使用 Takagi-Sugeno 模糊系統進行行動機器人之全方位視覺的影像轉換

Omnidirecitonal Image Mapping Using a Takagi-Sugeno Fuzzy System for a Mobile Robot

游智鈞 魏瑋廷 吳宣誼 王銀添*
Zhi-Jun You, Wei-Ting Wei, Xuan-Yi Wu, Yin-Tien Wang*

摘要

本文發展一套具備全方位視覺系統的行動機器人，並應用在機器人足球比賽。全方位視覺系統包含一個 CCD，向上面對一個曲面鏡，並擷取曲面鏡反射的影像。此曲面鏡頃斜 23°裝設，以便取得機器人面前更大範圍的視角。本文提出一個基於模糊系統理論的影像轉換方法，由 2D 影像取得 3D 場地座標訊息。同時，我們建立一個有效的程序以設定物件色彩比對範圍。整合的系統在機器人足球比賽場地測試，以評估所發展系統的實用性。
關鍵詞：全方位視覺，自主行動機器人，影像伺服控制，特徵辨識

Abstract

A mobile robot with an on-board omnidirectional vision system is developed in this paper, and utilized to play soccer games. The omnidirectional vision system includes a CCD facing upwards beneath the convex mirror and taking the picture reflected in the mirror. The mirror is tilted 23° to get a larger field of vision in front of the robot. We propose a new mapping method based on the fuzzy system theory to obtain a 3D ground coordinate from a 2D omnidirectional image. An effective procedure is constructed to set the color range of the object in the image. The integrated system is tested on a robot soccer field to evaluate the practical usage of the developed system.
Keywords: omnidirectional vision, autonomous mobile robot, visual servo control, pattern recognition

I. INTRODUCTION

A mobile robot with an on-board vision has the capability to capture an image and handle image processing locally. However, due to the limitation of the view angle, an on-board vision system equipped in front of the robot cannot take the picture of the whole surrounding. An omnidirectional vision system is developed to overcome the limitation. Omnidirectional vision makes it possible to cover a 360° filed of vision, by analyzing only one image. The ideal of omnidirectional vision was firstly proposed by Rees [1]. Recently, Nayar [2] has geometrically analyzed the complete class of single-lens single-mirror omnidirectional vision systems and developed an ideal vision system using a parabola mirror. In this paper, an omnidirectional vision with the mirror tilted 23° is implemented to get a larger field of view in front of the mobile robot and provide the robot with a fast tracking capability.

An omnidirectional vision system is composed of two parts: a spherical mirror is utilized to reflect the image around the robot, and a CCD is placed facing upwards beneath the mirror to take the picture reflected in the mirror, as shown in Figures 1 and 2. However, the image taken by the omnidirectional camera is distorted. It is required to develop a technique to correct the distortion and obtain an unwarped image. Gaspar and Victor [3] solved the image distortion problem using a look-up table. However, this method required a large volume of computer memory for mapping an asymmetric image. The proposed image mapping method here is based on the concepts of Takagi-Sugeno (TS) fuzzy system [4-6] and the procedures of clustering methods to resolve the problem of memory consumed.

Motion control system of the wheeled mobile robot can be divided into three levels, namely, self-localization, action selection and path planning, and contour control. The research in this paper investigates the problems in-

淡江大學機械與機電工程學系
*Corresponding author. E-mail: ytwang@mail.tku.edu.tw
Department of Mechanical and Electro-Mechanical Engineering, Tamkang University, Tamsui, Taipei Hsien, TAIWAN 25137
Manuscript received 2 July 2007; revised 26 February 2008; accepted 7 April 2008

cluding self-localization and contour control. The proposed algorithm utilizes an omnidirectional vision system to locate the position of the mobile robot and provide the robot for object tracking control.

In Section II, the problem of image distorted and image transformation is discussed. The proposed fuzzy image mapping is presented in Section III. The color model for robot vision is depicted in Section IV. An experimental example of mobile robot object tracking is presented in Section V. Some conclusion remarks are discussed in the last section.

