Efficient Neural Network Approach of Self-Localization for Humanoid Robot

Shih-Hung Chang, Wei-Hsuan Chang, Chih-Hsien Hsia, Fun Ye, and Jen-Shiun Chiang
Department of Electrical Engineering, Tamkang University, Taipei, Taiwan
E-mail: 696450682@s96.tku.edu.tw; whchang@ee.tku.edu.tw; chhsia@ee.tku.edu.tw; fyee@mail.tku.edu.tw; chiang@ee.tku.edu.tw

Abstract

Robot soccer game is one of the significant and interesting areas among most of the autonomous robotic researches. Following the humanoid soccer robot basic movement and strategy actions, the robot is operated in a dynamic and unpredictable contest environment and must recognize the position of itself in the field all the time. Therefore, the localization system of the soccer robot becomes the key technology to improve the performance. This work proposes efficient approaches for humanoid robot and uses one landmark to accomplish the self-localization. This localization mechanism integrates the information from the pan/tilt motors and a single camera on the robot head together with the artificial neural network technique to adaptively adjust the humanoid robot position. The neural network approach can improve the precision of the localization. The experimental results indicate that the average accuracy ratio is 88.5% under frame rate of 15 frames per second (fps), and the average error for the distance between the actual position and the measured position of the object is 6.68cm.

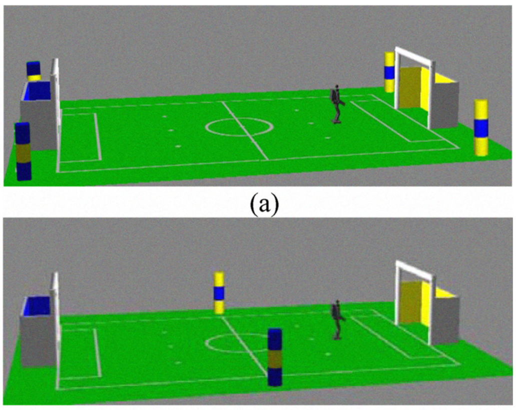
Keywords: Self-Localization, Humanoid Soccer Robot, Precision, Neural Network, Accuracy Ration.

1. Introduction

A good self-localization system cannot only make a robot acquire the information quickly and accurately in the whole field, but also makes an appropriate decision correspondingly. For easy manipulation we can preset all the locations in the field as a Cartesian coordinate system, and the robot will self-localize itself by the coordinate system. In recent years, the competition fields of RoboCup [1] and FIRA Cup [2] become more and more conformed to human environments. Fig. 1 shows the RoboCup soccer fields for humanoid kid-size of 2007 [3], 2008 [4], and 2009 [5], respectively. The landmarks decrease from four to two (2007-2009). In other words, the reference checkmarks for self-localization becomes less and less, and how to use less landmarks and increase the degree of accuracy becomes important issues [6]-[7].

Basically there are three types of techniques for robot localization [8]. The first approach is based on the stereo vision. This approach can obtain a lot of information, however the distance between the target and the camera is not accurate [9] and may reduce the accuracy of localization. The second one is based on the omni-directional vision. Although this method obtains better features, the omni-directional device causes geometry distortions to the perceived scene [10]. The third one uses the monocular vision technique. It must have robust features within a specific region [11]. This work proposes a visual self-localization approach that uses a single CCD camera and pan/tilt motors on the robot head to find the robust features and to analyze the environmental information for the RoboCup soccer field of 2009 [5].

The rest of this paper is organized as follows: Section 2 presents the general localization methods and encountered problems. The proposed self-localization mechanism is described in Section 3, and the experimental results are shown in Section 4. Finally, Section 5 gives a brief conclusion.



(a)

(b)

Figure 1. The configuration of RoboCup soccer field for humanoid kid-size. (a) for 2007, (b) for 2008 and 2009.

