Effective and efficient 3D object location using automatic camera calibration

Wei Xiong, Yi Jin and Ming Xie

Robotics Research Center
School of Mechanical and Production Engineering
Nanyang Technological University, Singapore

ABSTRACT

Locating an object is an essential work in a robot system. In this paper, we present a method designed for the effective and efficient object location using the stereo vision. The 2D image information from the uncalibrated stereo cameras is used firstly to control the robot hand to approach the target object. Subsequently, the cameras are calibrated automatically using the information from the robot hand movement and therefore to have the robot reach the destination rapidly and precisely. By doing this, we can overcome the limitations of the conventional camera calibration method and achieve higher efficiency and accuracy than the uncalibrated camera methods. Some key issues such as robust 3D and 2D points mapping are addressed. Experimental results are given to show the effectiveness of the proposed ideas.

1. INTRODUCTION

An essential task in a vision guided robot system is to discover the 3D world coordinates from 2D image coordinates, i.e. to locate an object in 3D world based on the 2D image information. One of the most common ideas is to establish the mapping relationship between the 3D world coordinates and 2D image coordinates by camera calibration [1,2,3,4,]. In the classical method of calibration, a calibration object (a special designed pattern) is put in the view of the cameras to get some corresponding points between 3D world and 2D images. The corresponding points are often obtained manually. The method can produce a quite accurate estimate to camera parameters if the pattern is carefully set and the corresponding points are precisely observed. Although the method has been widely used, it has some drawbacks. An accurate calibration object is made expensively both in cost and in time; the manual set of this object and corresponding points in images affect the accuracy of the calibration; the method lacks of flexibility because the camera parameters such as camera orientation and focal length is no longer changeable after calibration; in some applications of robot, using a calibration object may not be possible in working locale. To solve these problems, Luong and Faugeras develop a self-calibration method[5]. The self-calibration method does not need the calibration pattern and can autonomously calibrate cameras given enough point correspondences can be established between images. However, the method calibrates cameras up to a scale factor. This is not sufficient for our actual control of robot movement. In case of that the cameras are movable and controllable, some fully autonomous camera calibration methods have been developed [6,7,8]. In our system, the cameras are not actively controlled by camera motion planning. Therefore, we can not count on the information from active vision.

In our previous work, a solution of robot hand-eye coordination with uncalibrated cameras was proposed [9]. Direct discovery from uncalibrated cameras is an interesting topic [10,11,12]. It has the advantage of being flexible. However, the method can not give precise location of an object in 3D world and therefore can not control the robot to capture the object rapidly. In fact, the uncalibrated methods can only approach the target object gradually.

In this paper, we propose a method that uses both the uncalibrated idea and the calibrated idea to locate an object in 3D world both flexibly and rapidly. At beginning, the 2D image information from the uncalibrated stereo cameras is used to control the robot hand to approach the target object. While the robot is moving, the cameras are calibrated automatically and then to have the robot reach the destination rapidly and precisely. The basic idea is to calibrate "robot eyes" by using robot hand. By doing this, we can overcome the limitations of the conventional camera calibration method and achieve higher efficiency and accuracy than the uncalibrated camera methods. Our discussion will consists of five parts. In the first part, we briefly introduce how to use the uncalibrated image information to move the robot towards the target object. In the second part, we present the detection of robot hand and establish the correspondence between 3D and 2D coordinates. In the third part, we describe the camera calibration from the correspondence and the computation from the projection matrix. In the fourth part, we discuss

This work is supported by the ARC1/94 project.

the region-based image mapping and finally we will give a conclusion and further work discussion.

2. VISUAL SERVO WITH UNCALIBRATED CAMERAS

Let $\mathbf{P_r}=(X_r, Y_r, Z_r)^T$ denote the hand tip of a service robot in 3D real world. $\mathbf{P_o}=(X_o, Y_o, Z_o)^T$ denotes the position of a target object in the same 3D world. $\mathbf{I_r^l}=(u_r^l, v_r^l)^T$ and $\mathbf{I_r^r}=(u_r^r, v_r^r)^T$ denote the projections of $\mathbf{P_r}$ on the left and right images respectively. Similarly, $\mathbf{I_o^l}=(u_o^l, v_o^l)^T$ and $\mathbf{I_o^r}=(u_o^r, v_o^r)^T$ are the projections of $\mathbf{P_o}$ on the left and right images. Our task is to control the service robot hand approach/grasp the target object based on the visual information from two cameras. The target object may be static or moving.