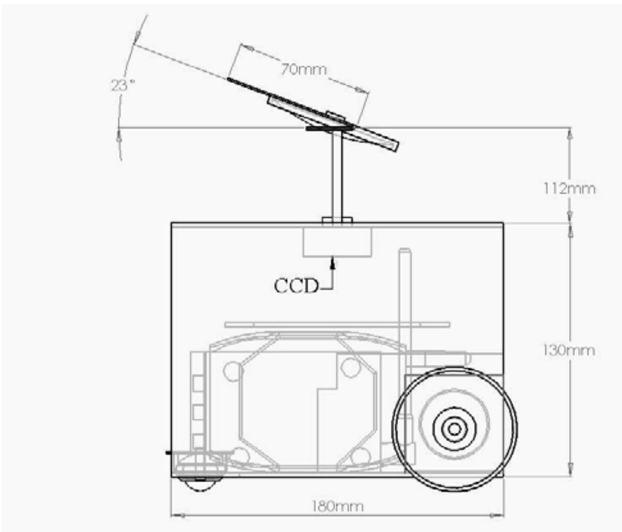


Fig. 1 Design of mobile robot with on-board vision

II. TRANSFORMATION OF OMNIDIRECTIONAL IMAGE

In the omnidirectional vision system, an object in 3D-space is reflected by a convex mirror and projected on a 2D image. The image taken by a CCD is digitized and fed back to the control system. In the robot servoing control process, the controller has to determine the object's coordinate in 3D-space from the 2D image. The inverse coordinate transformation method from the 2D image to the 3D coordinate can be derived by using the image formation theory of a convex mirror. Firstly, it determines the procedure of projecting a space point $P(x,y,z)$ in Cartesian coordinate onto a image point $P_i(\psi_i,r_i,z_i)$ in a cylindrical coordinate, and then derives the inverse coordinate transformation method. The notations ψ, r, and z represent the angle, radius, and height of the cylindrical coordinates, respectively. The space point $P(x,y,z)$ can be expressed in cylindrical coordinate as

$$P = [\psi \quad r \quad z]^T = [\arctan(\frac{y}{x}) \quad \sqrt{x^2+y^2} \quad z]^T$$

Assume that a spherical mirror is utilized in the omnidirectional vision system, as shown in Figure 3, so the equation of image formation can be determined as the following expression.

$$z_m = -\tan(\frac{\pi}{2} - \beta) \cdot r_m + L \qquad (1)$$

$$z_m^2 + r_m^2 = R^2 \qquad (2)$$

$$2\arctan(\frac{r_m}{z_m}) = \arctan(\frac{r - r_m}{z - z_m}) - \beta \qquad (3)$$

where r_m and z_m are the coordinates on the spherical mirror; R is the radius of the spherical mirror; L is the length from

