We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



18

Omni-directional vision sensor for mobile robot navigation based on particle filter

Zuoliang Cao, Yanbin Li and Shenghua Ye
Tianjin University of Technology
P.R. China

1. Introduction

Automatic Guided Vehicles (AGV) provide automated material movement for a variety of industries including the automobile, chemicals/ plastics, hospital, newspaper, commercial print, paper, food & beverage, pharmaceutical, warehouse and distribution center, and manufacturing industries. And they can reduce labor and material costs while improving safety and reducing product and equipment damage.

An AGV consists of one or more computer controlled wheel based load carriers (normally battery powered) that runs on the plant floor (or if outdoors on a paved area) without the need for an onboard operator or driver. AGVS have defined paths or areas within which or over which they can go. Navigation is achieved by any one of several means, including following a path defined by buried inductive wires, surface mounted magnetic or optical strips, or alternatively by way of visual guidance. (Crisan D, &Doucet A. 2002).

This chapter describes a total navigation solution for mobile robots. It enables a mobile robot to efficiently localize itself and navigate in a large man-made environment, which can be indoor, outdoor or a combination of both. For instance, the inside of a house, an entire university campus or even a small city lie in the possibilities.

Traditionally, other sensors except cameras are used for robot navigation, like GPS and laser scanners. Because GPS needs a direct line of sight to the satellites, it cannot be used indoors or in narrow city centre streets, i.e. the very conditions we foresee in our application. Time-of-flight laser scanners are widely applicable, but are expensive and voluminous, even when the scanning field is restricted to a horizontal plane. The latter only yields a poor world representation, with the risk of not detecting essential obstacles such as table tops. (Belviken E, &Acklam PJ. 2001)

That is why we aim at a vision-only solution to navigation. Vision is, in comparison with these other sensors, much more informative. Moreover, cameras are quite compact and increasingly cheap. We observe also that many biological species, in particular migratory birds, use mainly their visual sensors for navigation. We chose to use an omni-directional camera as visual sensor, because of its wide field of view and thus rich content of the images acquired with. Besides, we added a few artificial markers to the environment for navigation.

In a word, we present a novel visual navigation system for the AGV. With an omni-

directional camera as sensor and a DSP as processor, this system is able to recover distortion of a whole image due to fish-eye lens. It can recognize and track man-made landmarks robustly in a complex natural environment, then localize itself using the landmarks at each moment. The localization information is sent to a computer inside the robot and enables fast and simple path planning towards a specified goal. We developed a real-time visual servo technique to steer the system along the computed path.

The whole chapter falls into seven sections. Section I introduces various navigation strategies widely used in the mobile robot, and discusses the advantages and disadvantages of these navigation strategies firstly. Then a vision-only solution to navigation is proposed, with an omni-directional camara as visual sensor. This visual navigation strategy can guide the mobile robot to move along a planned path in a structured environment.

Section II illuminates the framework of the whole system, including the targeted application of this research, an automated guided vehicle (AGV). The two-color sequence landmarks are originated to locate and navigate the AGV. Besides, a visual image processing system based on DSP has been developed. The entire related distortion correction, tracking and localization algorithm are run in the built-in image processing system. This compact on board unit can be easily integrated into a variety of mobile devices and appears low power, well modularity and mobility.

Section III lists all sorts of fish-eye lens distortion models and sets forth a method of distortion correction. The involved fish-eye lens satisfies isometric projection, and the optical imaging center and a distortion parameter of the visual sensor need to be figured out in order to realize distortion correction. Thus the classical calibration for common lens paramenters is applied to the fish-eye lens, and those above parameters are figured out.

Section IV talks about the novel real-time visual tracking with Particle Filter, which yields an efficient localization of the robot, with focus on man-made landmarks. It is the key technology that make the whole system work. The original tracking algorithm uses the result of object recognition to validate the output of Particle Filter, which improves the robustness of tracking algorithm in complex environment.

Section V puts stress on the localization and navigation algorithm with the coordinates of the landmarks provided by particle filter. The location and the orientation of the AGV are worked out based on coordinate transformation, in the three-dimensional enviromnent rebuilt on the two-color sequence landmarks. Moreover, a PID control strategy is run in the built-in computer of the AGV for navigation.

Section VI presents the actual effect of the mobile robot navigation. The experimental environment and the experimental steps are introduced in detail, and six pictures of experiment results are shown and discussed.

Secion VII summarizes the work done about the research. A beacon tracker based on Particle Filter is implemented in the built-in image processing system. Real-time distortion correction and tracking algorithms are performed in the system, and the AGV is located and navigated with the tracking results of hte landmarks from the beacon tracker.

