



## OBJECT TRACKING SYSTEM BY TWO PAN-TILT MOVING CAMERAS AND ARM ROBOT \*

Hiroyuki Ukida <sup>1)</sup>, Yoshio Tanimoto <sup>2)</sup>, Hideki Yamamoto <sup>3)</sup>

<sup>1)</sup> Faculty of Engineering, The University of Tokushima, 2-1 Minamijosanjima-cho, Tokushima-shi, Tokushima, JAPAN, ukida@me.tokushima-u.ac.jp

<sup>2)</sup> Kibikogen Rehabilitation Center, 7511 Yoshikawa Kibichuo-cho, Kaga-gun, Okayama, JAPAN, tanimoto@kibirihah.rofuku.go.jp

<sup>3)</sup> Faculty of Education, Okayama University, 3-1-1 Tsushimanaka, Okayama-shi, Okayama, JAPAN, yamamoto@cc.okayama-u.ac.jp

**Abstract:** *We propose an object tracking system using an arm robot and two pan-tilt cameras. In this study, we use the active search method to detect the object location in images of cameras. Moreover, we estimate the 3D coordinates of this object from detected locations in images using the binocular stereo method. By using 3D object locations, the robot and cameras are rotated toward to the object. To realize the high speed performance, we use the parallel processing by the thread function for the processes of the object detection, the 3D coordinate estimation, the camera control and the robot control. We perform the object tracking experiments and confirm the efficiency of the proposed method.*

**Keywords:** *object tracking, pan-tilt camera, arm robot, active search, stereo method, parallel processing.*

### 1. INTRODUCTION

In the research of autonomous robot applications, the development of surveillance systems, the kinematics analysis of human body and so on, to track moving objects and to measure their trajectories are important problems. In this study, we use video cameras as the sensors to detect the object. Since the range of vision field is restricted with the usual camera, it is difficult to track widely moving objects. Hence, the method of mounting the camera on a robot and moving it freely is proposed in [1]. However, generally, the robot is large-sized and its motion is not much high-speed, hence, it is difficult to track a quickly moving object.

Recently, “the pan-tilt camera” is used in the teleconference systems, etc. This camera can rotate its lens and imaging device to horizontal (pan) and vertical (tilt) direction quickly and we can take images of more wide area than the fixed (non-moving) camera. Hence the object tracking methods using this camera are also proposed [2],[3]. In this case, although a range of camera's view spreads rather than a fixed camera, the ranges of pan and tilt rotations are still limited. Hence it is difficult to take omni-directional images. To overcome such problems, we propose a new tracking system using an arm robot equipped with two pan-tilt cameras, and discuss the high speed and wide range object

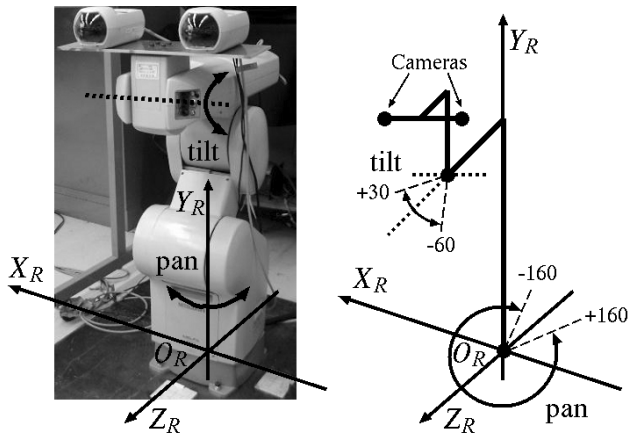
tracking method.

In the object tracking process, first, we must find the object in the images taking by cameras. But, since the object is moving, the appearance of the object, that is the size and the location of the object in the image, varies every moment. In order to search the object quickly in the image, we apply “the active search method”[4],[5],[6] for the object search process. This method can detect the object faster than the standard template matching method.

Moreover, the object tracking process consists of many other processes not only the object detection but the camera motion control and the robot motion control processes. Among these processes, the object detection process and the camera motion control process can not perform separately. Because, if the images are acquired when the cameras are moving, the acquired images will be blurred. On the other hand, since the robot motion speed is slow, the robot control process should be perform separately from other processes. Here, we propose an parallel algorithm of this system and show the effectiveness in the experiments.

The rest of this paper is organized as follows: In section 2, we show the configuration of cameras and robot. Section 3 explain the tracking approach. In section 4, we show experimental results of the object tracking. And we show the summary and future

works in section 5.



(a) Overview. (b) Robot coordinate system.  
Fig.1 - Arm robot and pan-tilt cameras.

