Research on Mobile Robot Vision Navigation Algorithm

Ling Guo¹*, Chao Sun²

¹Department of Automation Engineering, Guangdong Polytechnic of Water Resources and Electric Engineering, Guangzhou, Guangdong, 510635, China
²Department of Information Engineering, Guangzhou Hua Shang Vocational College, Guangzhou, Guangdong, 511300, China

Corresponding author’s e-mail: guol@gdsdxy.edu.cn

Abstract: Robot navigation is one of the key technologies of indoor robots and it is the primary solution to the problem of intelligent robot. In this paper, the visual navigation problem of mobile robot was analyzed and the robot vision navigation algorithm based on monocular vision was proposed. The algorithm obtained three-dimensional information of obstacle through data fusion, builder a three-dimensional model of obstacle, and optimized the navigation path by using the heuristic best search algorithm, and realized the drawing of the robot navigation path.

1. Introduction
Obstacle avoidance and path planning play an important role in the development of service robots[1-2], which is the basis for their autonomous decision-making and the prerequisite for their further research and development. Robot avoidance refers to the robot to follow certain performance requirements (optimal path, the shortest time, no collision, etc.), to find the optimal path. Obstacle avoidance is an important step in the navigation of most mobile robots. Indoor service robots often encounter positioning accuracy problems, environmental sensory problems, and obstacle avoidance navigation algorithms during obstacle avoidance navigation[3-4]. These sensors are susceptible to outside interference and have limited applications, and are difficult to detect on the ground, small or flat objects, and cannot distinguish between different types of surface. Advanced creatures in nature, such as humans, can take advantage of the visual access to environmental information and self-guide through the information they receive. As a way of sensing robots, vision provides a very attractive solution. The vision system can be thought of as a passive sensor, and getting the data by receiving light or waves does not change the environment compared to the aforementioned active sensors, and the resulting image contains more information. In stereo vision, you must know camera and robot related parameters. Considering the cost, image processing ability and task requirements of the small indoor wheel robots, monocular vision is introduced as obstacle avoidance solution. Corresponding image processing algorithms and direction decision algorithms are proposed for monocular vision in order to realize obstacle avoidance mission.

2. Material and Methods

2.1. robot navigation method
Robot autonomous navigation is through the calculation of the sensor to obtain the external environment information to extract the robot navigation available information. Common principles and characteristics of the navigation[5-13] were as shown in Table 1.


| Navigation mode          | principle                                                                 | Characteristic                                                                 |
|--------------------------|---------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| Inertial navigation      | By measuring the acceleration of the carrier in the inertial reference frame, integrating time and transforming it into the navigation coordinate system, we can get the information of speed, yaw angle and position in the navigation coordinate system to control the mobile robot. | Not limited by region, time and climate conditions, the output parameters are comprehensive. Small errors will be endless after integration, affecting navigation accuracy and inappropriate navigation and location for a long time. |
| Landmark navigation      | A number of guiding cables are continuously embedded on the path, passing through the current of different frequencies, sensing the path information by detecting the current through the induction coil. | It has high reliability, high practicability, high accuracy and long life. But the cost is high and the computation is more complex. For simple obstacle avoidance function, it cannot be easily realized, and its variability and maintainability are poor. |
| GPS Navigation           | Taking some special scenes in the environment as roadmap, the robot knows the characteristics of the road signs in the environment such as coordinates and shapes, and determines their location by detecting the road signs. At the same time, the global route is decomposed into a fragment between the road sign and the road sign, and the road sign is constantly detected to complete the navigation. | It is easy to implement and low cost, but it needs to artificially change the working environment of robot, and lacks flexibility and practicability. The stability and robustness of landmark detection is the main problem of research. |
| Visual navigation        | By analyzing the reception and analysis of satellite signals, we determine the location of the navigation objects on the surface. Based on the space satellites, we can determine the target location; speed and location time globally, and conduct high-precision navigation. | Good autonomy and strong adaptability. However, it will be influenced by high ground objects on the ground. The calculation is large, the real-time performance is reduced, the dynamic performance is strong, and the error is easily produced, and the positioning accuracy is affected. |
| Magnetic navigation      | The detection and recognition of obstacles and road signs (the robots own characteristics of the environment) is detected by the visual sensor. | It has strong adaptability, wide range of signal detection, complete information acquisition, no need to set up the navigation route map, and is limited by the computing speed and storage capacity of the existing computing equipment. |
2.2. **Classification of robot vision navigation**