2. System framework

Our mobile robot platform is shown in Fig. 1. With our method, the only additional hardware required is a fish-eye lens camera and an embedded hardware module. The fish-eye lens is fixed on the top of the vehicle to get omni-directional vision, and the embedded

system based on DSP takes charge in distortion rectification, target recognition, target tracking and localization.



Fig. 1. Mobile robot platform

2.1 Mobile robot

Usually, a mobile robot is composed of the body, battery and charging system, drives, steering, precision parking device, motion controllers, communication devices, transfer system, and visual navigation subsystem and so on. The body includes the frame and the corresponding mechanical structures such as reduction gearbox, motors and wheels, etc, and it is a fundamental part of the AGV.

There are three working ways for an AGV:

(1) Automatic mode. When the AGV is set in automatic operation mode, the operator enters the appropriate command according to the plan path, and the AGV start to work in the unmanned mode;
(2) Semi-automatic mode. The operator can directly assist the AGV to complete its work through the buttons on the AGV;
(3) Manual mode. The operator can also use remote control trolley to move the AGV to the desired location manually.

The mobile robot platform of this chapter is a tracked AGV, and it consists of an embedded Industrial Personal Computer (IPC), motion control system, multiple infrared sensors and ultrasound sensors, network communication system and so on. The IPC uses industrial-grade embedded motherboard, including low-power, high-performance Pentium-M 1.8G CPU, SATA 160G HDD, DDR400 2G memory, six independent RS232 serial ports, eight separate USB2.0 interface. Moreover, four-channel real-time image acquisition card can be configured on this motherboard. The specific hardware modules of the mobile robot are shown in Fig. 2.

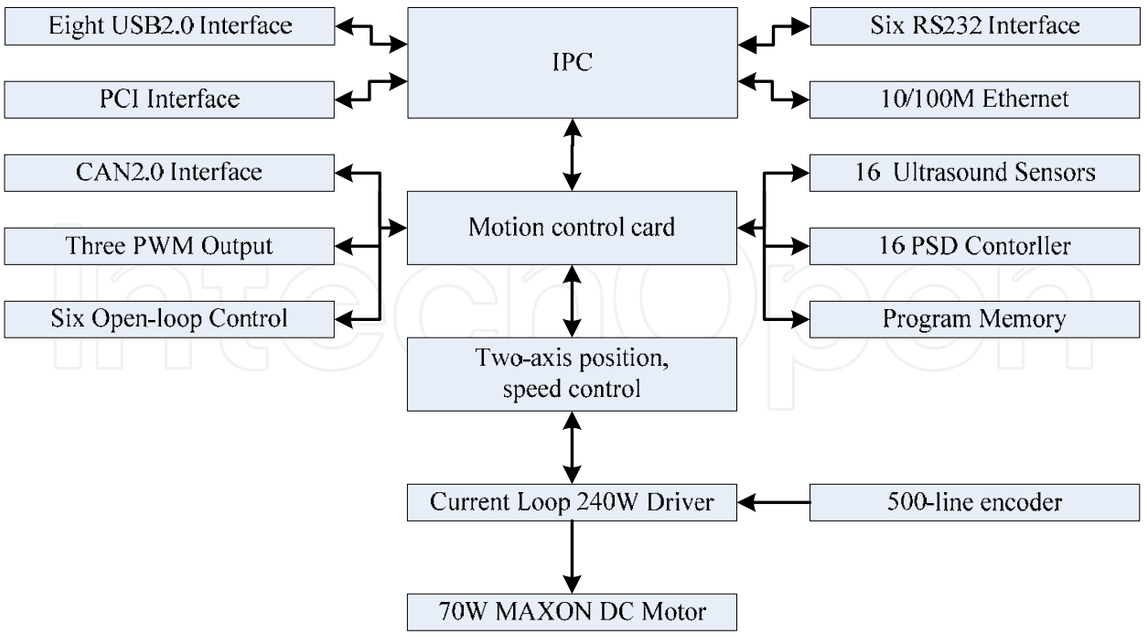


Fig. 2. Hardware modules of the mobile robot platform

2.2 Fish-eye lens

The project research needs high-quality visual image information, so that the camera features color, planar array, large-resolution and CCD light-sensitive device. We adopt imported Japanese ultra-wide angle fish-eye lens Fujinon FE185C046HA-1 as well as analog color CCD camera Watec221S, as shown in Fig. 3.



Fig. 3. Fish-eye lens camera

The performance parameters of the fish-eye lens are shown in Table 1. And the CCD size is 1/2 inches, PAL stardard, and the resolution (horizontal) is 50 lines (Y/C 480 lines). Its effective pixel (K) is P440K, minimum illumination is 0.1Lux, and lens mount method is CS installation, with operating voltage of DC +12V.