## 2. CONFIGURATION OF CAMERAS AND ROBOT

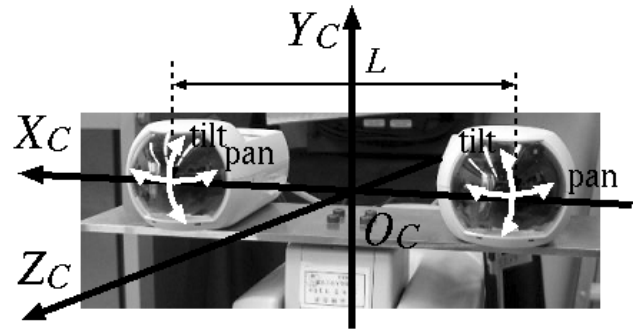
Fig.1 (a) shows the arm robot and pan-tilt cameras we use. Two pan-tilt cameras are equipped on the upper part of the arm robot. This robot has six joints (six degrees of freedom). But we use only the first and fifth joints. The range of horizontal (pan) rotation by the first joint is from -160 to +160[deg.], and the vertical (tilt) rotation by the fifth joint is from -60 to +30[deg.] (Fig.1(b)). The rotational speed of each joint (the angular velocity) is about 25[deg./sec.].  $O_R - X_R - Y_R - Z_R$  in Fig.1 (b) is called the robot coordinate system attached to the first joint.

Fig.2 (a) shows the pan-tilt cameras we use. The rotation angles of the imaging device in these cameras are from -30 to +30[deg.](the horizontal (pan) direction) and from -15 to +15[deg.](the vertical (tilt) direction) (Fig.2 (b)). The angular velocity of each rotation is about 45[deg./sec.]. The distance  $L$  between these cameras is 200[mm].  $O_C - X_C - Y_C - Z_C$  in Fig.2 (b) is called the camera coordinate system attached to the center of them.

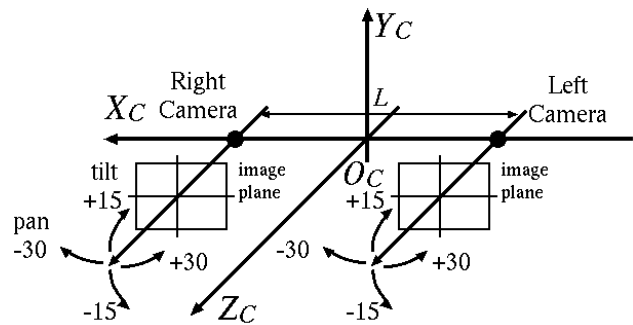
## 3. OBJECT TRACKING APPROACH

In order to track moving objects, we must process many tasks such as the object detection, the 3D coordinates estimation, the control of cameras and robot and so on. But if these tasks are processed sequentially, the total tracking process will cost much time and it will be difficult to track moving objects. Especially, the processing time to move robot need more long time than the software process such as the image processing.

In this paper, we divide the tracking process into two software modules and execute them in parallel (Fig.3). And we also prepare the shared memory to transfer some data between these modules. The



(a) Overview of cameras.



(b) Camera coordinate system.  
Fig.2 - Pan-tilt cameras.

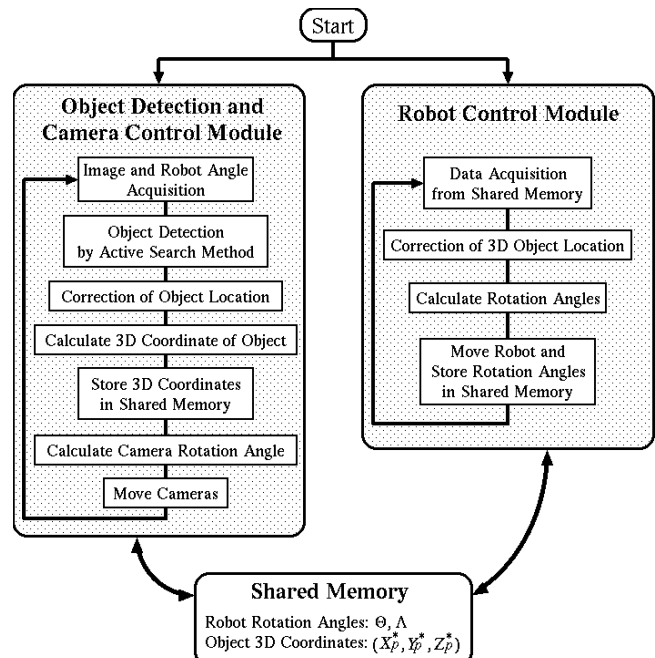


Fig.3 - Parallel processing modules.

following shows the overview of each module:

1. Object Detection and Camera Control Module  
Taking images by the right and left cameras, and detecting the object in these images. Then, estimating the 3D coordinates of this object by

the stereo method[7]. And, moving cameras in which the center of the camera image plane locates at the object position in the input image.