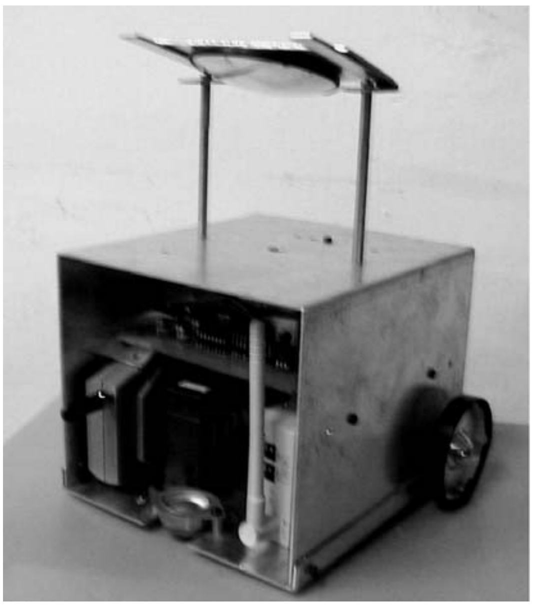


Fig. 2 Implementation of the robot with an omnidirectional camera

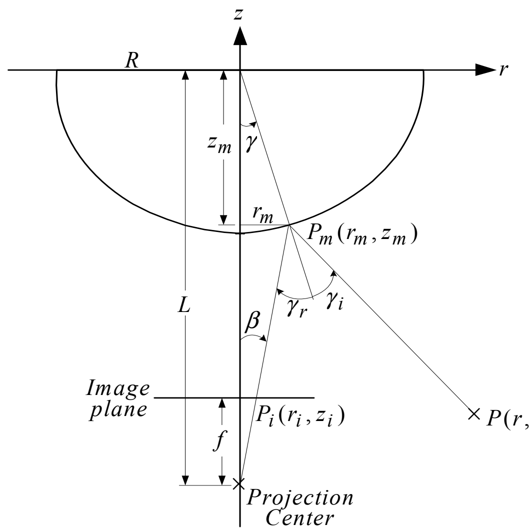


Fig. 3 Perspective projection model of a spherical mirror

the center of the spherical mirror to the projection center; β is the angle of projected point. Solve the equations (1)-(3), and then the expression of r_m, z_m, and β can be determined. The coordinates of a point on the image can be obtained by the perspective projection model,

$$r_i = \alpha_r f \tan\beta \cos\psi + r_{io} \tag{4}$$

$$z_i = \alpha_z f \tan\beta \sin\psi + z_{io} \tag{5}$$

$$\psi_i = \psi \tag{6}$$

where f is distance from the image plane to the projection center. α_r and α_z are the image scale factors; (r_{io},z_{io}) is the principal point of the image coordinate system. From the equations (1)-(6), the coordinate of the planar point $P_i(\psi_i,r_i,z_i)$ can be determined. Since these equations are nonlinear function, it is very difficult to solve these equations analytically and implement in real-time.

Gaspar and Victor [3] utilized a look-up table to handle the image distorted problem. They proposed the ideal that the distortion ratio ζ of an omnidirectional image is a function of the radius r_i shown in Figure 3, therefore a look-up table describing the function, $\zeta(r_i)$, can be used to determine the coordinate inversely. However, this method required a large volume of computer memory for mapping a distorted image.

In the paper, the spherical mirror is tilted 23° to get a larger field of view in the front of the robot. In this case, the distortion ratio of the image is asymmetric generally. Therefore, the perspective projection model is more complex than that of a flat mirror and the coordinate transformation can not be simply determined by a look-up table.

III. FUZZY IMAGE MAPPING

In order to improve the disadvantage of complexity in the image recovery procedure using the perspective projection model of a convex mirror, a TS fuzzy system model for image mapping is proposed based on the clustering method [4]. Assume that there is a group of input-output data pairs, $x=(x(1), x(2), \ldots, x(p\text{-}1), x(p))$, in which $x(1), x(2), \ldots, x(p\text{-}1)$ represent $(p\text{-}1)$ input variables of the system, and $x(p)$ is the output variable. For a TS fuzzy system with c fuzzy rules, the rule base of the system can be described as the following expression [4]:

Rule m ： if x is $A_m(x)$ then y_m is

$$y_m = b_{m0} + \sum_{j=1}^{p-1} b_{mj}x(j) \quad m = 1,2,\ldots,c \tag{7}$$

where $A_m(x)$ is the antecedent proposition of the fuzzy rule; b_{mj} are coefficients; y_m is the consequence of the fuzzy rule. The proposition $A_m(x)$ can be viewed as the fuzzification from the original input data; while the consequence of the fuzzy rule, y_m, is set to be a linear combination of the input variables. If the fuzzy system has c fuzzy rules, the output of the TS fuzzy system can be obtained by the weighted average method,

$$y = \frac{\sum_{m=1}^{c} A_m(x)\cdot y_m}{\sum_{m=1}^{c} A_m(x)} = \sum_{m=1}^{c} g_m \cdot \left[b_{m0} + \sum_{j=1}^{p-1} b_{mj}x(j) \right] \tag{8}$$

$$g_m = \frac{A_m(x)}{\sum_{j=1}^{c} A_j(x)}, \quad m = 1,2,\ldots,c$$

As mentioned in the preceding section, the number of rules, c, in the rule base of the TS fuzzy model affects the performance of the system. If the fuzzy system contains too many rules than it needs, the system becomes a complex system. On the contrary, if the system has too few rules, it would not have enough information to model the real system. By the clustering method, the number of rules is decided by adjusting the relational grade of the clusters. It will be defined latterly, the relational grade as a Gaussian function, which is based on the distance between two vectors. In the procedure of clustering methods, the collected data is divided into several subsets or strings of data according to the relational grades between these data. Therefore, each string of data has its own characteristics that can be distinguished from other data strings. The procedures to determine the structure of the fuzzy system are described as follows.