Focus(mm)	1.4
Aperture Range	F1.4-F16
CCD size	1/2″
Minimum object distance(m)	0.1
BFL(mm)	9.70
Interface	C
Weight(g)	150

Table 1. The performance parameters of the fish-eye lens

2.3 Embedded hardware platform

The vast majority of image processing systems used currently by the AGV are based on the traditional PC or high-performance IPC. Although the PC-based architecture is simple and mature technically, and applied widely, these image processing systems are redundant in terms of both resource allocation and volume, besides they have poor flexibility, heavy weight and high power consumption, which is not suitable for the mobile vehicle system application.

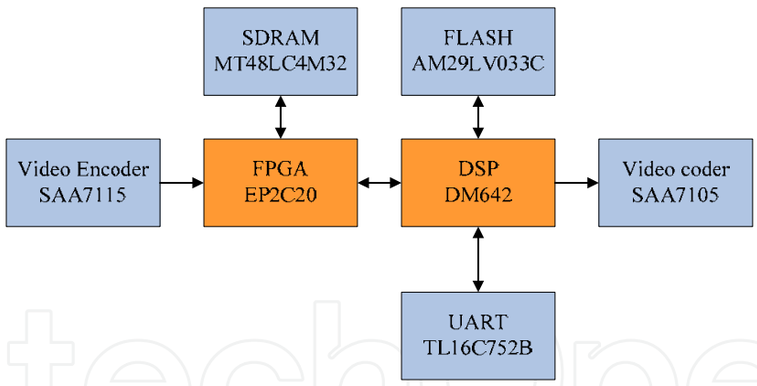


Fig. 4. System diagram of hardware platform

While the embedded system, as a highly-integrated application platform, features great practicability, low cost, small size, easy expansion and low power consumption. Therefore, we drew on the current successful application of DSP and FPGA chips in the multimedia processing, considering the characteristics of the vehicle image processing system such as large computation, high real-time requirement and limited resources, proposed a solution to build an embedded hardware image processor based on FPGA+DSP. And target recognition, tracking and localization are achieved on this hardware platform. The system diagram of the hardware platform is shown in Fig. 4.

We use Altera's Cyclone series FPGA EP2C20 and TI's DaVinci DSP TM320DM642 to build the embedded hardware image processor. Firstly, the input analog video signal goes

through clamp circuit, anti-aliasing filter, A/D conversion and YUV separation circuit to be converted to BT.656 video data stream. Secondly, the rich internal hardware resource of the FPGA is used to achieve image capture, color space conversion, image enhancement and distortion correction and so on. Thirdly, other advanced image processing algorithm such as target recognition, tracking and localization, are implemented in the DSP with its high-speed signal processing capability. Finally, the result of image processing is output to the SAA7105, which converts it into NTSC or PAL television signal and displays it on a liquid crystal screen. In a word, the entire system fully plays their respective advantages of the DSP and FPGA, and combines them closely to form a dedicated embedded hardware image processor.

In addition, we equip the hardware image processor with high-speed, large-capacity data memory and program memory, in order to meet the requirements of high-speed image processing and transmission.

2.4 Two-color sequential landmarks

The target recognition algorithm in this project belongs to the template matching method, and the invariant features of the target are the research focus of feature extraction. In an Omni-directional vision system built on fish-eye lens, because there is a serious distortion due to the fish-eye lens itself, it's difficult to find good invariant features. A large number of research and experiments show that color-based features have little change in a fish-eye image, but the remaining features such as widely used shape and corners, have obvious change due to the fish-eye lens distortion, and are not suitable as the target characteristics.

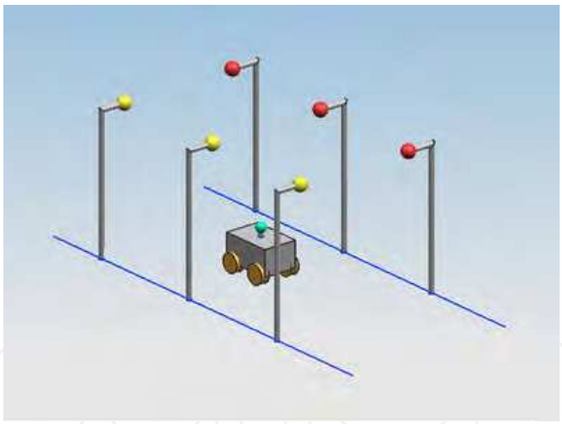


Fig. 5. Two-color sequential landmarks

In theory, considering the large filed of view of the fish-eye lens, using a set of two-color landmark can realize the precise position and navigation. But in fact, when the vehicle goes far away from the landmarks, the landmarks will be located in the image edge of the fish-eye lens, where there is a serious distortion. As a result, not only the landmarks become very small, but also the isometric projection imaging model is no longer applicable, which make the landmarks cannot be used to locate the AGV at this time. Therefore, we arrange equidistant two-color landmarks sequentially along the AGV path as Fig. 5 shows, when a group of landmarks doesn't meet the requirement of imaging model, it will automatically