2. Robotic Vision Based Localization

The issue of the localization for humanoid robot focuses on analyzing the probable position by itself on the field. The key technology of the self-localization for the robot is how to take the advantages of the information of various sensors to match the position of the robot. Because the perceivable ability of the robot is restricted and the ambient environment is with enormous interferences, it is difficult to make the robot have efficient and more robust localization. During localizing, owing to the restrictions of the performance of various sensors and the interferences of the outside environment, it may have uncertainties for the orientation. The main factors are: 1) the dynamic variance for the outside environment; 2) the undependable information for the outside sensors (CCD camera, electronic compass, gyroscope, …, etc); 3) the deviation of the inside sensors (the pan/tilt sever motors, stepper motors, …, etc). These non-ideal elements lead to reduce the localization precision. To solve the mentioned non-ideal factors many researches tried to find better ways to model the environments and mathematic tools for simulation [12]-[13]. This paper proposes an efficient mechanism to improve the orientation precision. Therefore the humanoid robot can recognize its position explicitly on the field, and further it can proceed to the following soccer ball tracking and strategic planning.

3. The Proposed Approach

In this section, the efficient self-localization approach for humanoid robot is proposed. The main issues are focused on the robot vision module. Together with the image processing and trigonometric theorem, the humanoid robot can find the rough positions by itself. Later on, the proposed approach can help to increase the accuracy of the position. The proposed visual self-localization approach has five steps, and the flow chart of the self-localization mechanism is shown in Fig. 2. The details of the self-localization approach are described in the following five subsections.

3.1. Establishment of the Coordinate System

If the coordinate of a geometric map is available, it will be convenient to retain a lot of information in the whole field. For easy manipulation of the self-localization of a robot, the coordinate system of the field must be established in advance. In this work, before processing the localization, we must establish two appropriate coordinate systems. One is called "absolute coordinate system" on the field, and the other is called "relative coordinate system" in the image. There are four steps to establish the absolute coordinate system: 1) to estimate the sizes of the field and robot; 2) to find the interested position in the soccer field; 3) according to the proportion of the robot in the field to adjust the value in each block; 4) dividing the field into several blocks with the same size and assigning the interested position as the center block. The relative coordinate system will store the interesting information of the objects. Through these coordinate systems, the location of the robot, landmark, and goal can be located explicitly.

3.2. Landmark Detection

In order to the catch a stable feature, we treat the landmark as the feature for localization. In the initialization of the orientation, the robot keeps searching the landmark until finding it. After finding the landmark, the system will take the interested feature by converting the image from RGB to HIS space. In order to remove the influence of brightness, it takes the HS space only. Finally, it will mark the upper left (X_1,Y_1), upper right (X_2,Y_2), lower left (X_3,Y_3), lower right (X_4,Y_4), and center (X_C, Y_C) for the landmark in the image, as shown in Fig. 3.

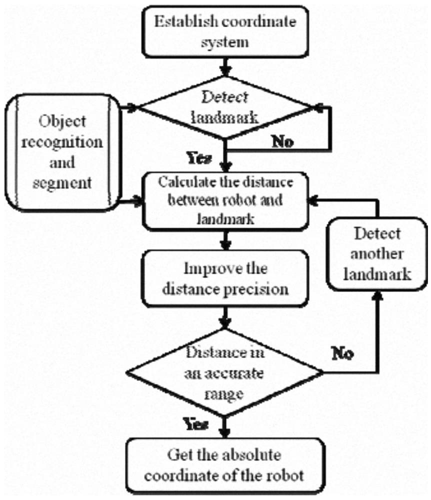


Figure 2. Robot self-localization and object ball localization flowchart.

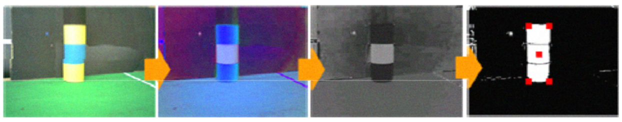


Figure 3. The process to mark the upper left, upper right, lower left, lower right, and center of the landmark.