With the uncalibrated cameras, the basic problem is to compute the distance/orientation of $\mathbf{P_r}$ and $\mathbf{P_o}$ directly from $\mathbf{I_r^l}$, $\mathbf{I_r^r}$, $\mathbf{I_o^l}$, $\mathbf{I_o^r}$. Let $\Delta\mathbf{P}=(X_r-X_o, Y_r-Y_o, Z_r-Z_o)^T$ and $\Delta\mathbf{I}=(u_r^l-u_o^l, v_r^l-v_o^l, u_r^r-u_o^r, v_r^r-v_o^r)$. As we know, $\Delta\mathbf{P}$ and $\Delta\mathbf{I}$ are related by following equation:

$$\Delta P = G \cdot J \cdot \Delta I$$

where $\boldsymbol{G}$ is control gain matrix and $\boldsymbol{J}$ is the mapping matrix. The further mapping between $\Delta\mathbf{P}$ and the robot's joint space can be done if we know the forward and inverse kinematics of the robot arm. In this paper, we are only concerned about the mapping between $\Delta\mathbf{P}$ and $\Delta\mathbf{I}$.

Let $\{\mathbf{P_i}$, i=1,2,3,4$\}$ be physical points in 3D world , $\{\mathbf{I_i^l}, \mathbf{I_i^r}$, i=1,2,3,4$\}$be the corresponding points in left and right images, where $\mathbf{I_i^l}=(u_i^l, v_i^l)^T$, $\mathbf{I_i^r}=(u_i^r, v_i^r)^T$. From the four points: $\mathbf{P_1}=(1,0,0,1)^T$, $\mathbf{P_2}=(0,1,0,1)^T$, $\mathbf{P_1}=(0,0,1,1)^T$, $\mathbf{P_1}=(0,0,0,1)^T$, it can be derived that [9]:

$$\Delta I = N \cdot \Delta P$$

where

$$N = \begin{bmatrix} u_1^l - u_4^l & u_2^l - u_4^l & u_3^l - u_4^l \\ v_1^l - v_4^l & v_2^l - v_4^l & v_3^l - v_4^l \\ u_1^r - u_4^r & u_2^r - u_4^r & u_3^r - u_4^r \\ v_1^r - v_4^r & v_2^r - v_4^r & v_3^r - v_4^r \end{bmatrix}$$

It is then straightforward to derive the following control equation to generate the control signal $\Delta\mathbf{P}$:

$$\Delta P = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \cdot J \cdot \Delta I$$

with

$$J = c \cdot \left(N^T N\right)^{-1} \cdot N^T$$

where c is a constant, (k_x,k_y,k_z) are control gains determined in the following way:

$$k_x = k_y = k_z = \min\{\ 0.1, \frac{1}{\|\Delta I\|}\} \cdot$$

Based on the derivation , we can control the robot arm to approach the target object without camera calibration. Without knowing the actual 3D coordinates of robot hand and target object, the method only allows robot come near the target gradually in small steps, as shown in Figure 1. The discovered $\Delta\mathbf{P}$ is up to a scale and therefore the control of the robot hand is actually in a testing manner. To achieve a more efficient and precise service robot control, some camera parameters are necessarily computed.

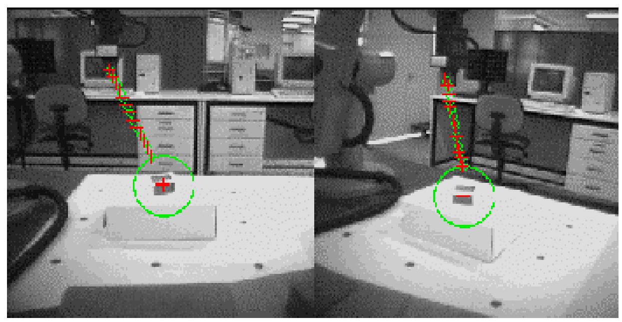

Figure 1. Visual servoing without camera calibration

Several methods have been reported for camera calibration [1,2,3,4]. The methods are not flexible as we mentioned in last section. In this paper, we propose to initially use the uncalibrated cameras to control the robot hand moving toward the target object. During the robot moving, we automatically calibrate the two cameras by using the robot arm as a calibration object. Once we get enough information to compute the camera parameters, we can make use of the calibrated information to control the robot in a more efficient and precise way.

In next section, we discuss the automatic robot hand detection in the left and right images. From the 2D information of the robot hand, we can calibrate the cameras in section 4.