Visual navigation classification is very diverse, the specific classification methods and principles[14-15] shown in Table 2.

| Classification basis | Concrete method | Principle |
|----------------------|-----------------|-----------|
| According to the type of sensor | Passive vision navigation | As a typical passive imaging sensor, the CCD camera is widely used in the vision navigation system, depending on the visible and invisible optical imaging techniques. |
| | Active vision navigation | Active vision navigation makes use of active detection methods such as laser radar sonar to navigate the environment. |
| | Map oriented visual navigation | Using pre-stored navigation map (map usage system), or obtaining local environmental information during navigation, we can conduct online safety assessment on the environment (map building system). It can also be divided into map use system and map building system. |
| If you need a map or not | Non Map Visual Navigation | The image segmentation, optical flow calculation and inter frame feature tracking are used to obtain visual information. |
| | Ground vision navigation | Early visual navigation is mainly used in unmanned ground vehicles. With the rapid development of visual navigation of UAV has been used more widely, due to the size and weight of the payload without man-machine system, many traditional navigation equipment is not suitable for carrying, and the premise of visual method using CCD camera in low weight, small size of the UAV system provides reliable navigation solution. Although the underwater visual navigation system is severely limited in turbid waters, it effectively reduces the size of the system, reduces the cost, and can significantly improve the system resolution. |
| | Unmanned aerial vehicle vision(UAV) navigation | |
| | Underwater visual navigation | |

2.3. **The composition of the robot vision system**

The robots studied at this stage have similar human functions, and they use the senses of vision, hearing and touch to observe and understand the external environment and solve some problems autonomously and effectively. Vision system enables robots to use image processing technology to analyze the collected images to obtain more valuable information. Visual system is composed of three modules: image acquisition, image processing and information output. (1) Image acquisition module. The acquired image quality is directly related to the performance of the vision sensor and has a direct impact on subsequent image processing. (2) Image processing module. As the core of visual system, image processing mainly includes image preprocessing, image segmentation, feature extraction, comprehension and decision-making. (3) Information output. The ultimate goal of image processing: according to the results of the previous module to understand and make judgments, in accordance with the provisions of the form (information or pictures) output.

3. **Results**

3.1. **Obstacle recognition**

The detection of obstacles is the process of distinguishing the objects on the ground and the ground, that is, the obstacle pixels and the ground pixels are separated in the collected images. The proposed algorithm for obstacle detection and recognition is shown in Fig.1.
The specific implementation steps of the algorithm are shown in Table 3. In order to avoid the impact of shading and illumination on image processing, the color image of RGB space is converted to HSI space by the geometric derivation method, and the formula is:

\[ S = 1 - \frac{3}{R + G + B} \min(R, G, B) \] (1)

\[ I = \frac{1}{3} (R + G + B) \] (2)

\[ \theta = \arccos \left[ \frac{1}{2} \frac{(R - G) + (R - B)}{(R - B)^2 + (R - G)(G - B)^2} \right] \] (3)

\[ H = \begin{cases} \theta, & B \leq G \\ 360 - \theta, & B > G \end{cases} \] (4)

Where, \( T \) is the weight, \( C_H(x, y), C_s(x, y), C_I(x, y) \) respectively, the edge \((x, y)\) in the H, S, I component of the edge information. The algorithm uses the maximum between-class variance method to determine the edge information threshold, the resulting binary edge:

\[ C(x, y) = \begin{cases} 1, & C(x, y) \geq T \\ 0, & C(x, y) < T \end{cases} \] (5)

It can be seen from Figure 2 that any point with coordinates \((u, v)\) in the image plane and the two-dimensional coordinate point in the robot / camera coordinate system \((x, y)\) the relationship[14] is:
\[ x_c = \frac{uh}{f \sin \theta + v \cos \theta} \]  
\[ y_c = \frac{h(v \sin \theta - f \cos \theta)}{v \cos \theta + f \sin \theta} \]

Where \( h \) is the height from the ground to the optical center of the camera; \( f \) is the camera focal length; and \( \theta \) is the camera pitch angle. In Fig. 2 (a) \( y_{c_{\text{min}}} = A \), \( y_c \) Less than A blind spot, \( y_{c_{\text{min}}} = B \), \( x_{c_{\text{min}}} = C / 2 \), \( y_{c_{\text{max}}} = D / 2 \). The process of regional growth is as follows:

Given a seed point, let the point be \((x_0, y_0)\). (2) Find the surrounding 8 fields points \((x, y)\) into the same area, pushing \((x, y)\) onto the stack. (3) Remove a similar pixel from the stack and return it to (2) as \((x_0, y_0)\). (4) return to (1) when the stack is empty. (5) Repeat (1) ~ (4) until the termination of the conditions, then the growth of the end. In this algorithm, pixels near the center of the bottom of the image are selected as seed points for regional growth. The pixel area to be grown is a free area, and the point at which the growth is ended is the edge of the obstacle. The area beyond this edge that is not grown is the hidden area, and the non-barrier area (free area) obtained through the area growth is the accessible area.