We must adjust the feature to an appropriate position in the picture frame for self-localization. For the adjustment, the robot head keeps rotating horizontally but still vertically, and (1) can help to search the feature of the landmark:

$$\begin{cases} Xcb = Xca + P, & if \quad Xca < 150 \\ Xcb = Xca - P, & if \quad Xca > 170 \end{cases} \tag{1}$$

where X_{ca} is a pixel value for the horizontal direction, X_{cb} the pixel value for the next image, and P a change pixel value as the robot head moves. The robot head will not stop moving horizontally until X_C falls within 150-170 of horizontal pixels. Then (2) is used to analyze the feature of the landmark:

$$\begin{cases} Y3b = Y3a + Q, & \text{if } Y3a < 110 \\ Y3b = Y3a - Q, & \text{if } Y3a > 130 \end{cases} \tag{2}$$

where Y_{3a} is a pixel value for the vertical direction, Y_{3b} the pixel value for the next frame, and Q a change pixel value as the robot head moves vertically.

If X_C is within 150-170 horizontal pixels and Y_3 is within 110-130 vertical pixels in the frame, the critical feature information including boundary points and size can be found. By this approach, it can obtain the head pan/tilt angle of the robot. The robot forbids walking at this moment until it loses the feature information and then it terminates the self-localization procedure as shown in Fig. 4.

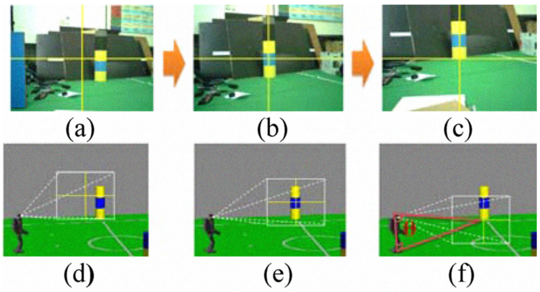


Figure 4. The images recognized by the robot through the CCD camera. (a) - (c) show the procedures of the robot to search the characteristic point and move the vision angle toward the object. (d) - (f) show how the robot head and CCD camera move.

3.3. Calculating the Distance between the Robot and Landmark

After obtaining a better feature, the distance between the robot and feature can be found by the following approach. At this moment, two data are obtained: the specific angle "θ" of the robot head and the height "h" of the robot. According to Fig. 5 and the trigonometric theorem, we can find the distance r as follows:

$$r = h \times \tan\theta \tag{3}$$

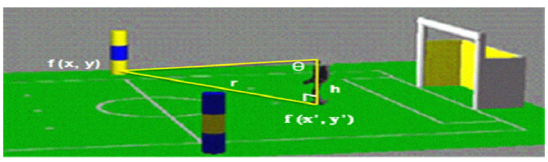


Figure 5. The relationship of r, θ, h, $f(x, y)$, and $f(x', y')$ between the robot and landmark.

Because the tangent angle has serious variance near $k\pi+\pi/2$, as shown in Fig.6, the distance r between the robot and landmark is not accurate. In order to find a more accurate r we propose an approach by using the technique of artificial neural network to find the distance r, and the detail of this approach is described in the following subsection.

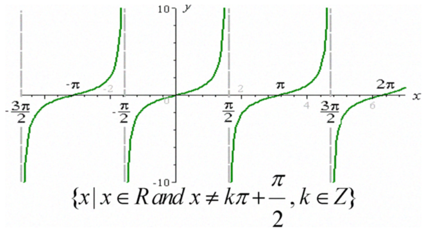


Figure 6. Graph of $y = \tan x$.

3.4. Improvement of the Distance Precision

In the localization system, if we want to analyze the information of the interesting features and the distance exactly, we must model the visual system by mathematics. However the visual localization system is complex and non-linear, for simplicity the neural network technique can be applied. By the neural network approach, we need not know the exact mathematic model of the visual system by simply replacing the mathematic model by the neurons, and we can still get the information of the interesting features and distance [15]. So far several neural networks have been proposed, such as back propagation neural (BPN) network, self-organizing neural network,…, etc. Here we use the technique of BPN network to find a more accurate distance between the robot and landmark.