Assume a set of n vectors in a p-dimensional space, $X = \{x_1, x_2, \cdots, x_n\}$, where $x_i=(x_i(1), x_i(2),\ldots, x_i(p))$ is a vector with p variables, in which $(x_i(1), x_i(2),\ldots, x_i(p\text{-}1))$ and $x_i(p)$ are input and output variables of i^{th} data point, respectively. Among these n vectors, the vectors which have a high relational grade can be collected to be a string named cluster. In this paper, the concept of similarity proposed by Wong and Chen [7] is utilized to determine the relational grade. According to the method, first select a data point as a reference vector, and then find a comparative vector that has high relational grade with the reference vector. Further, choose the comparative vector with high relational grade as a new reference vector, and repeat the procedure. By this recursion method, the reference vector can be replaced during each cycle of the procedure, and eventually converges to the center of a cluster. The procedure is summarized in five recursive steps [7]:

Step 1: define n movable vectors v_i $(i = 1,2,\ldots,n)$ and let $v_i=x_i$, where x_i is the initial value of v_i;

Step 2: calculate the similarity by the following equation,

$$r_{ij} = \exp\left(-\frac{\|v_i - v_j\|^2}{2\sigma^2} \right), \quad i = 1,2,\ldots,n \;;\; j = 1,2,\ldots,n \tag{9}$$

where r_{ij} represents the relational grades between the reference vector v_i and the comparative vector v_j; $\|v_i - v_j\|$ is the Euclidean distance between v_i and v_j; and σ is the width of the Gaussian function in Equation (9);

Step 3: modify the relational grades between the reference vector v_i and the comparative vector v_j according to the following rule,

$$r_{ij} = \begin{cases} 0 & if\ r_{ij} < \xi \\ r_{ij} & \text{otherwise} \end{cases}$$

where ξ is a small constant set up to be 0.01 in this paper;

Step 4: calculate a new vector set

$v_i' = (v_i'(1), v_i'(2), ..., v_i'(p))$, $i = 1,2,....,n$,

$$v_i'(k) = \frac{\sum_{j=1}^{n} r_{ij} v_j(k)}{\sum_{j=1}^{n} r_{ij}}, \ k = 1,2,....,p$$

Step 5: if all the vectors v_i' are the same as v_i, i=1, 2, …, n, then stop; otherwise let $v_i = v_i'$, i=1, 2, …, n, and go to Step 2.

By this procedure, the data points with high relational grades are collected as a cluster. The relational grade is modified in Step 3 to prevent the movable vector from being affected by the vectors with low relational grades. The movable vectors will gradually converge to a vector of values. Therefore, the number of convergent vectors is the number of clusters, and the convergent vector is viewed as the center of the corresponding cluster. By the clustering method, n input-output data are divided into c clusters,

$$c_m = \{c_m(1), c_m(2), \ldots, c_m(p)\}, \quad m = 1,2,...,c$$

Once the cluster centers are determined, the antecedent proposition, $A_m(x_i)$, of the i^{th} input, x_i, can be arranged according to the relationship between i^{th} input data and m^{th} cluster center. A Gauss function [4] is chosen to represent the membership function,

$$A_m(x_i) = \exp\left(- \frac{\sum_{k=1}^{p-1} (x_i(k) - c_m(k))^2}{2\delta_m^2} \right) \tag{10}$$

$$\delta_m = \sqrt{\frac{-\sum_{k=1}^{p-1} \left(x_m^*(k) - c_m(k)\right)^2}{2\ln(\alpha)}}, \ m = 1,2,...,c$$

where $m = 1,2,...,c$; $i = 1,2,...,n$; c_m is the m^{th} clustering center; δ_m indicates the width of the Gaussian function in Equation (10); x_m^* is the most far away data point of the m^{th} clustering data; α is a constant between 0 and 1.