switch to the next group of landmarks in order to achieve continuous landmark recognition and tracking for the next localization and navigation of the AGV.
In a word, this landmark mode has the topological features close to the natural scenery in indoor and outdoor environment, is simple and practical, easy to layout and maintain, which can effectively improve the efficiency of automatic recognition and tracking in a large scene image with distortion of the fish-eye lens.

3. Fish-eye lens distortion model

Ordinary optical imaging system follows the conventional guideline——similarity, that is, object point is always similar to image point. Optical design tries to ensure this similarity, so does distortion rectification. As a result, come the following imaging formula: (Doucet A, &Godsil S, &Andrieu C. 2000)

When the object is near: $y_0^{'} = \beta y$

When the object is infinitely far: $y_0^{'} = f \tan \omega$

Where y is the height of the object, $y_0^{'}$ is the height of its image, β is lateral magnification, f is the focus of the optical system, ω is half-field angle. But when $\tan 90^{\circ} = \infty$, similarity can not be met.
In order to cover a wider field angle with a single visual sensor, image compression and deformation is introduced, just as the curves in Fig.6 shows.

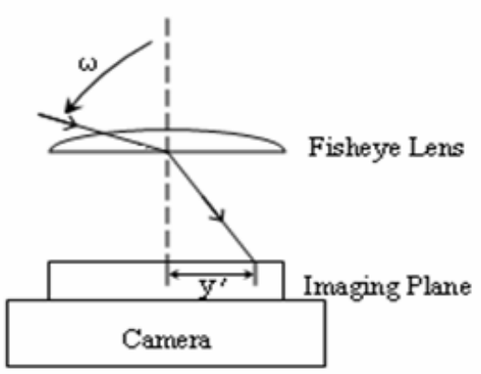


Fig. 6. fish-eye lens distortion

They make the image to achieve deformation to a certain extent, in order to ensure that the solid angle covers expected object space. In addition, the distortion of optical system is determined by the pathway of primary optical axis, so it just causes image deformation, but doesn't affect image clarity. Despite the obvious distortion, as far as mathematics is considered, there is still one-to-one correspondence between the object and its image, which ensures the correctness and feasibility of the non-similar imaging model. (Fox D. 2003)
Our fisheye lens satisfies isometric projection, the imaging formula is as follows: (Thrun S, &Fox D, &Burgard W. 2001)

$$y' = kf\omega \qquad (0 < k < 1) \tag{1}$$

Where y′ is the height of its image, f is the focus of the optical system, ω is half-field angle.

And $$y' = X^2 + Y^2 \quad \tan\omega = \frac{\sqrt{\left(x_c^2 + y_c^2\right)}}{z_c}$$

$$\begin{cases} u - u_0 = X/d_x \\ v - v_0 = Y/d_y \end{cases} \tag{2}$$

Where (X, Y) is the image coordinate, (xc, yc, zc) is the camera coordinate, (u, v) is the pixel coordinate, and (u0, v0) is the center of the image. According to the equation (2), if (u0, v0) and k are figured out, the conversion relation between the image coordinate and the world coordinate will be fixed, and the rectification of the fisheye image will be accomplished. In a word, the rectification procedure can be divided into two main steps. (Gilks W R, &Berzuini C. 2001)

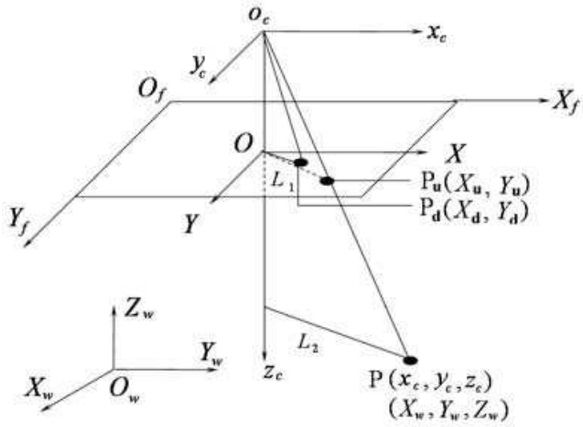


Fig. 7. coordinate system

Firstly, the conversion relation between the camera coordinate (Xc, Oc, Yc) and the world coordinate (Xw, Ow, Yw) should be figured out. As shown in Fig.7, the relation can be expressed as:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = R\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + T = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad \text{or}$$

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3}$$

Where R is the rotation matrix, and T is the translation matrix.