![Imaging model](image.png)

![Camera installation and perspective](image.png)

Figure 2 Robot vision system
Table 3  Algorithm specific steps

| Serial number | Step                                      | Specific operation content                                                                 |
|---------------|-------------------------------------------|-------------------------------------------------------------------------------------------|
| Step 1        | Capture images from a camera              | The camera mounted on the robot is constantly collecting video streams in front of the environment and collecting a frame image from the video stream for processing, such as Figure 3 (a). |
| Step 2        | Converting the collected color images to HSI color space | In the HSI model, the luminance component is independent of the color information of the image. The external factors in the indoor environment, such as the uneven light and the shadow of the object, have a great influence on the visual navigation of the robot. |
| Step 3        | Converting the collected color images to HSI color space | The three components obtained by space conversion are low pass filtering to reduce noise interference and produce smooth or fuzzy effects on adjacent pixels. The algorithm uses arithmetic mean filtering. In experiment, a $5 \times 5$ convolution mask is used for $S$ and $I$ components, while $H$ component needs larger convolution mask for each pixel center. |
| Step 4        | Canny edge detection for filtered images  | In edge detection, the false response to single edge caused by noise is more common. This "texture" phenomenon can be eliminated by lag thresholding. There are two thresholds for lag thresholding, that is, high threshold and low threshold. The value can be determined according to the estimation of SNR, and the recommended high and low threshold ratio is about 3:1. |
| Step 5        | Edge information synthesis                | The edge information of three components of a pixel HSI is synthesized to get the edge of the pixel. In the algorithm, the weighted edge is used to get the final edge, the synthetic edge $C(x, y)$ of pixels $(x, y)$. |
| Step 6        | Threshold processing                       | There are many weak edge information and non-obstacle edges in the synthetic edge information image, and the weak edge makes the obstacle outline not clear. In this algorithm, the maximum inter class variance method is used to determine the edge information threshold. |
| Step 7        | Thickening and coordinate conversion      | The edge image contains many small stray edges. The edge is thickened by the expansion of the morphological operation, and the thickened shape |

![Image processing result](a) Original image  
(b) Regional growth  
(c) Area Threshold Processing  
(d) Coordinate conversion
connecting the edges of the image can be controlled by selecting the appropriate structural elements for the execution of the expansion.

**Step 8 Region growth segmentation for coarse images**

In order to find out the obstacle area and the non-obstacle area in the image, the region growth segmentation algorithm is performed on the edge thickened image. Region growing segmentation algorithm is to gather the pixels according to the similarity of pixels within the same region, from the initial region (such as a single pixel), will have the same properties of adjacent pixel or region merging to the initial region, so as to gradually increase the area, until there is no point can merge or other small areas so far. The results of regional growth depend on the initial point - the selection of seed points, the growth criteria and the terminating conditions.

**Step 9 A small area in the image after the growth of the region**

Due to the interference of the ground texture or abnormal spot, there will be some smaller areas in the two value image after the growth of the area, which are mainly spotted or striped. These small areas will interfere with the subsequent processing. In this algorithm, we find the contour of all the regions in the image and calculate the area, and set the suitable area threshold. If the area of the area is less than the threshold, we will remove the area according to the idea of regional growth. The result is shown in Figure 3(c).

**Step 10 Transform the image into a ground area**

The whole growth area that contains the information of the obstacles is converted from the image plane to the ground, such as Figure 3(d). In order to use a single camera to calculate the distance in the three-dimensional coordinate system, it is assumed that all the obstacles are in contact with the ground. Figure 2(a) indicates that at a given camera height and pitching angle, any point on the plane is directly mapped to the ground relative to the camera position. By this method of coordinate conversion, the coordinates of the free point and the obstacle point in the world coordinate system can be determined in the image collected by the camera.

