy logic control generally consists of four steps: fuzzification process, rule base building, fuzzy inference, and defuzzification [12]. In the path planning problem, the input quantity is the information about the robot's travel orientation and the distance between the robot and the surrounding obstacles, and the control quantity is the translational linear velocity and rotational angular velocity of the robot. In this article, the input variables of the controller are the distance d between the robot and the obstacle and the azimuth angle θ of the obstacle with respect to the target direction (travel direction), and the output variable is the rotation angle of the robot after encountering the obstacle. A schematic diagram of the relationship between the input variables is shown in Figure 6.

In fuzzy control used for fuzzy inference are fuzzy variables, the information of obstacle distance and target direction in each direction in the environment collected by the mobile robot are specific quantities, so there is a conversion process. In this article, we use a continuous type theoretical domain to quantize the range of distance d between the mobile robot and the obstacles uniformly to the interval [0, 8] using a simple linearization process, and the theoretical domain of d is [0, 1, 2, 3, 4, 5, 6, 7, 8], and the range of input angle θ is nonuniformly quantized to the interval [−4, 4] (uniformly quantized in the interval [−60°, 60°] when $\theta > 60°$). The range of the output angle [−60°, 60°] is uniformly quantized to the interval [−4, 4], and the range of the output angle [−60°, 60°] is uniformly quantized to the interval [−4, 4], and the domain is the same as θ. Usually, in practical applications, seven to nine fuzzy states are selected, namely, positive large (PB), positive medium (PM), positive small (PS), zero (Z), negative small (NS), negative kind

(NM), and negative large (NS). The affiliation function and fuzzy partition graph are shown in Figures 7 and 8.

The shape of the affiliation function of each linguistic variable is a symmetric triangle, and the fuzzy partition is completely symmetric; the fuzzy partition graph of d and the partition graphs of azimuthal angle θ and output angle φ are shown above (the partition graphs of θ and φ are similar, except that the right turn is positive in the partition graph of φ, and the affiliation functions of the input and output variables are shown in Tables 1–3).

3.1.2. Establishing Fuzzy Control Rules.
Fuzzy control rules are the key part of a fuzzy controller. Fuzzy control rules are usually established in the following ways:

(1) From the operator's experience: for a specific process, a set of rules is generalized based on long-term operating experience.

(2) From field experiments: in the case of conditions permitting, through the manual setting of the control role, after the synthesis of experimental data and generalization, to obtain control rules.

(3) From the knowledge and reasoning of the process: based on fuzzy models to establish fuzzy control laws, i.e., both the controller and the control object are described in a fuzzy way.

(4) Based on learning, let fuzzy control have a human-like learning function, i.e., the ability to generate fuzzy control rules and modify them based on experience and knowledge. They are not mutually exclusive; on the contrary, a combination of these methods can better help to build a fuzzy rule base [13].

For the same controlled object, different methods and different designers may result in different control rule tables. However, as far as the implementation of control is concerned, any control rule table must have the three properties in the following table.

Based on the above methods and principles of establishing fuzzy rules, imitating the combination of artificial driving behavior, when the measured obstacle distance is far, that is, in a safe state, the mobile robot should follow the target azimuth to try to align with the target forward; when the obstacle distance is close, the robot makes a reasonable decision according to the obstacle distribution combined with the target orientation, in order to ensure obstacle avoidance while moving as close to the target direction as possible. In this article, the basic idea of obstacle avoidance in path planning is as follows: when the obstacle is located on the left (right) side of the axis position line (travel orientation) of the mobile robot, then the robot turns right (left), and when the obstacle is located at a small distance directly in front of the robot, then the default robot turns left in the maximum direction. The input variables of the fuzzy controller are d and θ, both with a fuzzy gradation number of 5, so they correspond to 25 fuzzy control rule numbers. Tables 4 and 5 show the established rule table.
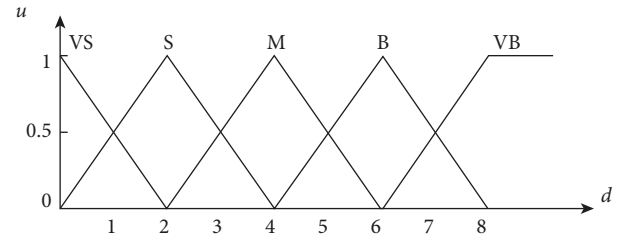

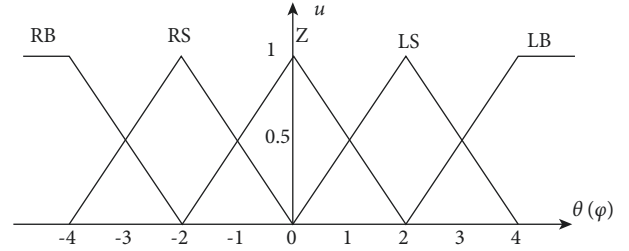FIGURE 7: The affiliation function and fuzzy partition graph of d.