## 2. Robot Control Module

Correcting object 3D location and moving the arm robot facing to the object.

The parallel processing of these modules is realized by using the thread mechanism[8] provided by the computer operating system function. The multitasking computer can usually execute many application programs in parallel by the time sharing system (TSS). The thread mechanism can execute many function modules in an application program in parallel. By using this mechanism, when the waiting time for movement of the robot is long, it becomes possible to perform another processing. In the following sections, we describe the details of the object detection modules.

### 3.1 Object Detection and Camera Control Module

This module consists of three processes as follows:

1. For each (right and left) camera, detecting the object location in the captured image.
2. From the object locations in the right and left camera images, estimating the 3D coordinates of the object by the stereo method.
3. Move cameras in which the center of camera locates at the object position.

In general, as the method which detects an object pattern in an input image, the template matching method is famous and is often used. In this method, the sum of absolute differences or the normalized cross-correlations of the intensities of the input image and the object pattern image are used as the similarity measure[9]. And this method scans all over the input image and detect the object patten location where it obtains best similarity. However, the object pattern in the input image is variously changed about the location, pose (direction), and size. Since this method must perform the matching process changing parameters of the location, pose and size in round robin, it takes much computational time.

To avoid such problems, we use “the active search method” to detect the object in the input camera images. This method compares the object image (we call it “the reference image”) with the input image evaluating the similarity calculated from the histogram of the color data in these images. So, this

similarity is invariable about the object pose. And, from the histogram calculation, when the similarity of a region can be calculated:

1. In the neighbor regions of this region, the maximum value considered as the similarity can be also estimated without calculating the similarity directly, and,
2. We can estimate a range around the region in which the matching process does not need to perform.

From these characteristics, the active search method does not perform the matching process all over the input image like the template matching. And, because of the histogram calculation, this method guarantees the accuracy as same as the template matching. In other words, this method performs:

- The dense matching processes in the high similarity places, and,
- The rough matching processes in the low similarity places,

and reduces the number of matching processes. Therefore, this method can detect the object more efficiently and fewer computational time than the template matching. In the preliminary experiments, we can confirm that the active search method can be performed at high speed about 40 times rather than the template matching method.

Moreover, the performance of the active search method depends on the object scanning method in the images. From the preliminary experiments, we confirm that the scanning method which starts from the previously object detected position to the outside of the image is more effective than the normal scanning method (from top-left to bottom-right).

The following shows the processes of the object detection and camera control module using the active search method.

#### Step 1-1. Image Acquisition

Acquire the input images by right and left cameras simultaneously. In this step, the rotational angles of the robot ( $\Theta$ ,  $\Lambda$ ) are also acquired from the shared memory.

#### Step 1-2. Object Detection by Active Search Method

For each input image, detect the object location in the image by using the active search method. Let the center of gravity locations of the object areas in these images be  $(x_r, y_r)$ ,  $(x_l, y_l)$ . If the object location can not be detected in one or both images, return to Step 1-1.

#### Step 1-3. Correction of Object Location

The object locations in input images  $(x_r, y_r)$  and

$(x_l, y_l)$  are not always corresponding coordinates because of the following reasons:

a. There are some differences between the appearances of the object in the input and reference images of the active search method.

b. There are also some differences of the appearances between the right and left images.

But, when the  $(x_r, y_r)$  and  $(x_l, y_l)$  are not corresponded, we can not calculate the 3D coordinates accurately in next Step 1-4. Hence, we correct the object location by the template matching using sub images around  $(x_r, y_r)$  and  $(x_l, y_l)$ . Let the corrected object locations be  $(x_r^*, y_r^*)$  and  $(x_l^*, y_l^*)$ .

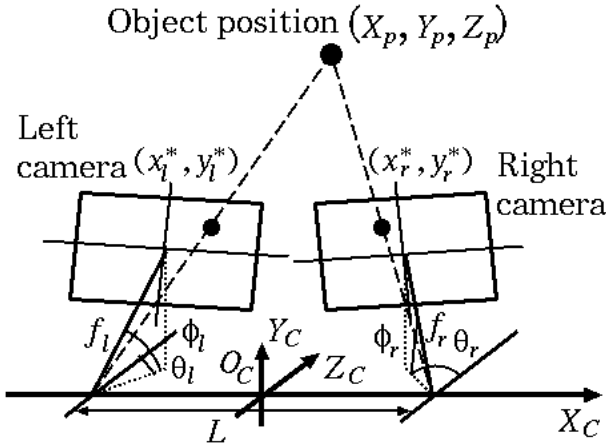


Fig.4 - 3D coordinates in camera coordinate system.