3.4.1. Back Propagation Neural Network

The mechanism of the BPN network belongs to multilayer feed-forward networks and uses supervised learning. The multilayer feed-forward network approach deals with the non-linear relationships between the input and output, and the supervised learning can correct the values of the relationships. Because of these network structures, the BPN network has the advantages for higher learning precision and fast recall speed, and therefore the BPN becomes the most popular neural network module nowadays [15]. The block diagram of the BPN network is shown in Fig. 7.

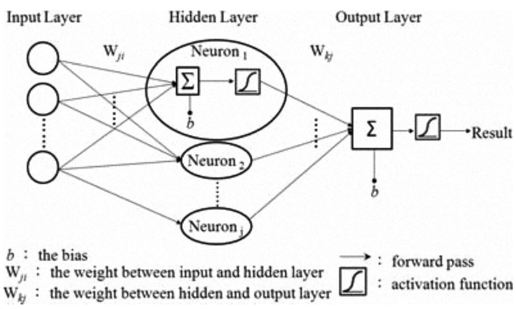


Figure 7. The BPN network method.

The basic element of a BPN network is the processing node. Each processing node behaves like a biological neuron and performs functions. It sums the values of its inputs, and this sum is then passed through an activation function to generate an output. Any differentiable function can be used as the activation function, f. All the processing nodes are arranged into layers and are fully interconnected to the following layers. There is no interconnection between the nodes of the same layer. In a BPN network, there is an input layer that acts as a distribution structure for the data presented to the network, and this layer is not used for any type of processing. One or more processing layers, called hidden layer, will follow this layer; the final processing layer is called the output layer.

3.4.2. The BPN Network for Humanoid Robot Localization

There are seven steps to improve the distance precision by the BPN network, and the procedures are shown in Fig. 8 [15].

Step 1: Prepare robust information including the interesting features of X_C, Y_C , and size, …, etc. In the picture frame, it sets the expectable distance value as the objective function and then normalizes these data to the appropriate values. The appropriate normalization is referred to the activation function f as follows:

$$y_j^n = f(net_j^n). \tag{4}$$

where y_j^n is the output value of the nth layer, and it is also the input value of the first layer. net_j^n is the weight accumulative value for the output value of the (n-1)th layer and is represented as follows:

$$net_j^n = \sum_i^n w_{ji}^n y_i^{n-1} + b_j^n. \tag{5}$$

where w_{ji}^n is the weighted connections between the jth neuron in the nth layer and the ith neuron in the (n-1)th layer, and b_j^n is the bias of the jth neuron in the nth layer.

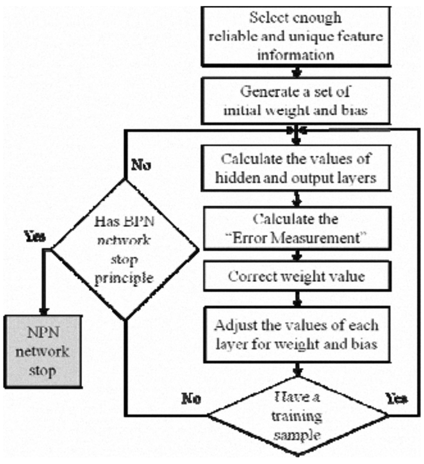


Figure 8. The procedure for improving precision.

Step 2: Initialize W_{ji} and W_{kj} by random values.

Step 3: Select a suitable activation function from Fig. 9 and input the trained data to the selected activation function. Then it calculates the output value y_j from the hidden layer and outputs value y_k from the output layer.