Using the procedure of the clustering method, n data points are distributed into c clusters. According to Equation (8), the output of the TS fuzzy system with n input-output data points and c cluster centers can be expressed as the following equations,

$$y_i = \sum_{m=1}^{c} \left[g_{im} b_{m0} + g_{im} \sum_{j=1}^{p-1} b_{mj} x_i(j) \right] \tag{11}$$

$$g_{im} = \frac{A_m(x_i)}{\sum_{j=1}^{c} A_j(x_i)}, \quad m = 1,2,\cdots,c; \quad i = 1,2,\cdots,n$$

Define the vectors and matrices containing input and output variables as the following equations,

$$Y = [y_1 \quad y_2 \quad \cdots \quad y_n]^T$$

$$h = [b_{10} \quad b_{11} \quad \cdots \quad b_{1p-1} \quad \cdots \quad b_{c0} \quad b_{c1} \quad \cdots \quad b_{cp-1}]^T$$

$$W = \begin{bmatrix} g_{11} & g_{11}x_1(1) & \cdots & g_{11}x_1(p-1) & \cdots & g_{1c} & g_{1c}x_1(1) & \cdots & g_{1c}x_1(p-1) \\ g_{21} & g_{21}x_2(1) & \cdots & g_{21}x_2(p-1) & \cdots & g_{2c} & g_{2c}x_2(1) & \cdots & g_{2c}x_2(p-1) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n1}x_n(1) & \cdots & g_{n1}x_n(p-1) & \cdots & g_{nc} & g_{nc}x_n(1) & \cdots & g_{nc}x_n(p-1) \end{bmatrix}$$

The coefficient vector, h, can be determined using the pseudo-inverse matrix method [8] or the Root-Mean-Squared-Error (RLSE) method [9]. Once the coefficients are determined, the fuzzy system in Equation (11) can be used to model the perspective projection system.

As an example to demonstrate the validity of the TS-fuzzy system for image recovery, we arrange a test field to model the surrounding of a robot as shown in Figure 4, and a TS fuzzy system is developed to replace the perspective projection model. In the figure, the black circles are utilized to mark the whole area of the soccer field. The distances between these marks are $3cm$, $5cm$, and $10cm$ for the black circles located in the right-hand side, the middle, and the left-hand side of the field, respectively.

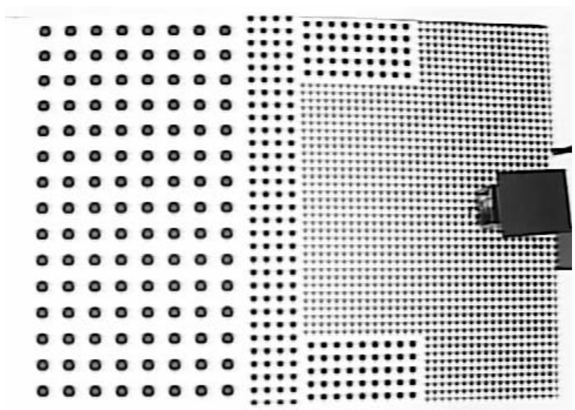


Fig. 4 Black circle marks in a soccer field (top view)

One image was taken by the omnidirectional vision system and shown in Figure 5. The developed TS fuzzy system is employed to represent the mapping of one pixel on the warped image to the unwarped image. We calculate the Root-Mean-Squared error for the image positions recovered by the proposed TS fuzzy system,

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(P-\widetilde{P}\right)^2}{n}}$$

where P is the real position of a block circle, and $\widetilde{P}$ is the estimated position of the same circle. n is the total number of circles been considered. The results are listed in Table 1, where the values of RMSE are in *cm*. Several different widths of the Gaussian function, σ, are chosen for the fuzzy system, and result in different number of clusters for the fuzzy system. When the value of σ decreases, the RMSE of the system output will decrease. However, the complexity of the system will increase as we can see from the increase number of rules in third column. Another example as shown in Figure 6, we see that a square block on the soccer field is twisted on the omnidirectional image, and the recovery mapping result by the TS fuzzy system is shown in Figure 7. The RMS errors are listed in Table 2.