3. ROBOT HAND DETECTION

One main difficulty of carrying out on-line camera calibration is to robustly and precisely detect the points of the image. The points are used for building up 2D and 3D correspondence. In our system, we detect the moving robot hand to obtain a series of mapping points. For the sake of robustness and convenience, we paste some man-made labels on the robot hand, as shown in

Figure 2. Our purpose is to locate the crossing points on the robot hand. The advantage of using crosses is that the figures are invariant to the camera zoom factor and image resolution. The crosses are also convenient to locate points precisely.

We divide the cross figure into two parts: *in-cross region* (the black part) and *out-cross region* (the yellow part).

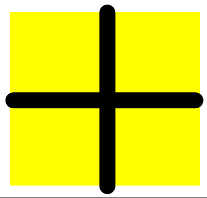

Figure 2. Robot hand with labels

We compute the mean value of the in-cross pixels (M_{in}) and the mean value of the out-cross pixels (M_{out}). Let V_{in} denote the variance of the pixels in in-cross region and Vout denote the variance of the pixels in out-cross region. A point is interesting if $|M_{in}-M_{out}|$ is large and V_{in} , V_{out} are small. This idea can be used to detect any binary pattern detection.

After selecting a group of interesting candidates, the colors of the in-cross and out-cross regions are used to verify the underlying points. For each of the interesting points, we compute the symmetry of the pixels around it. The pixels with small symmetric values are the possible crossing points. A coarse to fine method is used to speed up the detection procedure. A detection result is shown in Figure 3.

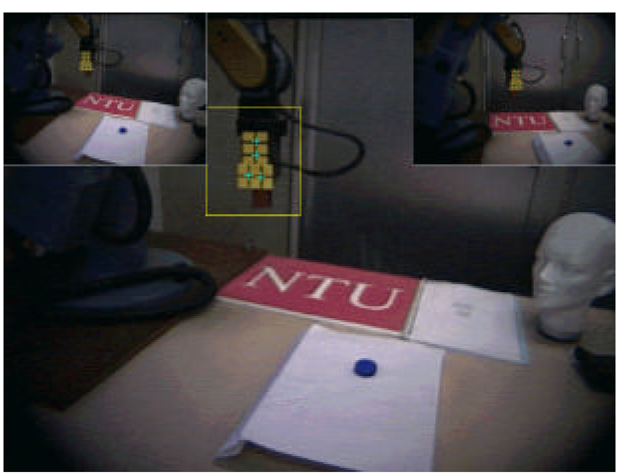

Figure 3. Detection of crossing points on the robot hand

Because of image noise, lighting variance, observing angle change, and the degradation in the process of image formation, it is hard to guarantee that all detected points are the crossing points on the robot arm and no such point is missing. We use the geometric information to check the obtained points. Only the points with correct geometric distribution are accepted. In our example, two of the four crossing points should be approximately on a vertical line and two on a horizontal line. The distance among accepted points should be in a reasonable range. Using these ideas, some outliers are avoided.

4. AUTOMATIC CAMERA CALIBRATION

A common used pinhole model is used as our camera model, as shown in Figure 4. The model implies that the relationship between the world coordinates and the pixel coordinates is linear projective.

Projective transformation matrix (M) is used to describe the mapping from (X,Y,Z) to (u,v).

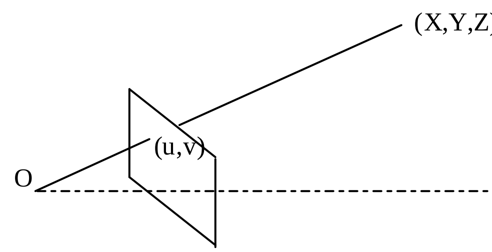

Figure 4. Pinhole camera model

$$Z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = M_{3\times 4} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

The projective matrix contains the intrinsic and extrinsic parameters of a camera. The matrix can be computed from six or more mapping pairs.