#### Step 1-4. Calculate 3D Coordinates

First, the 3D coordinates  $(X_p, Y_p, Z_p)$  of the object in the camera coordinate system are calculated by using the object location in images  $(x_r^*, y_r^*)$  and  $(x_l^*, y_l^*)$  (Fig.4):

$$X_p = \frac{L}{2\Delta} \left\{ (y_r'^2 x_l'^2 - x_r'^2 y_l'^2) + (z_r'^2 x_l'^2 - x_r'^2 z_l'^2) \right\}, \quad (1)$$

$$Y_p = \frac{L}{2\Delta} \left\{ 2y_r' y_l' (y_r' x_l' - x_r' y_l') + (y_r' z_l' + y_l' z_r') (z_r' x_l' - x_r' z_l') \right\}, \quad (2)$$

$$Z_p = \frac{L}{2\Delta} \left\{ 2z_r' z_l' (z_r' x_l' - x_r' z_l') + (z_r' y_l' + y_r' z_l') (y_r' x_l' - x_r' y_l') \right\}, \quad (3)$$

where,

$$\Delta = (y_r' x_l' - x_r' y_l')^2 + (z_r' x_l' - x_r' z_l')^2 + (z_r' y_l' - y_r' z_l')^2, \quad (4)$$

$$\begin{cases} x_r' = x_r^* \cos \theta_r + f_r \sin \theta_r, \\ y_r' = -x_r^* \sin \theta_r \sin \phi_r + y_r^* \cos \phi_r + f_r \cos \theta_r \sin \phi_r, \\ z_r' = -x_r^* \sin \theta_r \cos \phi_r - y_r^* \sin \phi_r + f_r \cos \theta_r \cos \phi_r, \end{cases} \quad (5)$$

$$\begin{cases} x_l' = x_l^* \cos \theta_l + f_l \sin \theta_l, \\ y_l' = -x_l^* \sin \theta_l \sin \phi_l + y_l^* \cos \phi_l + f_l \cos \theta_l \sin \phi_l, \\ z_l' = -x_l^* \sin \theta_l \cos \phi_l - y_l^* \sin \phi_l + f_l \cos \theta_l \cos \phi_l, \end{cases} \quad (6)$$

$\theta_r$ ,  $\phi_r$  and  $f_r$  are the pan angle, the tilt angle and the focal length of the right camera,  $\theta_l$ ,  $\phi_l$  and  $f_l$  are those of the left camera.

Next, the 3D coordinates  $(X_p^*, Y_p^*, Z_p^*)$  of the object in the robot coordinate system are calculated

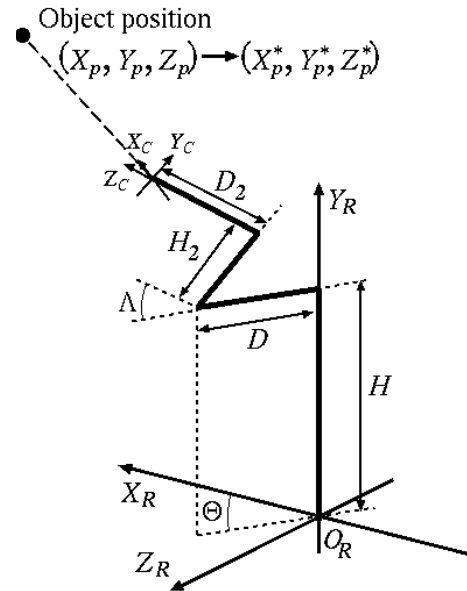


Fig.5 - 3D coordinates in robot coordinate system.

by the following equations:

$$X_p^* = \left( (Z_p + D_2) \cos \Lambda - (Y_p + H_2) \sin \Lambda + D \right) \cos \Theta + X_p \sin \Theta, \quad (7)$$

$$Y_p^* = (Z_p + D_2) \sin \Lambda + (Y_p + H_2) \cos \Lambda + H, \quad (8)$$

$$Z_p^* = \left( (Z_p + D_2) \cos \Lambda - (Y_p + H_2) \sin \Lambda + D \right) \sin \Theta - X_p \cos \Theta, \quad (9)$$

where  $\Theta$  is the robot pan angle,  $\Lambda$  is the robot tilt angle,  $D$ ,  $H$ ,  $D_2$  and  $H_2$  are the lengths of arms of the robot (Fig.5).

The estimated 3D coordinates of the object  $(X_p^*, Y_p^*, Z_p^*)$  are stored in the shared memory.