Fig. 8. circular points

The world coordinates of those circular points in Fig.8 are known in advance, if we calculate the image coordinates of those points by extracting their centers, the R and T can be figured out with the equation (3).
Secondly, the coincidence relation between the camera coordinate (Xc, Oc, Yc) and the image coordinate (X, O, Y) should be figured out with the equation (3). (Jung Uk Cho, &Seung Hun Jin. 2006)
Finally, we use 30 circular points to calculate these above equation in the fish-eye distortion rectification, and part of the coordinate data is presented in Table 2. With these equations, the conversion relation between the world coordinate and the image coordinate is figured out, and the center of the image (u0, v0) is (360,288), k = 1.5. Using these parameters and equations, we can implement real-time distortion rectification in DSP.

The world coordinate			The image coordinate	
X	Y	Z	U	V
2026.189	-7127.78	-562.917	298.21	20.87
1756.365	-7141.03	-559.438	578.32	21.32
2174.074	-7118.34	-693.84	143.34	162.08
1900.543	-7133.42	-686.855	433.87	132.65
2028.248	-7124.78	-844.925	281.93	324.59
1759.955	-7138.48	-825.114	601.8	293.74
1970.002	-7121.13	-773.602	348.47	234.49

Table 2. experiment data

4. Improved particle filter

To localize the AGV with landmarks, the coordinate of the beacons in an image should be figured out firstly. So an improved particle filter is employed to track these beacons and get their coordinates.

As an algorithm framework, a particle filter can be used to track multiple objects in the case of nonlinear and non-Gaussian problem. And the object is quite flexible, like man-made or natural landmarks. Particle filter is a Monte Carlo sampling approach to Bayesian filtering. The main idea of the particle filter is that the posterior density is approximated by a set of discrete samples with associated weights. These discrete samples are called particles which describe possible instantiations of the state of the system. As a consequence, the distribution over the location of the tracking object is represented by the multiple discrete particles. (Kwok C, &Fox D, &Meil M. 2004)

In the Bayes filtering, the posterior distribution is iteratively updated over the current state Xt, given all observations Zt = {Z1,..,Zt} up to and including time t, as follows:

$$p(x_t \mid Z_t) = kp(Z_t \mid X_t) \bullet \int_{X_{t-1}} p(X_t \mid X_{t-1}) p(X_{t-1} \mid Z_{t-1}) dx_{t-1} \tag{4}$$

Where p(Zt|Xt) expresses the observation model which specifies the likelihood of an object being in a specific state and p(Xt|Xt-1) is the transition model which specifies how objects move between frames. In a particle filter, prior distribution p(Xt-1|Zt-1) is approximated recursively as a set of N weighted samples , which is the weight for particle . Based on the Monte Carlo approximation of the integral, we can get:

$$p(X_t \mid Z_t) \approx kp(Z_t \mid X_t) \sum_{i=1}^{N} w_{t-1}^{(i)} p(X_t \mid X_{t-1}^{(i)}) \tag{5}$$

Particle filter provides robust tracking of moving objects in a cluttered environment. However, these objects have to be specified manually, and then the particle filter can track them, which is unacceptable for autonomous navigation. Thus, we combine an object recognition algorithm with particle filter, and use the object recognition algorithm to specify landmarks automatically. And once particle filter fails to track the landmarks occasionally, the object recognition algorithm will function to relocate the landmarks. In order to facilitate object recognition, the landmarks are painted certain colors. Based on color histogram, we can find out these landmarks easily.