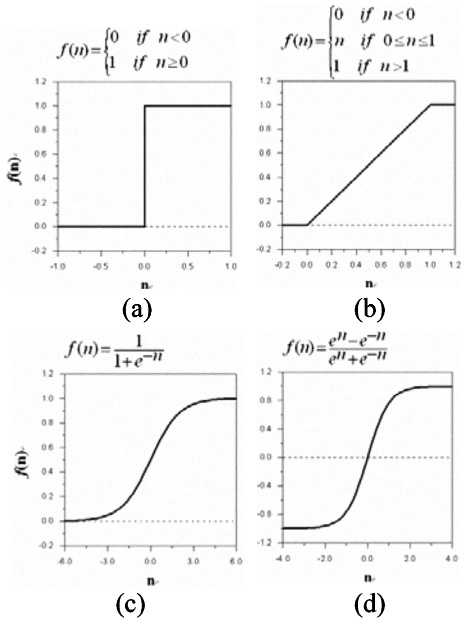


Figure 9. Four activation functions. (a) step function. (b) saturating linear function. (c) sigmoid function. (d) hyperbolic function.

Step4: Calculate the error function E. In order to find the optimum solution of E, we use the steepest descent method approach, as shown in (6).

$$E = \frac{1}{2}\sum_k (d_k - y_k)^2. \tag{6}$$

where d_k is the kth neurons objective output value, and y_k is the output value of the kth neuron at the output layer. In this step we try to reduce the difference between the input and output values.

Step 5: Calculate $\delta_k^n, k = 1,\ldots,K$, in the output layer as (7), and $\delta_j^n, j = 1,\ldots,L$, in the hidden layer as (8) respectively.

$$\delta_j^n = (d_j - y_j^n) f'(net_j^n). \tag{7}$$

$$\delta_j^n = \left[\sum_k \delta_k^{n+1} w_{kj}\right] f'(net_j^n). \tag{8}$$

Step6: Correct the weight $\left(W_{kj}(p+1) = W_{kj}(p) + \eta\delta_k^n(p) y_j^{n-1}(p)\right)$ in the output layer and the weight $\left(W_{ji}(p+1) = W_{ji}(p) + \eta\delta_j^n(p) y_i^{n-1}(p)\right)$ in the hidden layer, where p is the module of group p (the training module includes input and output values); η is the

learning rate, and generally the value is between 0 and 1.
Step7: Go back to Step 3 and then repeat the calculation and correction until the objective function reaches the stop standard or the largest training times.

By the above procedure, we can obtain a very accurate distance between the robot and the landmark. If the distance is too large to be in the accuracy range, the robot will search the other landmark.

3.5. The Absolute Coordinate of the Robot

The pan motor on the robot head can be used to estimate the direction of the robot. The angle "Φ" of the motor is rotated in clockwise, and the range is between 0° and 180°, as shown in Fig. 10. According to Fig. 10, the location of the robot can be derived by (9):

$$\begin{cases} x' = x + r\cos\phi \\ y' = y - r\sin\phi \end{cases} \text{,if the angle of the compass is } 0^{\circ} \text{ to } 359^{\circ} \quad (9)$$

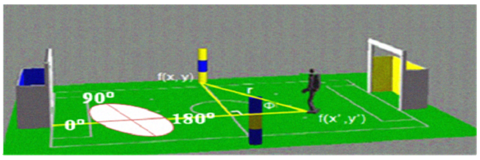


Figure 10. The direction of the robot in the soccer field.

4. Experimental Results

4.1. The Experimental Environment and the Robot Vision Module

The experiment is based on the feature of the competition field for 2009 RoboCup soccer humanoid league. The field contains two goals and two landmark poles, as shown in Fig. 1(b). Because of the width of the robot shoulder is 25cm, we set the unit length of the coordinate to be 30cm in length and the field can be divided into 29×17 blocks as shown in Fig. 11. The experimental robot vision module comprises a single CCD camera and pan/tilt motors as shown in Fig. 12. The CCD camera is the Logitech QuickCam® Pro [16] for Notebooks, and the pan/tilt motors are ROBOTIS Dynamixel RX-28 [17].