IV. COLOR MODEL

The purpose of a color model is to allow the convenient specification of colors within some color gamut. The red, green, blue (RGB) color model is a hardware-oriented and used with color CRT monitors. The RGB model employs a Cartesian coordinate system and the subset of interest is the cube shown in Figure 8. Unfortunately, the RGB color model is not easy to use, because it does not relate directly to intuitive color notions of hue, saturation,

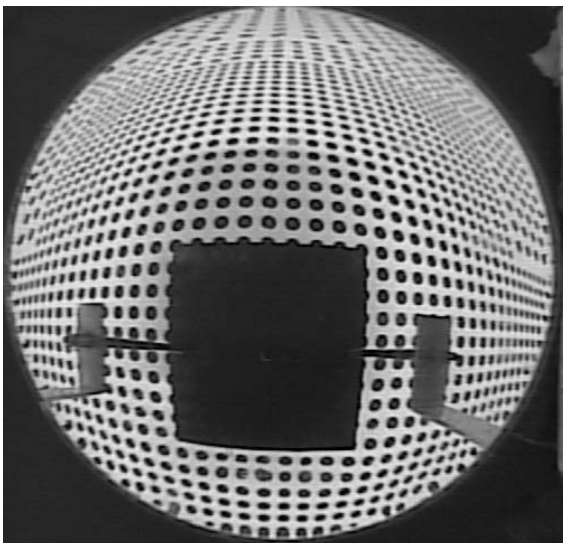


Fig. 5 Image taken by an omnidirectional camera

Table 1 Results of the fuzzy system model

	σ	Rules	RMSE	Matrix W
x	0.5	19	1.0983	$9p\times9n$
	0.4	29	0.6976	$11p\times11n$
	0.361	60	0.5198	$19p\times19n$
y	0.5	13	1.2334	$7p\times7n$
	0.4	46	0.8134	$13p\times13n$
	0.33	55	0.6778	$16p\times16n$