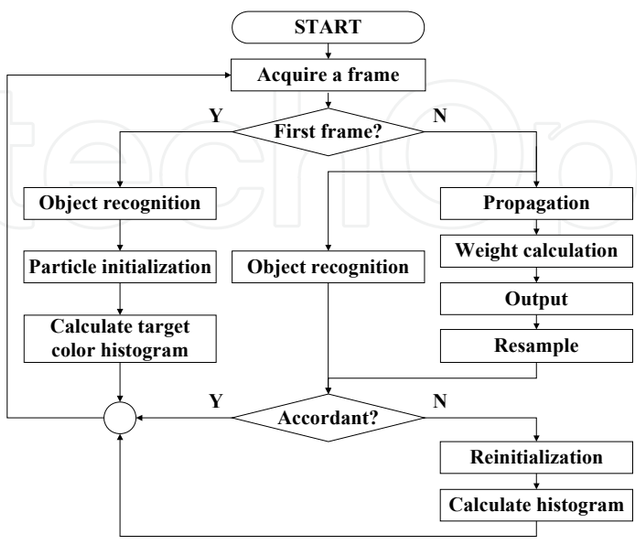


Fig. 9. flow chart of improved Particle Filter

The flow chat of improved PF is shown in Fig.9. When started, a frame is acquired and if it's the first frame, the initialization will be operated. Firstly, the target is recognized by searching in the whole image area based on color histogram, and its location is figured out for next step. Then, the initial particles come out randomly around the above location, and the target color histogram is calculated and saved. Thus, the initialization is finished.
At the next frame, a classical particle filter is carried out. The principal steps in the particle filter algorithm include:

1) Initialization

Draw a set of particles for the prior $p(X_0)$ to obtain $\{X_0^{(i)}, w_0^{(i)}\}_{i=1}^{N}$.

2) Propagation and Weight calculation

a) For i=1,…,N, sample $X_k^{(i)}$ from the probability distribution $p(X_k^{(i)} \mid X_{k-1}^{(i)})$.

b) Evaluate the weights $w_k^{(i)} = p(Z_k \mid X_k^{(i)})$, i=1,…,N

c) Normalize the weights $w_k^{(i)} = \frac{w_k^{(i)}}{\sum_{j=1}^{N} w_k^{(j)}}$, i=1,…,N

3) Output

Output a set of particles $\{X_k^{(i)}, w_k^{(i)}\}_{i=1}^{N}$ that can be used to approximate the posterior distribution as $p(X_k \mid Z^k) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(X_k - X_k^{(i)})$ and the estimate as $E_{p(g|Z^k)}(f_k(X_k)) \approx \sum_{i=1}^{N} w_k^{(i)} f_k(X_k^{(i)})$, where $\delta(g)$ is the Dirac delta function.

4) Resample

Resample particles $X_k^{(i)}$ with probability $w_t^{(i)}$ to obtain N independent and identically distributed random particles $X_k^{(j)}$, approximately distributed according to $P(X_k \mid Z^k)$.

5) K=k+1, go to step2.

Meanwhile, object recognition is executed and its result is compared with the particle filter. If the recognition result is not accordant with the particle filter, which means that the particle filter fails, the PF will be reinitialized based on the recognition result. According to the combination of object recognition and particle filter, the improved particle filter can keep tracking the target even though the target is blocked or disappeared for a while.

5. Localization and navigation

5.1 Localization

With the coordinates of the landmarks provided by particle filter, we can locate and navigate the AGV automatically.

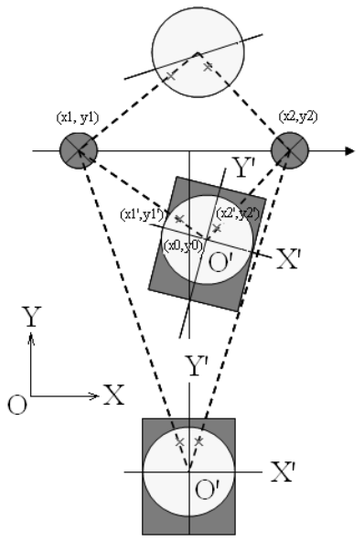


Fig. 10. coordinate conversion

As Fig.10 shows, the world coordinates of the two beacons are (x1, y1) and (x2, y2), and their relative camera coordinates are (x'1, y'1) and (x'2, y'2). In order to steer the AGV, we need to figure out the position of the AGV, (x0, y0) and the angle between the world coordinate and the camera coordinate, θ. According to the coordinate conversion principle, the following equations exist:

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = \begin{bmatrix} -\cos\theta & \sin\theta & x_1 \\ -\sin\theta & -\cos\theta & y_1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix}$$

$$\theta = arctg\frac{y_2 - y_1}{x_2 - x_1} - arctg\frac{y'_2 - y'_1}{x'_2 - x'_1}$$

$$\begin{cases} x_0 = x_1 - kx''_1\cos\theta + ky''_1\sin\theta \\ y_0 = y_1 - kx''_1\sin\theta - ky''_1\cos\theta \\ \theta = arctg\dfrac{y_2 - y_1}{x_2 - x_1} - arctg\dfrac{y''_2 - y''_1}{x''_2 - x''_1} \end{cases} \quad (6)$$

Apparently, we can get (x0, y0) and θ with the above equations and realize real-time location in DSP. Then the position and angle information of the AGV is sent to the built-in computer for navigation.