Let $\mathbf{I}_{ri}^{l} = (u_{ri}^{l}, v_{ri}^{l})^T$ be the projection point of the robot arm at $\mathbf{P}_{ri}$ on left image, where i=1,2,…, n and n≥6. Suppose: $m=(m_{11},m_{12},m_{13},m_{14},m_{21},m_{22},m_{23},m_{24},m_{31},m_{32},m_{33})^T$, $m_{34}=1$, $U=(u_{r1}^{l}, v_{r1}^{l}, \ldots, u_{rn}^{l}, v_{rn}^{l})^T$ and,

The projective matrix of left camera can be solved from

$$K = \begin{bmatrix} X_{r1} & Y_{r1} & Z_{r1} & 1 & 0 & 0 & 0 & 0 & -u_{r1}^{l}X_{r1} & -u_{r1}^{l}Y_{r1} & -u_{r1}^{l}Z_{r1} \\ 0 & 0 & 0 & 0 & X_{r1} & X_{r1} & X_{r1} & 1 & -v_{r1}^{l}X_{r1} & -v_{r1}^{l}Y_{r1} & -v_{r1}^{l}Z_{r1} \\ .. & .. & .. & .. & .. & .. & .. & .. & .. & .. & .. \\ X_{rn} & Y_{rn} & Z_{rn} & 1 & 0 & 0 & 0 & 0 & -u_{rn}^{l}X_{rn} & -u_{rn}^{l}Y_{rn} & -u_{rn}^{l}Z_{rn} \\ 0 & 0 & 0 & 0 & X_{rn} & Y_{rn} & Z_{rn} & 1 & -v_{rn}^{l}X_{rn} & -v_{rn}^{l}Y_{rn} & -v_{rn}^{l}Z_{rn} \end{bmatrix}$$

the following linear equations:

$$Km = U$$

Using least mean square method, we have

$$m = \left(K^T K\right)^{-1} K^T U$$

Similarly, we can compute the projective matrix of right camera using the corresponding points of robot arm on the right image.

After camera calibration, we can easily compute the 3D coordinate of a point from its projections on left and right images. Suppose we have known the left and right retinal coordinates $(u^l,v^l)^T$ and $(u^r,v^r)^T$. The world coordinates $(X,Y,Z)^T$ can be obtained from the following equations using the least mean square method.

$$\left(u^l m^l_{31} - m^l_{11}\right)X + \left(u^l m^l_{32} - m^l_{12}\right)Y + \left(u^l m^l_{33} - m^l_{13}\right)Z = m^l_{14} - u^l m^l_{34}$$
$$\left(v^l m^l_{31} - m^l_{11}\right)X + \left(v^l m^l_{32} - m^l_{12}\right)Y + \left(v^l m^l_{33} - m^l_{13}\right)Z = m^l_{14} - v^l m^l_{34}$$
$$\left(u^r m^r_{31} - m^r_{11}\right)X + \left(u^r m^r_{32} - m^r_{12}\right)Y + \left(u^r m^r_{33} - m^r_{13}\right)Z = m^r_{14} - u^r m^r_{34}$$
$$\left(v^r m^r_{31} - m^r_{11}\right)X + \left(v^r m^r_{32} - m^r_{12}\right)Y + \left(v^r m^r_{33} - m^r_{13}\right)Z = m^r_{14} - v^r m^r_{34}$$

In next section, we discuss how to get $(u^l,v^l)^T$ and $(u^r,v^r)^T$ for the target object.

5. REGION-BASED CORRESPONDENCE

Our goal is to find the projections of the target image in left and right images by image mapping and then compute the 3D coordinates of the object using the calibrated information. Mapping image points is very difficult and unreliable in real complex images. People have reported various methods but the problem still exists. Instead of mapping image points, we discuss a region-based method for computing image correspondence. As we know, the projected regions in left and right images are different in shape because of the different intrinsic and extrinsic parameters of the cameras. It is therefore difficult to compare the regions directly from the different camera images. We propose a novel algorithm that can constructs regions for comparison.

Assume the front face of the target object is planar and two vectors from an interesting point are known. In Figure 5, X_0 is the interesting point we are looking for and vector $\overrightarrow{X_0X_1}$, $\overrightarrow{X_0X_2}$ are known prior.

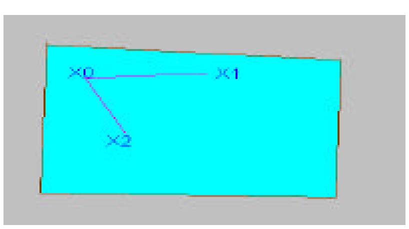

Figure 5. Two vectors are known prior.

Suppose

$$\overrightarrow{X_0X_1} = \left(V_{X1} \quad V_{Y1} \quad V_{Z1}\right)^T; \ \overrightarrow{X_0X_2} = \left(V_{X2} \quad V_{Y2} \quad V_{Z2}\right)^T.$$

Our algorithm is as following.