4.2. The Precision Simulation of Distance Measurement

For the BPN network approach, we need data for the three neurons in the input layer (the tilt angle, the landmark Ymin, and the size of the frame) and one in the hidden layer. The simulation result indicates that the most suitable neurons are ten as shown in Fig. 13. The learning rate is 0.1 and the output layer is one. By training these data, the precision can reach 2.44cm as shown in Fig. 14.

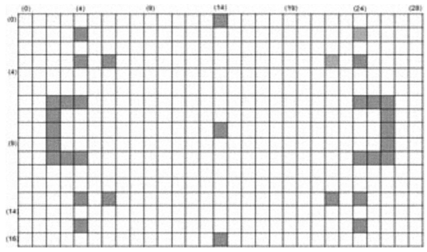


Figure 11. The RoboCup soccer field. The original field with 29×17 blocks [5].

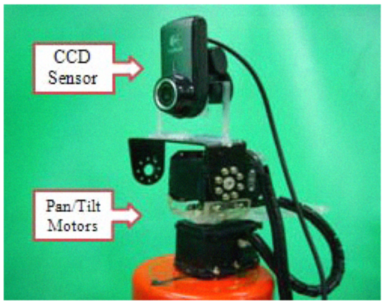


Figure 12. The robot vision module.

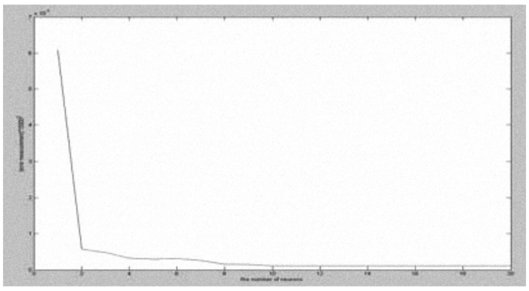


Figure 13. The number of the neuron error rate from 1 to 20.

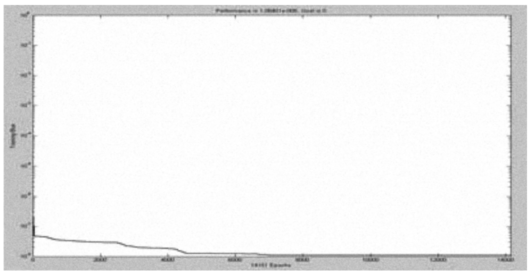


Figure 14. The error between the simulated and realdistance is about 2.44cm.

4.3. The Actual and Measured Distance

According to the experimental data, Fig. 15 shows the errors of the distance between the original and improved approach. The black line is for the actual distance, the red dotted line for the improved approach, and the blue dotted line for the original method. According to Fig. 15, the average error for the improved approach is 6.68cm and that of the original method is 87.23cm. Therefore, the proposed approach improves the accuracy significantly. Since the left and right sides of the field are with the same situation (Fig. 11), without loss of generality this experiment focuses on the right side of the field. Fig. 16 shows the measurement results of various locations of the robot, where the stars indicate the various locations of the robot. Table 1 shows the comparisons of the correct rates of the actual distance and the measured distance for the original method and the improved approach. The accuracy rate for the improved approach is 88.5%; on the other hand that of the original method is only 71.0%.

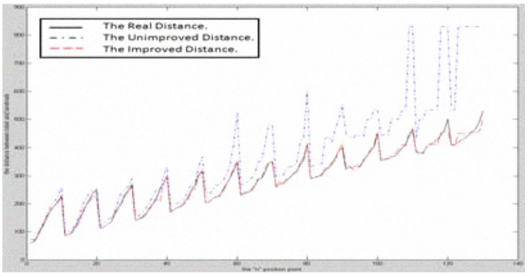


Figure 15. The error rates of the distance between the original and the improved approach.

Table 1. Comparisons of the correct rates for different methods.