### Step 1-5. Calculate Camera Rotation Angles

In this paper, the policy of the camera movement is to move the center of the camera image to the location of the object obtained in Step 1-3. By this policy, our system will be able to track the object which moves anywhere.

As the characteristic of this camera, the focal position is corresponding to the center of the rotation. From this characteristics and the location of the object in the input images, the relative rotation angle of the pan and tilt direction ( $\Delta\theta_r$  and  $\Delta\phi_r$ : the right camera) are calculated by the following equations (Fig.6):

$$\Delta\theta_r = \arctan(x_r^*/f_r), \quad (10)$$

$$\Delta\phi_r = \arctan(y_r^*/f_r). \quad (11)$$

In case of the left camera,  $\Delta\theta_l$  and  $\Delta\phi_l$  are also calculated by the same way.

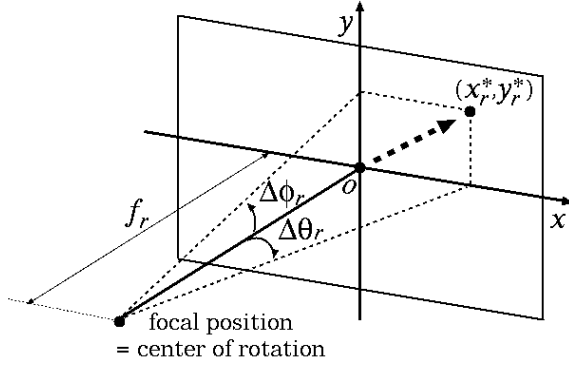


Fig.6 - Camera rotation angle.

### Step 1-6. Move Cameras and Repeat

The right and left cameras are rotated by the calculated pan and tilt angles. And iterate from Step 1-1.

## 3.2 Robot Control Module

In this paper, the policy of the robot movement is to move the front of the robot facing to the object 3D coordinates obtained by the object detection and camera control module. By using this policy, our system will be able to track the object which moves anywhere.

### Step 2-1. Get Data from Shared Memory

The 3D coordinates of the object ( $X_p^*, Y_p^*, Z_p^*$ ) are acquired from the shared memory.

### Step 2-2. Correction of 3D Object Position

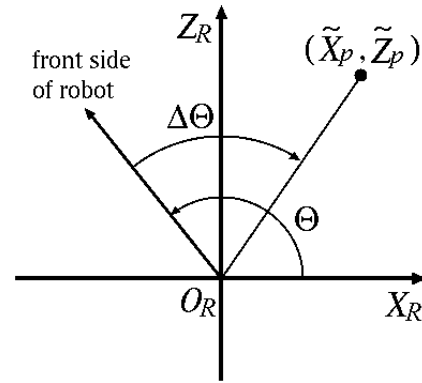
Since the 3D object position ( $X_p^*, Y_p^*, Z_p^*$ ) includes some errors, we correct its coordinates using some of previously estimated 3D object positions. Let ( $X_{pi}^*, Y_{pi}^*, Z_{pi}^*$ ) ( $i=1 \dots n$ ) the previously estimated 3D coordinates,  $t_i$  the time

when 3D coordinates are calculated, we estimate the corrected 3D coordinate ( $\tilde{X}_p, \tilde{Y}_p, \tilde{Z}_p$ ) at the present time ( $t_p$ ) by follows:

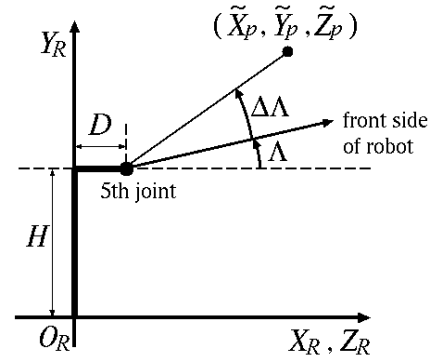
$$\begin{cases} \tilde{X}_p = a_x \cdot t_p + b_x, \\ \tilde{Y}_p = a_y \cdot t_p + b_y, \\ \tilde{Z}_p = a_z \cdot t_p + b_z, \end{cases} \quad (12)$$

where ( $a_x, b_x$ ), ( $a_y, b_y$ ) and ( $a_z, b_z$ ) is calculated from the following equations by the linear least square method:

$$\begin{bmatrix} t_1 & 1 \\ \vdots & \vdots \\ t_n & 1 \end{bmatrix} \begin{bmatrix} a_x \\ b_x \end{bmatrix} = \begin{bmatrix} X_{p1}^* \\ \vdots \\ X_{pn}^* \end{bmatrix}, \quad (13)$$



(a) Pan rotation angle.