Table 2 RMSE of the fuzzy system

	x (σ =0.12)	y (σ =0.17)
RMSE	0.5520	0.5768

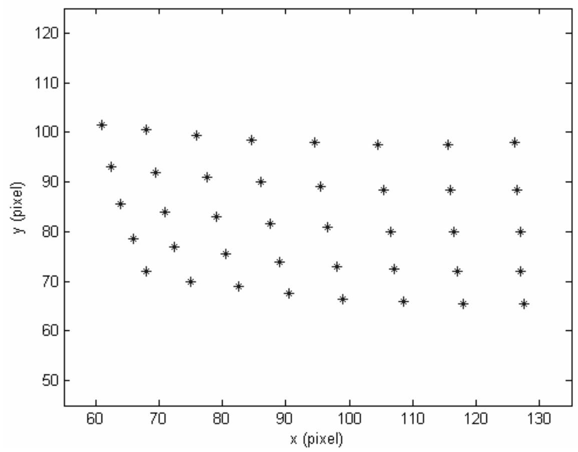


Fig. 6 A square area on the omnidirectional image

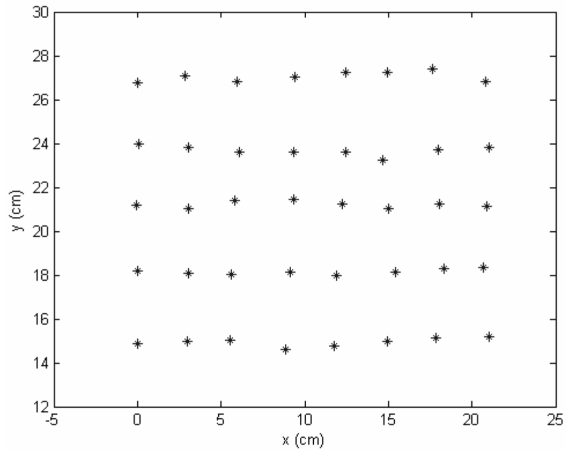


Fig. 7 Unwarped image of the square area

and brightness. Therefore, another class of models has been developed with ease of use as a goal [10]. The HSV (hue, saturation, value) model is utilized in the paper. By contrast to the RGB model, the HSV model is user-oriented, being based on the intuitive appeal of the artist's tint, shade, and tone [10]. The HSV coordinate system is cylindrical, and the subset of the space within which the model is defined is a hexcone, or six-sided pyramid. Hue is measured by the angle around the vertical axis. The value of S is a ratio ranging from 0 on the center line to 1 on the triangle sides of the hexcone. Saturation is measured relative to the color gamut represented by the model. The RGB-to-HSV mapping is defined as follows [10]:

$$H = \cos^{-1}\left(\frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)(G-B)}} \right) \tag{11}$$

$$S = \frac{\max(R,G,B)-\min(R,G,B)}{\max(R,G,B)} \tag{12}$$

$$V = \frac{\max(R,G,B)}{255} \tag{13}$$

The algorithm of Figure 9 is an approximate calculation for Equation (11)-(13). Alternative method can be used to convert RGB to HSV by creating a look-up table. However, the total computer memory required for the table is about 48 megabytes (=256×256×256×3 bytes). For robot object and path tracking, it is needed to set a specific color range for each of the tracked object.

V. EXPERIMENTAL RESULTS

The fuzzy image mapping method developed in the preceding sections is integrated with a mobile robot and tested in a soccer filed to track a ball. The result is shown in Figure 10. The figure records the continuous motion of the robot approaching to the object.

As another example of robot visual servo control, the images shown in Figure 11 are the top-views of robot continuous motion in the soccer field. The white arrow is placed on the top of the robot and indicating the motion direction of the robot. The red (dark) mark initially located near to the robot represents an obstacle, while the yellow (light) mark represents the goal position of the robot. We can see that the robot avoids the obstacle by turning right to follow a detour and approach the goal.

```
Void RGB_To_HSV(int r, int g, int b, int &h, int &s, int &v)
//Given: r, g, b, each in [0,255].
//Desired: h, s, v in [0,255] except if s=0, then h=UNDEFINED, which
// is some constant with a value outside the interval [0,255].
{
    // r*=255; g*=255; b*=255;    If r, g, b are given in [0,1]
    int max = maximum(r,g,b);
    int min = minimum(r,g,b);
    v = max;  //This is the vale v.
    //Next calculate saturation, s.
    if (max <> 0)        s = (max – min)*255/max;     //s is the saturation.
    else s = 0;              //Saturation is 0 if red, green and blue are all 0.
    if (s == 0)    { h = UNDEFINED;}
    else   //Chromatic case: Saturation is not 0, so determine hue.
    {   delta = max – min;
        if (r == max)
        {  h=(g–b)/delta;  //Resulting color is between yellow and ma-
genta. }
        else
        { if (g == max)
            {h=2+(b–r)/delta;//Resulting color is between cyan and yel-
low. }
            else
            { if (b == max)
                {h=4+(r–g)/delta; //Resulting color is between magenta and
cyan.}
            }
        }
        h = h * 60;                            //Convert hue to degrees.
        if (h < 0) {h = h +360;}               //Make sure hue is nonnega-
tive.
        h*=255/360;                    //Convert hue to be in [0,255].
    }       // End of chromatic case.
}       //End of RGB_To_HSV.
```