### 3.2. Fuzzy logic decision algorithms

The size of the image obtained after the above processing is \( m \times n \), the upper left corner pixel point is the coordinate origin, and the lower right corner pixel point coordinate is \( (m,n) \). A seed point \( S \) is selected on the vertical center line of the image. The closer the seed point is to the bottom of the image, the actual position of the point corresponds to the actual environment point closer to the front of the robot. A horizontal line is drawn in the image at the seed point position. From the seed point \( S \), the scanning lines are traversed leftward and rightward to count the number of pixels having a brightness value (free area), so as to determine whether the left side of the robot or the left side right path width. \( L \) and \( R \) denote the number of pixels on the left and the right of the seed point respectively, \( T \) is a threshold, and the judgment logic is

- IF \( L > T \) AND \( R > T \) THEN forward
- ELSEIF \( L > R \) THEN turn was left
- ELSE turn right.

In fact, the robustness of this decision-making method is poor, susceptible to misinterpretation. At the same time, it should be noted that the forward direction decision method is not suitable for directly counting the number of pixels with brightness values on the left and right sides of the vertical line in the image. Thus, fuzzy logic and membership are introduced to improve the decision-making method. The
two input variables are $x_L$ and $x_R$, which respectively represent the number of pixels traversed left and right along the scan line. The two output variables are the linear velocity command $v$ and the steering angle command $\theta$, which are used to control the movement of the robot. In order to increase the robustness of the algorithm, three seed points are taken perpendicular to the image, which are $S_1$, $S_2$ and $S_3$ respectively. Three corresponding horizontal scanning lines are drawn through seed points, and then the pixels on the scanning lines are respectively traversed. The number of pixels of the brightness value and calculate the average value.

\[
x_L = \frac{1}{3} \sum_{i=1}^{3} L_i \tag{8}
\]

\[
x_R = \frac{1}{3} \sum_{i=1}^{3} R_i \tag{9}
\]

Where $L_i$ is the number of pixels to the left of the seed point $S_i$ along the scan line and $R_i$ is the number of pixels to the right of the $S_i$ along the scan line of the seed point $S_i$.

Figure 4 shows the membership function of the input variables. The input variables are fuzzified with three membership functions of small ($S$), medium ($M$) and large ($L$). The membership function of the input variables is:

\[
\mu_{1x}(x) = \begin{cases} 
1, & 0 \leq x < a \\
\frac{c-x}{c-a}, & a \leq x < c \\
0, & \text{others}
\end{cases} \tag{10}
\]

\[
\mu_{2x}(x) = \begin{cases} 
\frac{x-b}{d-b}, & b \leq x < d \\
\frac{f-x}{f-d}, & d \leq x \leq f \\
0, & \text{others}
\end{cases} \tag{11}
\]
\[ \mu_L(x) = \begin{cases} \frac{x-e}{g-e}, & e \leq x < g \\ 1, & x \geq g \\ 0, & \text{others} \end{cases} \]

Where \( e \in \{x_L, x_R\} \).

Define fuzzy rules as: If \( x_L \) is \( U_1 \) and \( x_R \) is \( U_2 \) then \( y_1 \) is \( v \) and \( y_2 \) is \( \theta \). Where \( U_1, U_2 \in \{S, M, L\} \), \( v \in \{v_1, v_2, v_3\} \), \( \theta \in \{\theta_1, \theta_2, \theta_3\} \), where \( n \) is the total number of rules in the established rule base. The algorithm established the fuzzy rule base in Table 4.

| Order | output | input | \( \theta/° \) |
|-------|--------|-------|-------------|
| 1     | S      | S     | 0.9         | 0           |
| 2     | S      | M     | 0.5         | -30         |
| 3     | S      | L     | 0.1         | -45         |
| 4     | M      | S     | 0.5         | 30          |
| 5     | M      | M     | 0.9         | 0           |
| 6     | M      | L     | 0.5         | -30         |
| 7     | L      | S     | 0.1         | 45          |
| 8     | L      | M     | 0.5         | 30          |
| 9     | L      | L     | 0.9         | 0           |

In Table 4, the fuzzy output of the linear velocity \( v \) is a per unit value, the steering angle \( \theta \) is the steering angle of the mobile robot, the negative value represents to the right and the positive value to the left. De-obfuscation of two output variables can use weighted averaging.