(b) Tilt rotation angle.

Fig.7 - Robot rotation angle.

$$\begin{bmatrix} t_1 & 1 \\ \vdots & \vdots \\ t_n & 1 \end{bmatrix} \begin{bmatrix} a_y \\ b_y \end{bmatrix} = \begin{bmatrix} Y_{p1}^* \\ \vdots \\ Y_{pn}^* \end{bmatrix}, \quad (14)$$

$$\begin{bmatrix} t_1 & 1 \\ \vdots & \vdots \\ t_n & 1 \end{bmatrix} \begin{bmatrix} a_z \\ b_z \end{bmatrix} = \begin{bmatrix} Z_{p1}^* \\ \vdots \\ Z_{pn}^* \end{bmatrix}. \quad (15)$$

**Step 2-3. Calculate Robot Rotation Angles**

From the corrected 3D coordinates of the object, the relative rotation angle of the pan and tilt direction of the robot ( $\Delta\Theta$ ,  $\Delta\Lambda$ ) are calculated by the following equations:

$$\Delta\Theta = \arctan\left(\frac{\tilde{Z}_p}{\tilde{X}_p}\right) - \Theta, \quad (16)$$

$$\Delta\Lambda = \arctan\left(\frac{\tilde{Y}_p - H}{\sqrt{\tilde{Z}_p^2 + \tilde{X}_p^2} - D}\right) - \Lambda, \quad (17)$$

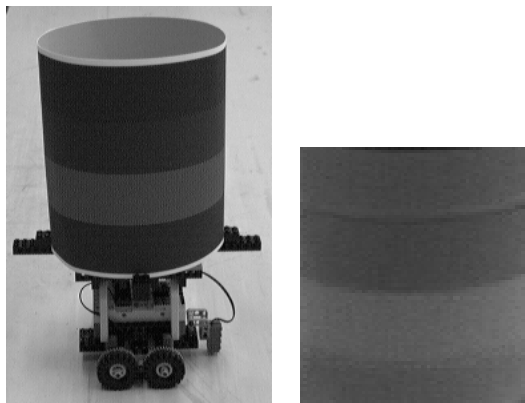
where  $\Theta$  and  $\Lambda$  are current pan and tilt angles of the arm robot, and  $H$  and  $D$  are the height and length at fifth joint (Fig.7).

**Step 2-4. Move Robot and Repeat**

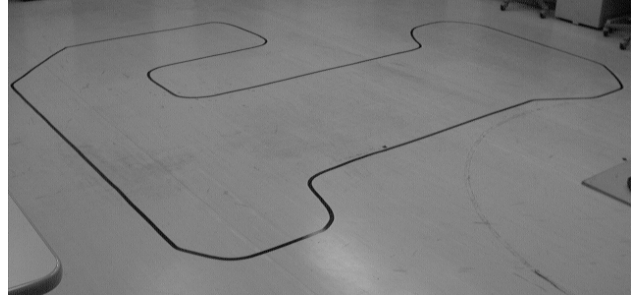
The robot is rotated by the calculated pan and tilt angles. While rotating, the rotation angles are stored in the shared memory one by one. And iterate from Step 2-1.

**4. EXPERIMENTAL RESULTS**

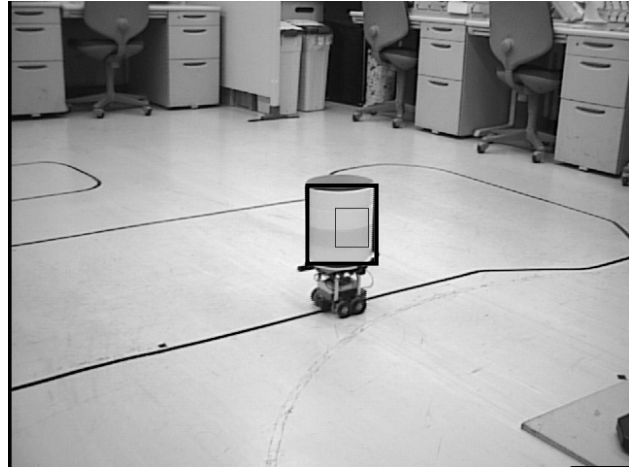
In this experiment, we use a line tracing robot as the object for the tracking and its marker as the reference image of the active search method (Fig.8). This robot moves along the black line course drawn in a floor(Fig.9). Its moving speed is about 70[mm/s]. Fig.10 shows the images of the right and left cameras and the detected object locations in the tracking process. In these figures, the color in the area of detected object (in the black frame) is reversed, and the center points of the grey rectangles in these areas denote the corresponding points for the stereo method.



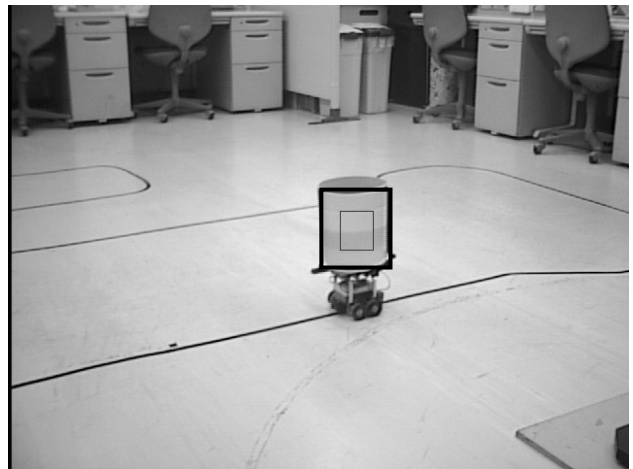
**Fig.8 - Tracking object (line tracing robot) and reference image.**



**Fig.9 - Object moving course.**



**(a) Left camera.**



**(b) Right camera.**

**Fig.10 - Camera images and detected locations.**

Fig.11 and Fig.12 show the 3D trajectories of the tracking object (the line tracing robot). Fig.11 is the bird's-eye view and Fig.12 is the top view of this trajectory. In these figures, the thin line shows the object 3D positions estimated by the object detection module, and the thick line shows the corrected 3D

positions estimated by the robot control module. From these results, almost object 3D coordinates can be obtained along the black line course and this system can track the object.

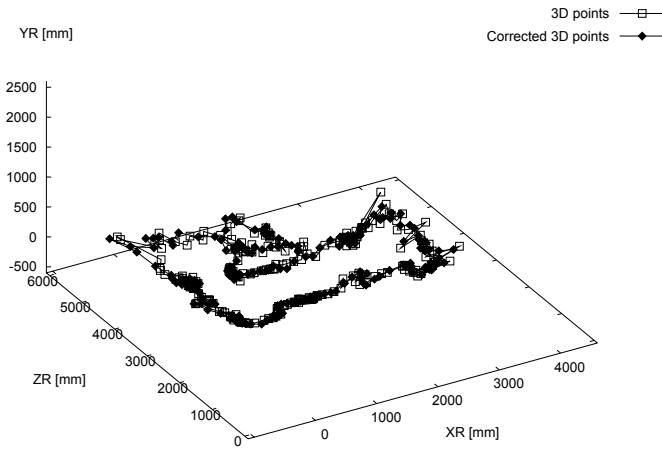
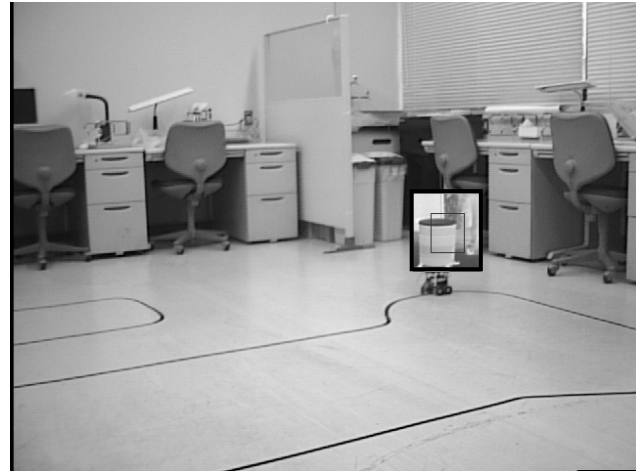


Fig.11 - 3D object trajectory.

In Fig.12, some 3D positions estimated by the object detection module are different from the actual coordinates. This is because that in the Step 1-3 of the object detection module, the corresponding coordinates  $(x_r, y_r)$  and  $(x_l, y_l)$  are not estimated accurately, because the object size is much smaller than the reference image. Fig.13 shows the results of the object detection at one of these locations. We can find that the corresponding positions are not matched in these images. However, in the Step 2-2 of the robot control module, the 3D position is corrected, so the 3D positions are estimated more accurately.



(a) Left camera.



(b) Right camera.

Fig.13 - Case of unmatched corresponding points.