Total experimental points = 130			
Situation	**Correct**	**Incorrect**	**Accuracy Rate**
Original Method	92	38	71.1%
Improved Approach	115	15	88.5%

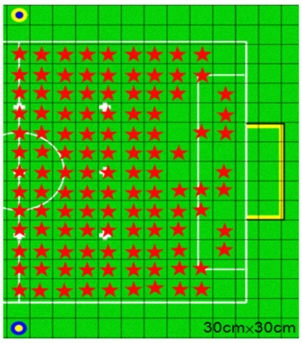


Figure 16. The various locations of the robot to measuring the distance between the robot and the landmarks.

5. Conclusions

This work proposes an efficient approach of self-localization for humanoid robot by the BPN technique. The proposed method can increase the precision of localization significantly. Due to the simple processing operation the processing speed can be as high as 15 fps. Upon the restrictions of the RoboCup soccer field, this work uses at most two landmarks for self-localization. Besides, we apply the adaptive two-dimensional head motion to have the localization to be elastically. Since the robot vision module can measure the distance between the robot and the landmark more accurately, the robot can localize itself on the absolute coordinate more precisely. The simulation results indicate that it is an efficient localization approach.

Acknowledgement

This work was supported by the National Science Council of Taiwan, R.O.C. under grant number NSC 98-2218-E-032-003.

References

[1] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. “Robocup: The robot world cup initiative,” IJCAI-95 Workshop on Entertainment and AI/ALife, pp. 19-24, 1995.

[2] FIRA RoboWorld Congress. http://www.fira2009.org.

[3] RoboCup Soccer Humanoid League Rules and Setup for the 2007 competition. http://waziwazi.com/robocup

[4] RoboCup Soccer Humanoid League Rules and Setup for the 2008 competition. http://www.robocup-cn.org/

[5] RoboCup Soccer Humanoid League Rules and Setup for the 2009 competition. http://www.robocup2009.org/

[6] I. Shimshoni, “On Mobile Robot Localization from Landmark Bearings,” IEEE Transactions on Robotics and Automation, vol. 18, no. 6, pp. 971-976, December 2002.

[7] M. Betke and L. Gurvits, “Mobile robot localization using landmarks,” IEEE Transactions on Robotics and Automation, vol. 13, no. 2, pp. 251-263, April 1997.

[8] Z.-G. Zhong, J.-Q. Yi, D.-B. Zhao, Y.-P. Hong, and X.-Z. Li, “Motion vision for mobile robot localization,” IEEE International Conference on Control, Automation, Robotics and Vision, vol. 1, pp. 261-266, December 2004.

[9] D. J. Kriegman, E. Triendl, and T. O. Binford, “Stereo vision and navigation in buildings for mobile robots,” IEEE Transactions on Robotics and Automation, vol. 5, no. 6, pp. 792-802, December 1989.

[10] S.-K. Choi, J. Yuh, and G. Y. Takashiqe, “Development of the omni-directional intelligent navigator,” IEEE Robotics & Automation Magazine, vol. 2, no. 1, pp. 44-53, March 1995.

[11] P.-R. Liu, M.-Q. Meng, and P.-X. Liu, “Moving object segmentation and detection for monocular robot based on active contour model,” Electronics Letters, vol. 41, no. 24, November 2005.

[12] C.-J. Zhang, S.-J. Ji, and X.-N. Fan, “Study on distance measurement based on monocular vision technique,” Journal of Shandong University of Science and Technology, vol. 26, no.4, pp. 65-68, October, 2007.

[13] Y. Xie and Y.-M. Yang, “A self-localization method with monocular vision for autonomous soccer robot,” Computer Science and Information Engineering, vol.22, no.10, pp. 129-132, January 2005.

[14] L.-H. Chiang, Neural Network — Application of MATLAB, Gau-Lih publish, Seventh Edition, July 2005.

[15] F.-J. Chang and L.-C. Chang, Artificial Neural Network, Tun-Ghua publish, Third Edition, August 2007.

[16] Logitech QuickCam® Pro for Notebooks. http://www.logitech.com/index.cfm/home/&cl=us,en.

[17] RX-28 MANUAL (ENGLISH) UPDATE v1.10. http://www.robotis.com/zbxe/5436.