Fig. 9 RGB-to-HSV algorithm

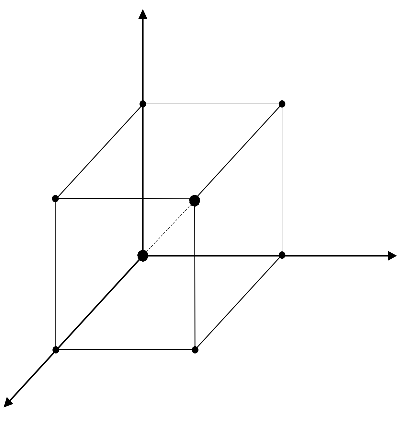


Fig. 8 The RGB color model

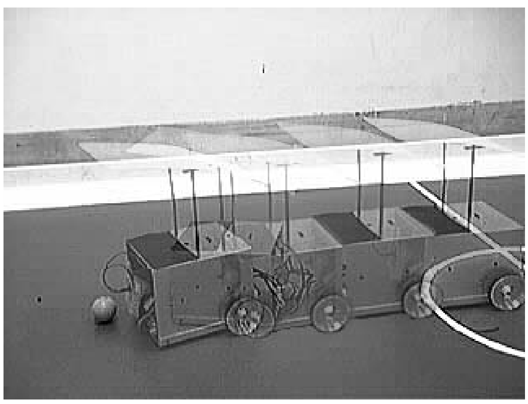
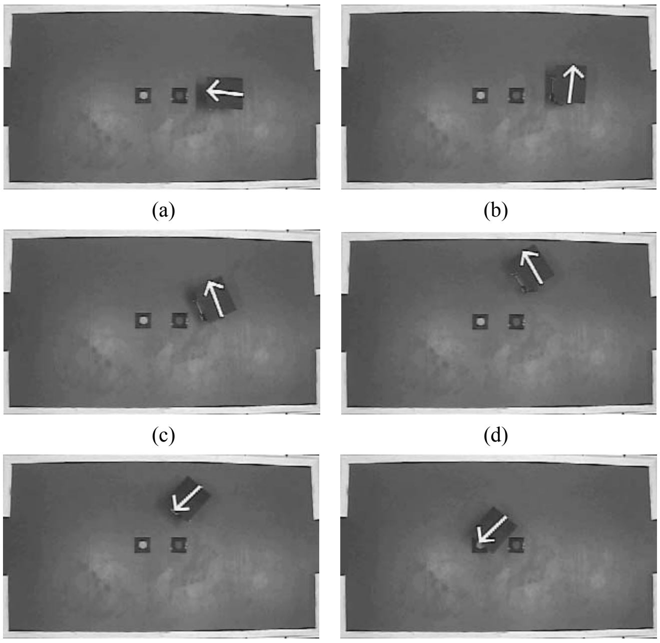


Fig. 10 Robot tracking a ball

VI. CONCLUSIONS

A mobile robot with an on-board omnidirectional vision system is developed in this paper. The complex problem of mapping a pixel on a 2D image to a 3D ground coordinate is solved in this paper by using the concepts of the TS-fuzzy system. The integrated system was tested on a robot soccer field, and two examples of robot visual servo control are presented that the validity of the proposed TS-fuzzy system for image recovery was demonstrated. The results indicate that the proposed method is effective and accurate.



(a) (b) (c) (d) (e) (f)

Fig. 11 Robot visual servo control

ACKNOWLEDGEMENTS

This work was supported by the National Science Council in Taiwan under grant no. NSC91-2213-E-032-005.

REFERENCES

[1] D. W. Rees, "Panoramic television viewing system," *United States Patent*, No. 3,505,465, 1970.

[2] S. K. Nayar, "Catadioptric omnidirectional camera," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 482-488, 1997.

[3] J. Gaspar and J. S. Victor, "Visual path following with a catadioptric omnidirectional camera," *International Symposium on Intelligent Robotic Systems*, pp. 139-147, 1999.

[4] L. X. Wang, *A Course in Fuzzy System and Control*, Prentice-Hall, 1997.

[5] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol. 28, no. 1, pp. 15-33, 1988.

[6] G. T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernation*, vol. 15, no. 1, pp. 116-132, 1985.

[7] C. C. Wong and C. C. Chen, "A clustering-based method for fuzzy modeling," *IEICE Transaction on Information and System*, vol. E82-D, pp. 1058-1065, 1999.

[8] G. Strang, *Linear Algebra and Its Applications*, 2nd ed., Academic Press, 1980.

[9] C. L. Phillips and H. T. Nagle, *Digital Control System*, 3rd ed., Prentice Hall, 1995.

[10] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice in C*, Addison-Wesley, 1996.