- For a given point on image one, compute the epipolar line to get the possible correspondence on the other image;
- From the given point and each point on the epipolar line, compute X_0 in the real world.
- From X_0 and $(Vx1, Vy1, Vz1)^T$, $(Vx2, Vy2, Vz2)^T$, compute X_1 and X_2;
- For every point in the region defined by X_0, X_1, and X_2, compute the corresponding points in image one and image two; The two sets of corresponding pixels form two regions respectively.
- Compute the difference between the two c*onstructed* regions;
- The point on the epipolar line that gives the minimum difference is considered as a **true** mapping.

In Figure 6, suppose we are trying to locate a box. It's obvious that the regions of box in left and right images are quite different in shape. Instead of directly comparing the regions from left and right images, we match the two constructed regions, as shown at the left side of the figure. The pixels in the blue shadowed part are used for constructing the regions.

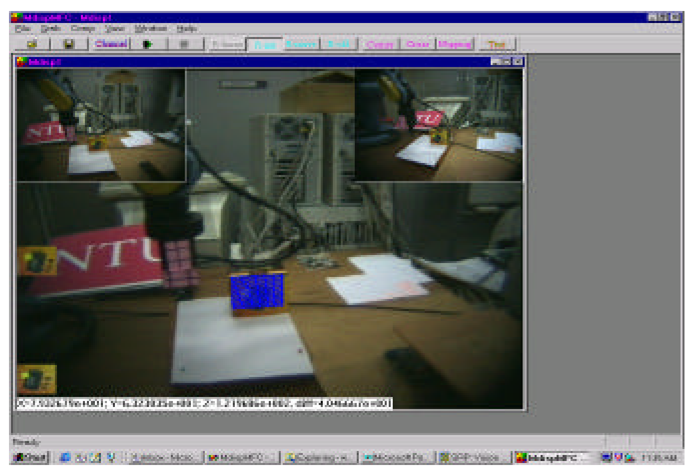

Figure 6. Region-based image mapping.

6. CONCLUSION

We have presented in this paper the visual control of our humanoid service robot. A new concept of integrating the uncalibrated cameras method and the calibrated cameras method is proposed. For the purpose of automatic camera calibration, we present a method to reliably locate the robot arm. We have also introduced a new algorithm of region-based image matching. All of the ideas discussed in this paper have been implemented. Our experimental results have showed their effectiveness .

7. REFERENCES

[1] R. Y. Tsai, "An efficient and accurate camera calibration technique for 3D machine vision", *Proceedings of CVPR'86 (IEEE Computer Society Conference on Computer Vision and Pattern Recognitio)*, Miami Beach, FL, pp. 364-374, 1986.

[2] O. D. Faugeras and G. Toscani, "The calibration problem for stereo", *Proceedings of CVPR'86*, pp.15-20, 1986.

[3] M. H. Han and S. Rhee "Camera calibration for three dimensional measurement", *Pattern Recognition*, vol. 25, no. 2, pp.155-164, 1992.

[4] T. Echigo, "Camera calibration technique using three sets of parallel lines", *Machine Vision and Applications*, vol. 3, no. 3, pp.159-167, 1990.

[5] Q. T. Luong and O. D. Faugeras, "Self-calibration of a moving camera from point correspondences and fundamental matrices", *International Journal of Computer Vision*, vol. 22, no. 3, pp.261-289, 1997.

[6] S. D. Ma, "Self-calibration technique for active vision systems", *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp.114-120, 1996.

[7] R. Y. Tsai and R. K. Lenz, "A New Technique for fully autonomous and efficient 3D robotics hand/eye calibration", *IEEE Journal of Robotics and Automation*, vol. 3, no. 3, pp.345-358, 1989.

[8] G. Q. Wei, K. Arbter, and G. Hirzinger, "Active self-calibration of hand cameras and hand-eye relationships with motion planning", *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1359-1364, 1997.

[9] M. Xie, "Robotic hand-eye coordination: new solutions with uncalibrated stereo cameras", *Machine Vision and Applications*, vol. 10,, pp.136-143, 1997.

[10] W. Y. Yau, and H. Wang, "Robust hand-eye coordination", *Advanced Robotics*, vol. 11, no., 1, pp. 57-73, 1997.

[11] J. Huber, and V. Graefe, "Motion stereo for mobile robots", *IEEE Transactions on Industrial Electronics*, vol. 41, no. 4, pp. 378-383, 1994.

[12] N. Hollinghurst, and R. Cipolla, "Uncalibrated stereo hand-eye coordination", *Image and Vision Computing*, vol. 12, no. 3, pp. 187-192, 1994.