5.2 Navigation

An incremental PID algorithm is employed as navigation algorithm, which can be expressed as the following equation:

$$\begin{aligned}\Delta u(k) &= u(k) - u(k-1) \\ &= (K_p + K_i + K_d)e(k) - (K_p + 2K_d)e(k-1) + K_d e(k-2)\end{aligned} \tag{7}$$

Suppose $A = (K_p + K_i + K_d), B = -(K_p + 2K_d), C = K_d$, we can get

$$\Delta u(k) == Ae(k) + Be(k-1) + Ce(k-2) \tag{8}$$

Equation (8) shows that we can get each speed increment with *e(k),e(k-1),e(k-2)* in the calculation of *u(k)*, as shown in equation (9).

$$\begin{aligned}SpeedL &= Speed - \Delta u(k) \\ SpeedR &= Speed + \Delta u(k) \\ e(k) &= K_1\theta(k) + K_2 r(k)\end{aligned} \tag{9}$$

Where *SpeedL* and *SpeedR* are the initial velocities of the left and right wheels respectively, the velocity is 0.6m/s. And *K1* and *K2* are the weight of angle deviation and position deviation respectively. We can use equation (9) to figure out the velocities of both wheels, and realize the AGV navigation finally.

6. Navigation experiments

6.1 Experimental conditions



Fig. 11. experiment environment

The experimental environment of the Omni-directional vision sensor for AGV navigation is shown in Fig. 11. The tracked robot is placed in the corridor of a teaching building in this experiment, and the landmarks are fixed on the top of the corridor. Apparently, there are windows, radiator, fire hydrants and other various interferences around the AGV path, so that we can test the robustness of the Omni-directional vision sensor for AGV navigation.



Fig. 12. fish-eye lens camera and embedded image processor

As Fig.12 shows, the fish-eye lens camera is fixed on top of the AGV with a steel frame to ensure that the lens be placed vertically upward. The fish-eye lens is 0.88m above the ground, and it will acquire Omni-directional image of the entire hemisphere domain above the AGV. While the embedded image processor is mounted in a black box, and the box is fixed to the roof of the AGV. Moreover, the black box sets aside hatches for a variety of processor interfaces, and facilitates thermal dispersal of the circuit board with the grid structure on the top of the box.

6.2 Experimental principle

Firstly, the Omni-directional vision sensor based on fish-eye lens outputs standard analog video signal in PAL format, with image resolution of 720 x 576 and frame rate of 25 fps. Secondly, the captured image signal is transmitted to the embedded image processor through a standard video interface, and the DSP on board runs composite particle filter tracking algorithm to achieve real-time recognition and tracking of the two-color sequential landmarks. Thirdly, we use the result of landmark tracking to calculate the position deviation and angle deviation of the AGV path relative to the target path, according to the localization algorithm based on two-color landmarks. Finally, the localization result is sent to the built-in IPC of the AGV as the input of PID control strategy with serial port, and we can change both wheels' speed of the AGV to achieve steering control, and realize the autonomous navigation of the AGV ultimately.

6.3 Experimental result

In accordance with the above experimental principle, we obtained two AGV navigation paths as shown in Fig.13. The blue curve is the target path pre-generated with the two-color sequential landmarks, and the red curve is the AGV navigation path employing angle

deviation as PID control input, while the yellow curve is the AGV navigation path employing both angle deviation and position deviation as PID control input.

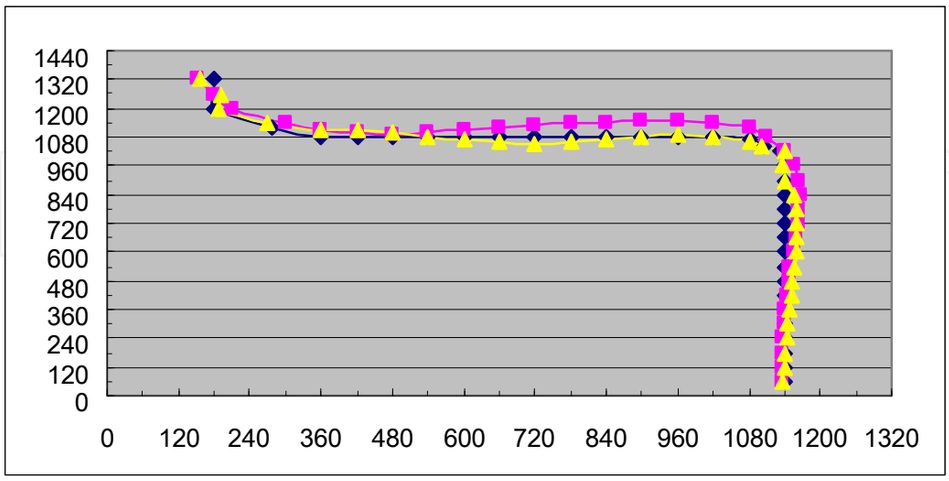


Fig. 13. experimental result (units: cm)