FIGURE 8: Affiliation function of θ and fuzzy partition graph.

TABLE 1: The affiliation function of d.

	0	1	2	3	4	5	6	7	8
VS	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S	0.0	0.5	1.0	0.5	0.0	0.0	0.0	0.0	0.0
M	0.0	0.0	0.0	0.5	1.0	0.5	0.0	0.0	0.0
B	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.5	0.0
VB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0

TABLE 2: The affiliation function of θ.

	−4	−3	−2	−1	0	1	2	3	4
RB	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RS	0.0	0.5	1.0	0.5	0.0	0.0	0.0	0.0	0.0
Z	0.0	0.0	0.0	0.5	1.0	0.5	0.0	0.0	0.0
LS	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.5	0.0
LB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0

TABLE 3: The affiliation function of φ.

	−4	−3	−2	−1	0	1	2	3	4
TLB	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TLS	0.0	0.5	1.0	0.5	0.0	0.0	0.0	0.0	0.0
TZ	0.0	0.0	0.0	0.5	1.0	0.5	0.0	0.0	0.0
TRS	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.5	0.0
TRB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0

3.1.3. Fuzzy Reasoning and Clarity.
There are two types of fuzzy controllers, Mamdani and Sugeno types [14]. Both controllers have a rule base, which is a set of if-then fuzzy rules. The typical rule of the controller is "if x is A and y is B, then $Z = f(x, y)$." Here A and B are fuzzy sets, and $Z = f(x, y)$ is a function of x, y, usually a polynomial in the input variables x, y. When $f(\ )$ is constant, it is a zero-order Sugeno model. Sugeno can be generally considered as a special case of Mamdani controller. So far, only Mamdani fuzzy controllers have been used in speed control systems. In this article, the

TABLE 4: Fuzzy control rule table properties.

Consistency	It is required that the fuzzy control determined by these index requirements cannot be contradictory to each other. The conclusions of the rules must be the same or close under the same or similar input conditions, called rule consistency.
Completeness	For any input, the fuzzy controller should give a suitable control output; this property is called completeness.
Regularity	The actual number of fuzzy rules that should be taken depends on many factors. The general principle is to take as few rules as possible to simplify the design and implementation of the fuzzy controller while satisfying the completeness condition.

TABLE 5: Fuzzy rule table.

	RB	RS	Z	LS	LB
VS	TLB	TLB	TLB	TRB	TRB
S	TLB	TLB	TLB	TRB	TRB
M	TLS	TLS	TZ	TRS	TRS
B	TLS	TZ	TZ	TZ	TRS
VB	TLS	TZ	TZ	TZ	TRS

Mamdani model is used to derive the clear values using the weighted average method, which is a clarification method, and then the clear values are transformed into the actual control quantities. The corresponding control quantities are calculated offline using MATLAB's fuzzy controller for different combinations of input cases to obtain a control table.

$$\varphi = d \times \theta \circ R,$$
$$R = \bigcup_{I=1}^{25} R_i, \quad (9)$$
$$R_i = (d_i \text{and} \theta_i) \longrightarrow \varphi_i,$$

where R is the fuzzy relationship matrix and the control quantity φ is clarified to be transformed to an exact value.

When $d = 0$, obstacles do not pose a threat to the mobile robot or there are no obstacles in the robot's field of view, i.e., autonomous pathfinding strategy (in an environment with navigation lines, it can be assumed that the robot has only navigation lines in its field of view, at which time driving along the navigation lines is the most appropriate solution); when $d \neq 0$, it indicates that the mobile robot has entered the danger area and needs to initiate obstacle avoidance operation to meet driving safety.

Based on the motion control decision of the fuzzy control table, the motion step length plays an important role in the result of path planning. If the robot step length is too small, the robot information processing system does not have enough time to process the adjacent two samples; if the motion step length is too large, the robot will not be able to "brake" and hit the obstacle. Therefore, the step length must be controlled within a reasonable range, and the robot step length should be reduced at the turn so that the robot can get enough response steps. Depending on the values of the input variables d and θ, the mobile robot cannot move forward with a large step (or a fixed step) immediately after turning at the output angle of the fuzzy controller. This is because the information about the environment, such as information about obstacles, or the perception of target points (navigation paths) in the environment where they exist, is unknown at the new travel position after the robot has turned.

Therefore, the mobile robot should detect again whether it needs secondary obstacle avoidance (or travel orientation rotation) after rotation and then advance a safe step. This can be considered as an iteration between fuzzy obstacle avoidance and autonomous tracing strategies during the travel of the mobile robot.

3.2. Validation and Analysis. Based on the above learning, simulations were conducted in MATLAB 7.0 environment to verify the fuzzy logic control algorithm. In the simulation process, the shape and location of the obstacles are set arbitrarily to simulate the actual application. Under the premise that the map is unknown, the positioning information is first obtained by the vision sensor, and the distance and orientation information between the mobile robot and the obstacles are sensed by the range sensor, which is used as the fuzzy input quantity for fuzzy inference using the Fuzzy Logic Toolbox in MATLAB. The simulation program assumes that the robot is a mass point and that the obstacles within 360° direction and 1 m~3 m distance can be detected accurately, and the positions of the starting point and target point are set arbitrarily, and the trajectory of the robot is depicted while the path planning is carried out so that the correctness and reliability of the algorithm can be checked. The simulation results are shown as follows:

(1) Set the coordinates of the start point (0, 0) and the target point (10, 10), and the obstacles are block obstacles with more obvious horizontal and vertical line features, where the obstacles are dense, the algorithm is computationally intensive, and the robot travels slowly, as shown in Figure 9.

(2) Set the coordinates of the starting point (5, 5) and the coordinates of the target point (25, 25), and set the circular obstacle; the effect of obstacle avoidance is shown in Figure 9.

The simulation results are shown in Figure 10 that the mobile robot can safely avoid obstacles with this algorithm, the motion trajectory is smooth, and it can reach the target point faster, with certain real-time and robustness. This section uses fuzzy understanding for robot local path planning: reference to the human driving experience to get control rules, converting real-time measurement information obtained from sensors into fuzzy input quantities, calculating fuzzy control table by fuzzy inference, and getting control decision information of the robot by querying the control table, which has a relatively satisfactory experimental effect, relatively clear and simple algorithm, and small computational effort.
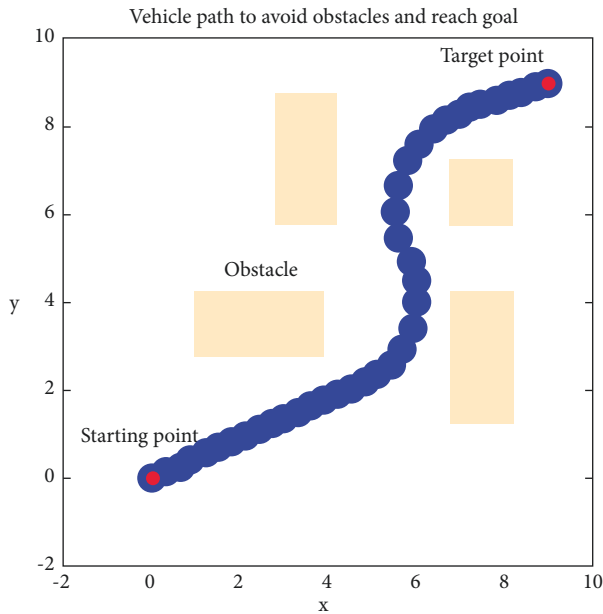
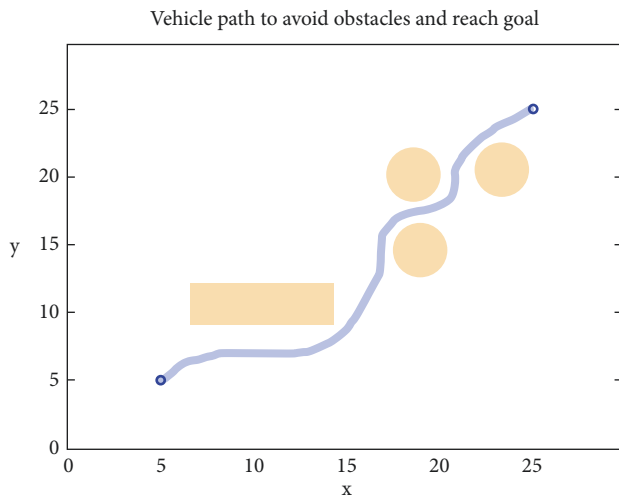FIGURE 9: Path planning in block obstacle environment.

FIGURE 10: Path planning in multiobstacle environments.

The fuzzy logic algorithm bypasses the disadvantages of traditional algorithms that are sensitive to the positioning accuracy of mobile robots and dependent on environmental information and uses a relative positioning approach that eliminates cumulative errors and keeps the computational effort low. It shows great superiority for planning problems in unknown environments where only approximate and uncertain information data are available, with strong real-time performance. However, fuzzy control theory for robot path planning also has inherent drawbacks, such as the establishment of rules, where human experience is not always complete, and the inference rules or fuzzy tables can expand dramatically when their input volume increases. Optimization of the algorithm can also consider the following: (1) global path planning and local path planning can be integrated according to different environments combined

with vision's own characteristics; (2) path search problems in different environments with and without obstacles can also be distinguished in the planning to improve efficiency. How to closely combine the traditional path planning algorithm with artificial intelligence technology to improve the planning capability of the algorithm is the trend of future research on robot path planning algorithms.

4. Conclusions

In this article, based on the AS-R robotics platform, the research on mobile robots can be summarized as follows: modeling of robots and their sensors, image processing, and path planning. Using vision and ranging sensors commonly used in mobile robots as the research platform, the path planning problem of mobile robots based on vision images is investigated. Comparing with the traditional RGB space image extraction method, the paths are extracted in HSV color space and the original path images are acquired based on the AS-R robot vision module. The path parameters are obtained by performing the coordinate transformation. The robot kinematic model is constructed, and the odometer gives the relative position in robot motion mainly by cumulative measurement. For the uncertain information obtained from multiple sensors, control rules are obtained with reference to the driving experience, a fuzzy controller is designed, the fuzzy method is used for local path planning of the robot, and simulation results are given for obstacle avoidance of the mobile robot. Improving the performance of image processing algorithms or using other sensors to partially eliminate the uncertain information in vision and orientation measurement is an important topic for future research.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

References

[1] G. Zhang, *Machine Vision*, Vol. 6, Science Press, , Beijing, 2005.

[2] Y. Dong, "Research on path planning methods for mobile robots," *Information & Technology*, vol. 30, no. 6, pp. 108–111, 2006.

[3] G. Welch and G. Bishop, *The Instruction of Kalman Filter*, Vol. 07, Department of Computer Science University of North Carolina at Chapel Hill, , Chapel Hill, 2006.

[4] A. J. Sousa, P. J. Costa, A. P. Moreira, and A. S. Carvalho, "Self localization of an autonomous robot: using an EKF to merge odometry and vision based landmarks," in *Proceedings of the 10th IEEE Conference on Emerging Technologies and Factory Automation*, vol. 1, pp. 19–22, Catania, Italy, September. 2005.

[5] J. Goncalves, J. Lima, and P. Costa, "Real-time localization of an omnidirectional mobile robot resorting to odometry and global vision data fusion: an EKF approach," in *Proceedings of the 2008 IEEE International Symposium on 2008 IEEE International Symposium on Industrial Electronics*, Cambridge, UK, July 2008.

[6] R. Jirawimut, S. Prakoonwit, F. Cecelia, and W. Balachandran, "Visual odometer for pedestrian navigation," in *Proceedings of the 19th IEEE Transactions on Instrumentation and Measurement Technology Conference (IEEE Cat. No.00CH37276)*, vol. 52, no. 4, pp. 1166–1173, Anchorage, AK, USA, May 2002.

[7] S. Frintrop and P. Jensfelt, "Attentional landmarks and active gaze control for visual SLAM," *Robotics, IEEE Transactions on Volume*, vol. 24, no. Issue 5, pp. 1054–1065, October. 2008.

[8] W. H. Huang, B. R. Fajen, J. R. Fink, and W. H. Warren, "Visual navigation and obstacle avoidance using a steering potential function," *Robotics and Autonomous Systems*, vol. 54, no. 4, pp. 288–299, 2006.

[9] F. Yang, "Multi-sensor-based map construction and navigation for mobile robots," vol. 03, Hefei University of Technology, Hefei, 2008, Master's thesis.

[10] Y. Fu, "Visual navigation and obstacle avoidance for wheeled mobile robots,", Master's thesis, Hefei University of Technology vol. 03, , 2009.

[11] Z. Cai, H. He, and H. Chen, *Theory and Methods of mobile Robot Navigation Control in Unknown Environments*, Vol. 1, Science Press, , Beijing, 2009.

[12] P. K. Gaonkar, D. S. Anthony, and K. S. Rattan, "*Fuzzy navigation for an autonomous mobile robot*," M.S.thesis, Wright State University, Dayton, 2005.

[13] C. Cai and Q. Zhu, "Simulation of fuzzy controller-based path planning for mobile robots," *Computer Simulation*, vol. 25, no. 3, pp. 182–186, 2008.

[14] Z. Cai, *Principles and Applications of Intelligent Control*, p. 6, Tsinghua University Press, Beijing, 2008.