4. Discussions

In order to verify the above method, we use C++ and OpenCV to write the algorithm. Wheeled mobile robot using a laptop computer to obtain video streaming from the USB camera, the algorithm processed the RS-232 serial port with control instructions. In order to reduce the computational complexity and ensure the real-time performance of the algorithm, the size of the captured image is set as 320 × 240 pixels. Figure 5 (a) and (b) show the robot reaching the end point from the starting point through the obstacle, and the robot knows the target position during planning of the local path in simple indoor environment. Figure 5 (c) is a binary image of the obstacle edge obtained by the robot in the initial stage. Figure 5 (d) for the robot to avoid obstacles in the process of visual effects. Figure 6 shows the results of mobile robot obstacle avoidance and path planning. It can be seen that the robot can successfully avoid the obstacle and reach the target position. In the experiment, the starting coordinate of the robot is (0, 0) and the coordinate of the center of the target position is set as (500, 200) cm. The distance between the robot and the edge of the obstacle in several key positions was recorded during the experiment. The actual distance was obtained by measurement. The calculated value of visual algorithm distance \( y_c \) was obtained by the formula.
5. CONCLUSION

In recent years, due to the continuous upgrading of computer and image processing technology, visual navigation has become one of the hot research fields in robot navigation. In this paper, a visual navigation algorithm for mobile robot is proposed. Edge detection is performed on the converted components by using the algorithm, and the detection results are synthesized. Filtering synthetic edges by threshold processing removes weak edge information and improves detection accuracy. The experimental results show that the conversion of image color space reduces the influence of surface reflection and shading. The algorithm can effectively eliminate the interference of ground stripes and detect the edge of obstacle accurately, and the fuzzy logic decision method improves the robustness and result of the algorithm the reliability.

Acknowledgment:
This paper was supported by the Characteristic Innovation Project of Guangdong province universities and colleges (project number: 2019GKTSCX049).

References
[1] DeSouza G N, Kak A C. (2002) Vision for mobile robot navigation: A survey.J. IEEE transactions on pattern analysis and machine intelligence, 24(2): 237-267.
[2] Tzafestas S G. (2018) Mobile robot control and navigation: A global overview. J . Journal of Intelligent & Robotic Systems, 91(1): 35-58.

[3] Obradovic D, Lenz H, Schupfner M. (2007) Fusion of sensor data in Siemens car navigation system. J . IEEE Transactions on Vehicular Technology, 56(1): 43-50.

[4] Woodman O J. (2007) An introduction to inertial navigation. R . University of Cambridge, Computer Laboratory.

[5] El-Sheimy N, Youssef A. (2020) Inertial sensors technologies for navigation applications: State of the art and future trends. J . Satellite Navigation, 1(1): 1-21.

[6] Lazanas A, Latombe J C. (1995) Landmark-based robot navigation. J . Algorithmica, 13(5): 472-501.

[7] Hile H, Grzeszczuk R, Liu A, et al. (2009) Landmark-based pedestrian navigation with enhanced spatial reasoning. In: International Conference on Pervasive Computing. Springer, Berlin, Heidelberg. 59-76.

[8] Xiong H, Tang J, Xu H, et al. (2017) A robust single GPS navigation and positioning algorithm based on strong tracking filtering. J . IEEE Sensors Journal,18(1): 290-298.

[9] Liu Y, Fan X, Lv C, et al. (2018) An innovative information fusion method with adaptive Kalman filter for integrated INS/GPS navigation of autonomous vehicles. J . Mechanical Systems and Signal Processing, 100: 605-616.

[10] Bonin-Font F, Ortiz A, Oliver G. Visual navigation for mobile robots: A survey[J]. Journal of intelligent and robotic systems, 2008, 53(3): 263-296.

[11] Zhu Y, Mottaghi R, Kolve E, et al. (2017) Target-driven visual navigation in indoor scenes using deep reinforcement learning. In: IEEE international conference on robotics and automation (ICRA). IEEE. 3357-3364.

[12] Riehle T H, Anderson S M, Lichter P A, et al. (2012) Indoor magnetic navigation for the blind. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE: 1972-1975.

[13] Gupta S, Davidson J, Levine S, et al. (2017) Cognitive mapping and planning for visual navigation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2616-2625.

[14] Guo L. (2018) Ground-constrained motion target tracking for monocular sequence images. J . International Journal of Pattern Recognition and Artificial Intelligence, 32(12): 1850044.

[15] Kidono K, Miura J, Shirai Y. (2002) Autonomous visual navigation of a mobile robot using a human-guided experience. J . Robotics and Autonomous Systems, 40(2-3): 121-130.