7. Analysis and Conclusion

The autonomous navigation of the AGV can be realized under the control of both angel deviation and position deviation. Experiment shows that the Omni-directional vision sensor can achieve the autonomous navigation of the AGV at a speed of 0.6m/s, and the absolute tracking accuracy is about 10 cm.

In conclusion, Dynamic localization employs a beacon tracker to follow the landmarks in real time during the arbitrary movement of the vehicle. The coordinate transformation is devised for path programming based on time sequence images analysis. The beacon recognition and tracking a key procedure for an omni-vision guided mobile unit. The conventional image processing such as shape decomposition, description, matching, and other usually employed techniques are not directly applicable in omni-vision. PF has been shown to be successful for several nonlinear estimation problems. A beacon tracker based on Particle Filter which offers a probabilistic framework for dynamic state estimation in visual tracking has been developed. In a word, the Omni-directional vision sensor implements the autonomous navigation of the AGV in the structured indoor and outdoor environment.

8. References

Belviken E, &Acklam PJ. (2001). Monte Carlo filters for non-linear state estimation. Automation, 2001, 37(1): pp.177-183.

Crisan D, &Doucet A. (2002). A survey of convergence results on particle filtering methods for practitioners. IEEE Transactions on Signal Processing, 2002, 50(2):pp.736-746.

Doucet A, &Godsil S, &Andrieu C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and Computing, 2000, 10(3):pp.197-208.

Fox D. (2003). Adapting the sample size in particle filters through KLD sampling. The International Journal of Robotics Research, 2003, 22(12):pp 3-5.

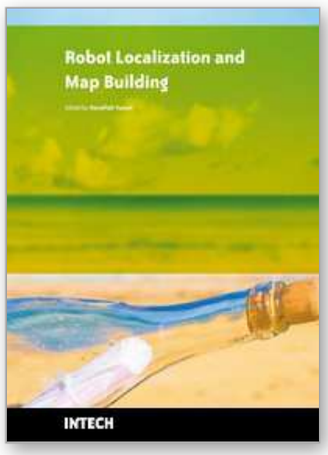
Gilks W R, &Berzuini C. (2001). Following a moving target Monte Carlo inference for dynamic Bayesian models. Journal of the Royal Statistical Society, Series B, 2001, 61(1):pp.127-146.

Jung Uk Cho, &Seung Hun Jin. (2006). A real-time object tracking system using a particle filter. international conference on intelligent robots and systems October 9-15, 2006, Beijing, China.

Kwok C, &Fox D, &Meil M. (2004). A real-time particle filters. Proceedings of the IEEE, 2004, 92(3):pp.469-484.

Thrun S, &Fox D, &Burgard W. (2001). Robust Monte Carlo localization for mobile robots. Artificial Intelligence, 2001, 128(1-2):pp.99-141.

IntechOpen



Robot Localization and Map Building

Edited by Hanafiah Yussof

ISBN 978-953-7619-83-1
Hard cover, 578 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Localization and mapping are the essence of successful navigation in mobile platform technology. Localization is a fundamental task in order to achieve high levels of autonomy in robot navigation and robustness in vehicle positioning. Robot localization and mapping is commonly related to cartography, combining science, technique and computation to build a trajectory map that reality can be modelled in ways that communicate spatial information effectively. This book describes comprehensive introduction, theories and applications related to localization, positioning and map building in mobile robot and autonomous vehicle platforms. It is organized in twenty seven chapters. Each chapter is rich with different degrees of details and approaches, supported by unique and actual resources that make it possible for readers to explore and learn the up to date knowledge in robot navigation technology. Understanding the theory and principles described in this book requires a multidisciplinary background of robotics, nonlinear system, sensor network, network engineering, computer science, physics, etc.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Zuoliang Cao, Yanbin Li and Shenghua Ye (2010). Omni-Directional Vision Sensor for Mobile Robot Navigation Based on Particle Filter, Robot Localization and Map Building, Hanafiah Yussof (Ed.), ISBN: 978-953-7619-83-1, InTech, Available from: http://www.intechopen.com/books/robot-localization-and-map-building/omni-directional-vision-sensor-for-mobile-robot-navigation-based-on-particle-filter

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen