Proceedings of the 2000 IEEE
International Conference on Robotics & Automation
San Francisco, CA • April 2000

A Human-Robot Interface Using an Interactive Hand Pointer that Projects a Mark in the Real Work Space

Shin Sato Shigeyuki Sakane

Department of Industrial and Systems Engineering
Chuo University
1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-8551 JAPAN

Abstract

A human-robot interface system is under development that takes into account the flexibility of the DigitalDesk approach. The prototype consists of a projector subsystem for information display and a real-time tracking vision subsystem to recognize the human's action. Two levels of interaction using a Virtual Operational Panel and Interactive Image Panel have been developed. This paper presents the third subsystem, the Interactive Hand Pointer used for selecting objects or positions in the environment via the operator's hand gestures. The system visually tracks the operator's pointing hand and projects a mark at the indicated position using an LCD projector. Since the mark can be observed directly in the real work space without monitor displays or HMDs, correction of the indicated position by moving the hand is very easy for the operator. The system enables projection of a mark not only at a target plane with a known height but also to a plane with an unknown height. Experimental results of a pick-and-place task demonstrate the usefulness of the proposed system.

1 Introduction

The human-robot interface is the key to extending the application field for next generation robot systems. Conventional interfaces for industrial robots have been developed in a wide spectrum, ranging from a teaching pendant for on-line teaching to CAD-based robot simulators for off-line teaching. They are, however, too complex and difficult to use for people who are not specialists and for those will use robots out of industry, such as at home, in offices or hospitals. Consequently a more easy-to-use and friendly interface should be developed for various levels of human-robot interaction.

In the field of human-computer interface, much attention has recently been paid to Augmented Reality [1] and Mixed Reality [2], systems which can enhance a human's daily life by blending multi-modal information with the real world. The approach suggests a promising direction for the development of the human-robot interface. By taking into account flexibility of the AR approach, especially the attempt of DigitalDesk [3], we have developed a human-robot interface system. The prototype consists of a projector subsystem for information display and a real-time tracking vision subsystem for recognizing the human operator's actions. Two levels of interaction using a Virtual Operational Panel (VOP) and Interactive Image Panel (IIP) have been developed [4]. This paper presents the third subsystem, an Interactive Hand Pointer (IHP) used for selecting objects or positions in the environment via the operator's hand pointing gestures. The system visually tracks an operator's pointing hand and projects a mark at the indicated position using an LCD projector. Since the mark can be observed directly in the real work space without monitor displays or HMDs, it is very easy for the operator to correct the indicated position by visually monitoring the mark and moving his hand. The system employs a constrained perspective transform for both the modeling of the cameras and the projector. The system enables projection of a mark not only at a plane with a known height but also to a plane with an unknown height. The system also has an option to use a pair of infrared cameras in stereo for reliable detection of the pointing hand.

2 Previous Studies

In recent years, much research has been done on the use of a human's hand gesture for teaching or di-

0-7803-5886-4/00/$10.00© 2000 IEEE

recting robot tasks. Cipolla et al.[5][6] developed a finger pointing system which uses uncalibrated stereo cameras. The system employs a constrained perspective transform for modeling the cameras. It enables an indicated position by the operator's pointing hand to be estimated as an intersection of the pointing lines in the stereo images. Yokokawa et al.[7] developed a system to recognize human pointing gestures. They used color extraction to identify the face and hands as skin color blobs. The system calculates the 3D positions of the face and the hands by stereo tracking. The pointing gesture is recognized based on shape of the hand and direction of the face, and the pointed direction is estimated by direction of the hand. Hayashi et al. [8] used multiple view affine invariance to measure the position and orientation of the hand with respect to the affine basis on the operator's body. The system was applied for moving a mobile robot and a pan-tilter via the operator's hand gestures.

One problem with the above systems is that the operator does not know the results of the gesture recognition until he/she observes a monitor display or the resultant movement of a robot. The operator, in fact, wants to know whether or not he should correct any inappropriate commands before moving the robot. Consequently, an alternative pointing system needed to be developed to enable such flexible human-robot interaction.

3 Interactive Hand Pointer

To overcome the limitations of the previously mentioned pointing systems, we have developed a prototype system named "Interactive Hand Pointer"(IHP). The system has the following features:

(1) It projects a mark continuously at the indicated position in the real work space using a real-time tracking vision system and a projector. Therefore flexible human-robot interaction is achieved in the operator's pointing task.

(2) It uses a constrained perspective transform for both modeling of the cameras and the projector. The determination of the transform matrix for the projector is thus simplified as well as for the cameras.

(3) It enables an indicated position on a plane with an unknown height to be estimated.

(4) It has an option to use a pair of infrared cameras in stereo for highly reliable extraction of the hand shape.

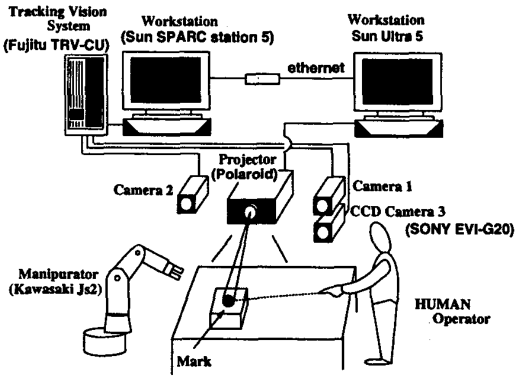

Figure 1: The prototype system

In the following sections, we will explain the features of the system in detail.

3.1 Hardware and Software

Figure 1 shows the hardware of our prototype system. A color tracking vision system (TRV-CU, Fujitsu) [9] tracks template images in real-time based on a block matching algorithm. This tracking system is used to recognize the operator's hand-pointing gesture in the captured images. Live images of the task environment are captured through three CCD cameras (EVI-G20, SONY). Two of the cameras are used to recognize the hand-pointing gesture. As shown in the later section, the system has an option to use a pair of infrared cameras (IR-U300M1, Mitsubishi) in place of the color CCD cameras for reliable detection of the hand gesture.

An LCD (Liquid Crystal Device) projector (COLORVIEW Light, POLAROID) is used to provide information for the operator by projecting a mark at the indicated position in the IHP as well as in the virtual panels of the VOP and IIP [4]. The resolution of the projector is 800 × 600(*pixels*) and the light output is 500 ANSI lumen. The projector is connected to a workstation (Ultra5, Sun) which is an X-window server for the projection. A 6-DOF manipulator (Js2, Kawasaki Heavy Industry) is used for robotic tasks such as object pick-and-place in the environment.

For the software in our prototype system we used EusLisp [10], an object-oriented lisp language developed for robotic applications at the Electrotechnical Laboratory. EusLisp allows the user to call X-window library functions and to integrate the library functions of the devices as lisp functions.

3.2 Estimation of the indicated position

3.2.1 A constrained perspective transform for modeling cameras and a projector

We also use a constrained perspective transform for modeling cameras as in [5]. It permits simple computation for estimating the indicated position based on the two captured images. We can use the perspective transform for modeling the projector as well as modeling the cameras since both devices are regarded as equivalent in the projective geometry, despite the reverse direction of light. This allows us simple and unified computation for estimating the indicated position and projecting a mark at that position.

Let us denote (u, v) for the 2D coordinates of the image plane of a camera or projector and denote (X, Y) for the 2D coordinates of a plane in 3D real world such as a desktop. We represent the perspective transform matrix as $T = \{t_{ij}\}$. We then obtain the following relationship between these coordinates:

$$\begin{pmatrix} su \\ sv \\ s \end{pmatrix} = \begin{pmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{pmatrix} \times \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \qquad (1)$$

where s is a scale factor and $t_{33} = 1$. We need at least four reference points to calculate the eight parameters. We used the four corners of a paper which is observed by the two cameras and is within the projection area of the projector. Then we can determine the transform matrices of the cameras T_{cam1}, T_{cam2} and the transform matrix of the projector T_{prj}.

3.2.2 Tracking the hand and estimating the indicated position

To estimate the lines of a hand-pointing gesture, a tracking vision system [9] based on a block matching algorithm is used. The lines of the pointing gesture are determined by two points: one point P is the fingertip and the other point Q is determined at the base of the finger. Point P is determined as a position of the best correlation using template images of the fingertip. Position Q is at the center of an area that has skin color and is sufficiently far from the point P(Figure 2. In the initial search phase, since we have no information on the positions of the hand, we have to search the whole image for the fingertip using template images with different orientations (Figure 3). We call the initial search a "full search" as in the following. After the initial search, we do not need to search the whole image since we can assume continuity of the position and the orientation of the fingertip during the tracking. Consequently, a set of three templates are adaptively selected for tracking. We call the search in this second phase a "local search".

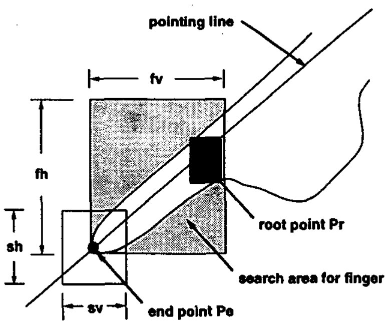

Figure 2: Search area for finger

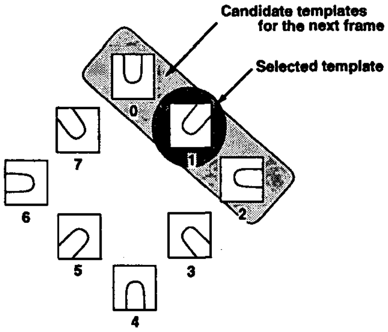

Figure 3: Candidate templates for the next frame

By using T_{cam1}, T_{cam2}, the pointing lines estimated in the image coordinates are transformed to the world coordinates which are assumed to be on a plane. Then a point of intersection (X_0, Y_0) is calculated on the world coordinates. We project this point with the projector based on the matrix T_{prj}. That is, the position input to the projector (u_0, v_0) is calculated by the following equation:

$$\begin{pmatrix} su_0 \\ sv_0 \\ s \end{pmatrix} = T_{prj} \times \begin{pmatrix} X_0 \\ Y_0 \\ 1 \end{pmatrix} \qquad (2)$$

Figure 4 shows the estimation of the hand-pointing lines and the projection of a mark at the indicated position.

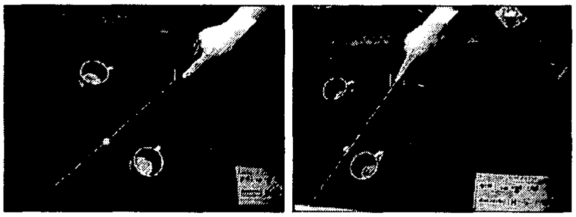

Figure 4: Projection of a mark at the estimated position

3.3 Projecting the mark at a plane with a different height from the base plane

Since the matrices of the perspective transform correspond to a plane (a desktop plane, for example) on which the world coordinates are defined, when we want to point at another plane with different height, the estimation of the indicated position will yield faulty results. To overcome this problem, we developed the following method.

We use two base planes to generate a transform matrix corresponding to planes of various heights. The two base planes have same parameters except for the height. We assume here the height is along the z-coordinate. When we know the height of the target plane, we can generate the matrix by interpolating or exterpolating the two base planes (Figure 5).

(u, v) is the coordinates system of a camera on the base plane, and (X, Y) is the world coordinates system. (u', v') is the coordinate system of a camera for a plane h higher than the base plane. In this case, note that the coordinates (X, Y) are the same though Z is different. When we generate the transform matrix of a plane that is H higher than the base plane, then the new image coordinates of the camera can be expressed by $(u + \frac{H}{h}(u' - u), v + \frac{H}{h}(v' - v))$. Therefore we obtain the following relation:

$$\begin{pmatrix} s(u + \frac{H}{h}(u' - u)) \\ s(v + \frac{H}{h}(v' - v)) \\ s \end{pmatrix} = T_{new} \times \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \qquad (3)$$

Based on a set of four points on the base plane (A, B, C, D) and another set of four points on the higher plane (A', B', C', D'), we can generate the new transform matrix of the cameras and the projector.

For cases when the height of the target plane is unknown, we developed the following algorithm to generate the corresponding transform matrices by extending the above method. As shown in Figure 6, suppose that the true position of the indicated point by the operator is P on the $PlaneT$, and a projected point is Q on the $PlaneT$. If we used correct transform matrices for the $PlaneT$, we could observe that P and Q are the same position in the image plane. But if we use incorrect transform matrices for the $PlaneT$, the transform matrices could correspond to some plane, for example, $PlaneT'$ with incorrect hight. In this case, we could observe the position errors between the indicated position P' and the projected point Q' in the image plane. Therefore we could estimate the height of the target plane by the following algorithm.

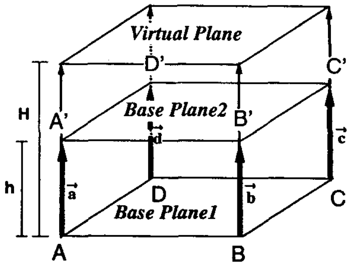

Figure 5: Exterpolating the height of the target plane to modify the perspective transforms

1. Using the method explained in Section 3.2, we estimate the indicated point P' on the $PlaneT'$, initially with $PlaneT' = PlaneB$, and transform them into the coordinates of the image plane.

2. We project the estimated point Q' on the target plane T and calculate the errors E between P' and Q' in the image plane.

3. Based on the equation (3), we calculate the position errors E by changing the height of the $PlaneT'$. Then we fit a quadratic curve to the data to obtain the height which has the minimum position errors.

Theoretically, one camera image is sufficient for estimating the height of target plane. But we used two camera images to estimate the height with more accuracy. We calculated the errors as a sum of the errors in both camera image planes. Figure 7 shows an example of the iterative modification process to deal with the unknown height of the target plane.

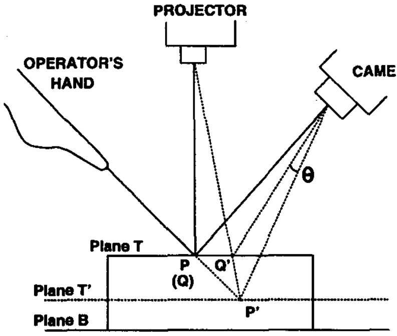

Figure 6: Evaluation of errors between the pointed position P' and the projected mark position Q' which are calculated using a virtual target plane T'

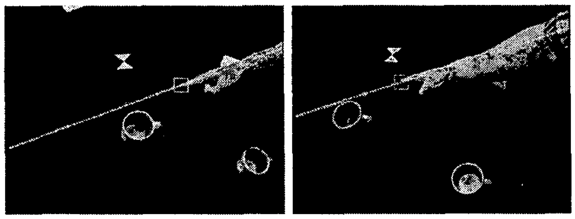

Figure 7: Projection of a mark at a plane of unknown height

4 Experiments

We present two experiments using the prototype system: (1) pointing at a target plane of unknown height and (2) using the hand pointer for a pick-and-place task.

The processing time of the full search for eight template images which correspond to different orientations of a pointing finger is 1.65(*sec*) when the size of the template is 32 × 32(*pixels*) and size of the whole image is 640 × 480(*pixels*). On the other hand, the local search for the three template images requires 20(*msec*) when the size of the template is 62 × 62(*pixels*). Therefore after the initial full search for the hand, the system can track the hand in real-time.

4.1 Pointing at a target plane of unknown height

Figure 8 shows the resulting estimation of the height of the target plane. The vertical axis $Error(pixel^2)$ is the error in the squared distance between a pointed position and a projected mark to the estimated position. More precisely, the value corresponds to the sum of errors in the pair of input images:

$$Error = |P'_{Cam1} - Q'_{Cam1}|^2 + |P'_{Cam2} - Q'_{Cam2}|^2$$

where P'_{Cam1} denotes the position of P' with respect to the image coordinates of $Cam1$. The horizontal axis $Height(mm)$ is the height of a plane that is iteratively modified. The data is approximated by a quadratic curve with a minimum $Error$ value when the $Height$ is close to the actual value. In this case, the actual value was 100.0 (mm) and we obtained the estimated height of 100.5(*mm*) with minimum error.

The processing time for the iterative calculation is about 1.76(*sec*) when changing the height from 0(*mm*) to 200(*mm*) in 5(*mm*) increments. When we need faster processing at the expense of accuracy, we can use 10(*mm*) as the increment which requires only twenty iterations. In this case, the processing time is about 1.21(*sec*). Both processing times include the search processes for the hand position. In fitting a curve to the data, we obtained satisfactory results in estimating the height of a plane even when the first half of the data was given. Thus we could have further reduced the processing time.

Table 1 shows the experimental results of changing the heights of the plane. There are two sources of errors: (1) errors in searching the finger-tip in estimating the indicated point, and (2) errors in determining the transform matrices for the cameras and the projector. In this experiment, the errors caused by factor (1) will be larger than those by factor (2) since the error of one pixel in a full search for finger-tip templates in the image may cause errors of 7(*mm*) in the height of the plane.

4.2 Using IHP for a pick-and-place task

We conducted experiments on a robot task of pick-and-place in which the selection of an object to pick up and the indication of a destination, the position to place the object, are performed by IHP. Figure 9 shows teaching the pick-and-place task. The top scenes show indicating a cup to pick up and the destination using IHP. The bottom scenes show executing the pick-and-place task by the manipulator.

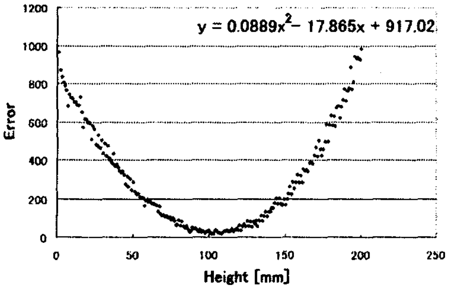

Figure 8: Errors between the projected mark and the estimated position in accordance with iterative modification of the height of the plane.

Table 1: Estimated height of the target plane.

Actual height	Avg	Max	Min	Std.
0	0.8	1	0	0.40
50	46.8	54	42	4.26
100	104.4	111	95	5.64
150	156.8	159	154	1.94

Actual height: actual height of a target plane [mm],
Avg: average value of the estimated height [mm],
Max: maximum value [mm], Min: minimum value [mm],
Std.: standard deviation

The system can project marks in a work space of $890 \times 680(mm)$. In our experiments, we used reference templates sized $fh = fv = 32(pixels)$ for the initial full search. In the local search in the skin color area, we used a search area sized $sh = sv = 62(pixels)$ as shown in Figure 2. Positioning error between the projected mark and the indicated point was $11(mm)$ on average.

The error is caused by the following factors: (1) error in estimating the lines of pointing and (2) error in determining the transform matrices of the cameras and the projector. Though the positioning error may be relatively large, it has little impact on the performance of the interface since the operator can modify the indicating point easily by visually monitoring the mark position which gives a continuous feedback to the operator from the system.

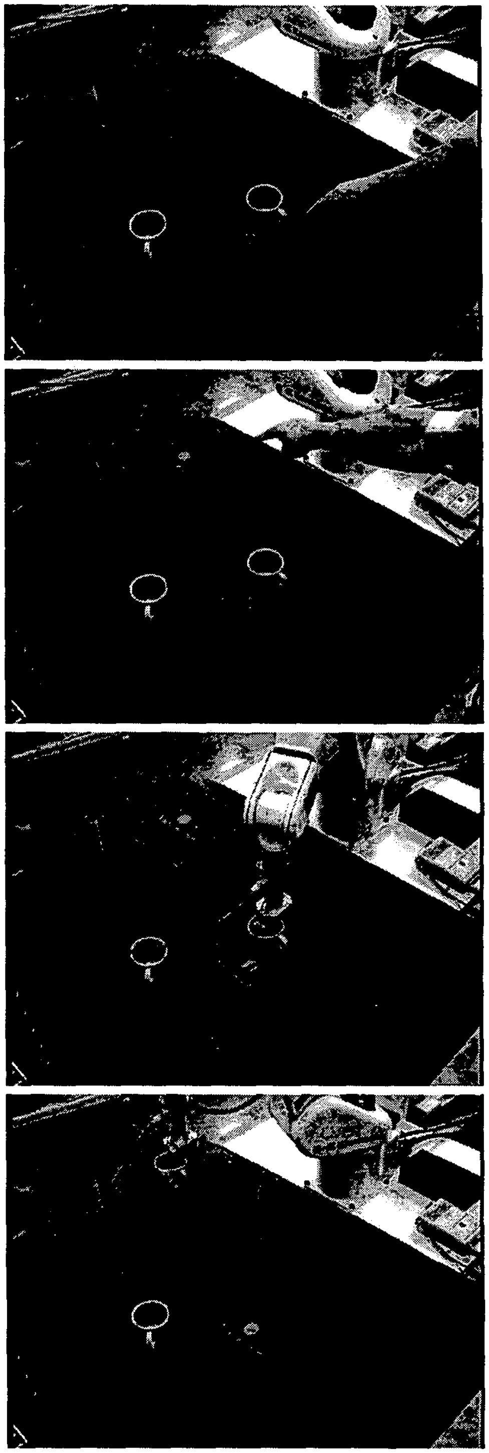

Figure 9: A pick-and-place task using IHP

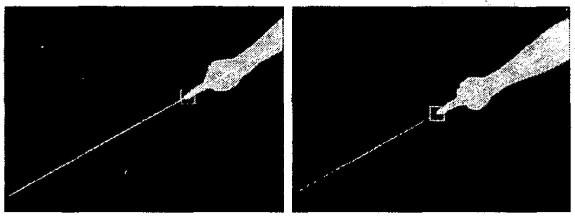

Figure 10: Reliable detection of operator's hand using a pair of infrared cameras

4.3 Use of infrared cameras in place of color CCD cameras

In a task environment where the color of the target plane might be similar to skin color or when the skin color may vary during the task, the reliability of detecting the operator's hand could decrease. To deal with this problem, the system has an option for using a pair of infrared cameras in place of the color CCD cameras. Since an infrared camera yields NTSC signal output, extension of the system is feasible. Figure 10 shows scenes where the operator's hand is detected using a pair of infrared cameras. We can obtain the clear shape of the hands with suppressed colors and textures in the normal image as shown in Figure 10.

The infrared cameras, however, cannot be used for pointing at a plane of unknown height since the projected mark is invisible in the infrared image. They can be used for pointing at a plane of a known height such as the base plane of the desk. Therefore a cooperative use with the color CCD cameras may be required depending on the task.

5 Conclusions

We presented an Interactive Hand Pointer for a human-robot interface that enables objects or positions in a task environment to be specified by an operator's pointing gesture. Since the system projects a mark at the indicated position using an LCD projector and the mark can be observed directly in the real work space, it is very easy for the operator to correct the indicated position by moving his/her hand. Flexible human-robot interaction by the pointing is thus achieved. The system enables the projection of a mark not only at a target plane of known height but also at a plane of unknown height. The experimental results using the IHP for a pick-and-place task confirm the usefulness of our proposed system.

Our plans for future research include integration of active range finder functions and cooperative use of other media such as a voice interface.

References

[1] R.T.Azuma: "A survey of Augmented Reality," *PRESENCE*, Vol.6, No.4, pp.355-385,1997.

[2] Y.Ohta, H.Tamura: Mixed Reality -Merging Real and Virtual Worlds, Springer, 1999.

[3] P.D.Wellner: "Interacting with Paper on the DigitalDesk," *Communications of the ACM,* Vol.36, No.7, pp.86-97, 1993.

[4] M.Terashima, S.Sakane: "A Human-Robot Interface Using an Extended Digital Desk" *Proc. IEEE Int. Conf. Robotics and Automation,* pp. 2874-2880, 1999.

[5] R.Cipolla, P.A. Hadfield, and N.J. Hollinghurst: "Uncalibrated Stereo Vision with Pointing for a Man-Machine Interface.", *Proc. IAPR Workshop on Machine Vision Applications, Kawasaki,* pp.163-166,1994.

[6] N.J.Hollinghurst: "Uncalibrated Stereo and Hand-Eye Coordination", Ph.D Thesis, University of Cambridge, 1997.

[7] T.Yokokawa, T.Mori, and T.Sato: "Recognition of Human Pointing Gesture Based on Color Extraction and Stereo Tracking" *Proc. of the 15th Annual Conference of the RSJ,* pp. 997-998, 1997 (in Japanese).

[8] K.Hayashi, Y.Kuno, and Y.Shirai: "Pointing Gesture Recognition System Permitting User's Freedom of Movement,"*Proc. Workshop on Perceptual User Interfaces,* pp. 16–19, 1997.

[9] T.Morita, N.Sawasaki, T.Uchiyama, and M.Sato: "Color Tracking Vision System" *Proc. of the 14th Annual Conference of the RSJ,* pp. 279-280, 1996 (in Japanese).

[10] T.Matsui: "Multithread Concurrent Object- Oriented Language EusLisp with Geometric Modeling Facilities" *Journal of the RSJ,* Vol. 14, No.5, pp. 650-654, 1996 (in Japanese).