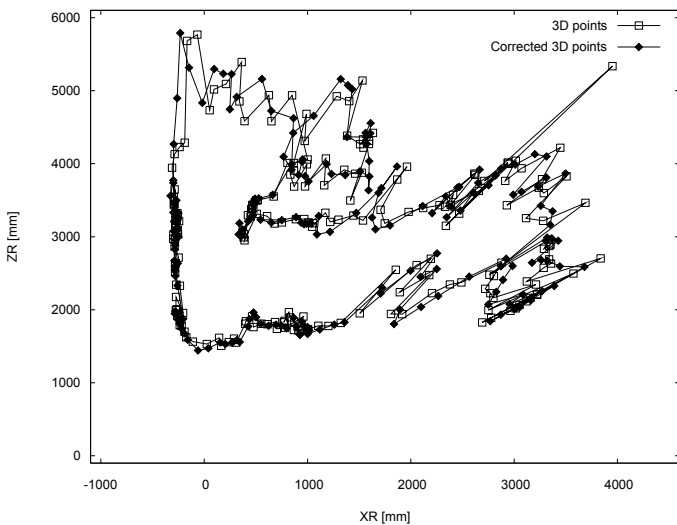


Fig.12 - Object trajectory (top view).

## 5. CONCLUSIONS

In this paper, we propose the high speed and large range object tracking system using two pan-tilt cameras and the arm robot. In this system, we use the active search method to detect the object in the input images. And we configure two modules about the tracking process and execute them in parallel for high speed processing. We also show the effectiveness of the proposed system in the experiments using a line tracing robot.

As the future works, in order to calculate the 3D object positions accurately, we must improve the method to estimate the corresponding points in the left and right camera images. And we will discuss the more fast tracking method using the motion prediction for the object which moves more quickly and we also discuss the method using the adaptive reference images to track the object which has various appearance like a human head.

## 6. REFERENCES

- [1] P.I. Corke, "Visual Control of Robots: high-performance visual servoing," Research Studies Press Ltd., Somerset, England, 1996.
- [2] T. Wada, N. Ukita and T. Matsuyama, "Fixed Viewpoint Pan-Tilt-Zoom Camera and Its Applications," Trans. IEICE D-II, Vol. J81-D-II, No. 6, pp. 1182-1193, Jun. 1998 (in Japanese).
- [3] S. Hiura, K. Murase and T. Matsuyama, "Real Time Object Tracking Using Dynamic Memory," IPSJ Journal Vol. 41, No. 11, pp. 3082-3091, Nov. 2000 (in Japanese).
- [4] V.V. Vinod and H. Murase, "Focused color intersection with efficient searching for object extraction," Pattern Recognition, Vol. 30, No. 10, pp. 1787-1797, Oct. 1997.
- [5] H. Murase and V.V. Vinod, "Fast Visual Search Using Focused Color Matching - Active Search -," Trans. IEICE D-II, Vol. J81-D-II, No. 9, pp. 2035-2042, Sep. 1998 (in Japanese).
- [6] T. Kawanishi and H. Murase, "Quick Object Search Using Color Histograms and Pan-Tilt-Zoom Camera - Dynamic Active Search -," Trans. IEICE D-II, Vol. J84-D-II, No. 8, pp. 1722-1730, Aug. 2001 (in Japanese).
- [7] O. Faugeras, "Three-Dimensional Computer Vision - A Geometric Viewpoint -," MIT Press, Cambridge, 1993.
- [8] B. Nichols, D. Buttler and J.P. Farrell, "Pthreads Programming," Trans. M. Sakaki, O'Reilly Japan, Tokyo, 1998 (in Japanese).
- [9] R. Jain, R. Kasturi and B.G. Schunck, "Machine Vision," McGraw-Hill, Inc., New York, 1995.

**Yoshio Tanimoto** was born in 1961. He received the B.S. and M.S. degrees in electrical engineering from Okayama University, Okayama, Japan in 1984 and 1986, respectively, and the Ph.D. degree in medical engineering from Hyogo University of Teacher Education, Katogun-Yashirocho, Hyogo, Japan, in 1999.



In 1988, he joined the Kibikogen Rehabilitation Center for Employment Injuries, Jobogun-Kayocho, Okayama, Japan, where he is a Rehabilitation Engineer. His current interest is in the area of rehabilitation engineering.



**Hideki Yamamoto** was born in 1944. He received the B.S. degree in electrical engineering from Okayama University, Okayama, Japan, in 1966 and the Ph.D. degree in electrical engineering from Kyoto University, Kyoto, Japan, in 1983.

He is currently a Professor in the Department of Education at Okayama University. His research interests include image processing and signal processing.

Dr. Yamamoto is a member of the IEEE.



**Hiroyuki Ukida** was born in 1969. He received the B.S. and M.S. degrees in information technology from Okayama University, Okayama, Japan in 1992 and 1994, respectively, and the Ph.D. degree in informatics from Kyoto University, Kyoto, Japan,

in 2003.

He is currently an assistant professor in the Department of Engineering at the University of Tokushima. His main research interests are image processing and robot vision.