Head-Eye Coordination: A Closed-Form Solution *

M. Xie
School of Mechanical & Production Engineering
Nanyang Technological University, Singapore 639798
Email: mmxie@ntuix.ntu.ac.sg

ABSTRACT

In this paper, we present a novel solution for implementing Head-Eye coordination which is an intelligent human behavior. We develop the principle of determining a third unknown view from two known views and apply it to solve Head-Eye coordination problem. The advantages of our solution are: (a) a closed-form solution is obtained and (b) "one-step" or "two-step" control algorithm can be implemented in a straightforward way.

KEYWORDS: Head-Eye Coordination, Visual Guidance.

1. Introduction

It is commonly believed that robotics research reaches the stage of looking into the aspect of intelligence which will be the key issue to make industrial/service robots more user-friendly and hence more useful than current cases. The challenge here is how to define machine intelligence that we want to replicate on robots. The perception of intelligence by human beings is quite subjective. It may be difficult to objectively state the definition of machine intelligence. Then, the question is whether it is still possible to build intelligent robot without knowing precisely what we mean by robot's intelligence ? (it is easy to say that an intelligent system must be reactive, sensitive, and have certain ability to understand and to communicate, etc). We believe that one possible way of developing intelligent robot is to make robot imitate nature or human beings.

Human beings is a complex and intelligent system. There are some basic human behaviors which are relevant to robotics. These include:

- Hand-Eye Coordination (e.g., picking up and placing object by hands being guided by eyes).
- Head-Eye Coordination (e.g., looking and searching object of interest with controlled head motion).
- Leg-Eye Coordination (e.g., walking around a 3D environment by legs being guided by eyes).
- Hand-Hand Coordination (e.g., manipulating objects with two hands).

*This work is supported by the project ARC1/94 funded by the Ministry of Education (Singapore)

- Full-Body Coordination (e.g., playing sports like basketball).
- and others.

Current R&D in robotics has already moved into the direction of imitating nature or human beings. In the area of Hand-Eye coordination with robotic system [4], extensive research works have been done so far (see [2] for a detailed survey). As for Head-Eye coordination (sometime called image-based visual servoing or misclassified as Hand-Eye coordination), only highly iterative approaches have been developed (see [2] and [6]). These approaches are based on the use of specially designed minimization process to generate motion control signal which guides head to a position where the image seen by eyes (the observation) looks like a reference image (the goal). No other investigation has been made to see whether a non-iterative and thus deterministic approach exists and is able to fulfill Head-Eye coordination with robotic system.

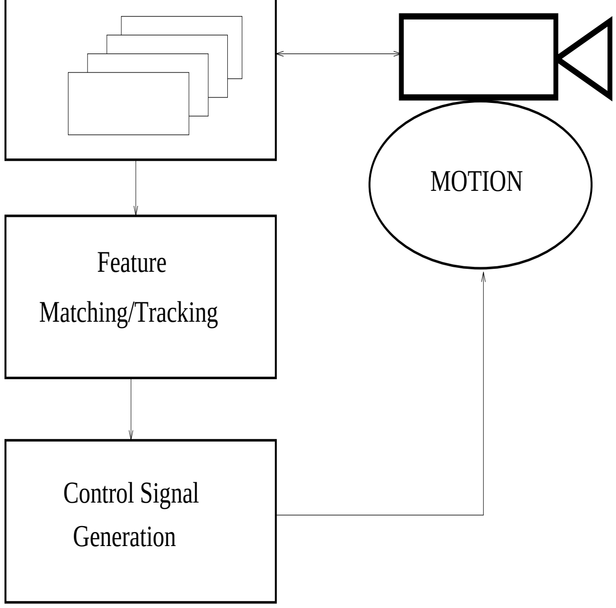
In this paper, we present a novel method for implementing Head-Eye coordination. The principle underlying our method is the determination of a third unknown view from two known views. The application of this principle to Head-Eye coordination allows to implement either "one-step" or "two-step" control algorithm. In this way, high speed and high accurate Head-Eye coordination can be realized because no iteration is involved in the coordination process. Head-Eye coordination is an intelligent human behavior and is also an important problem of robotics. Its importance in robotics can be explained by some practical applications such as:

- Automatic Car Following (special case of object tracking):
  Automatic car following is an useful function for a smart car system. If two cars (A, B) driving on a highway will go to a same destination, car B can just follow car A if the driver of car B wishes to do so. This can be achieved by maintaining observed image of car A seen by the vision system mounted on car B to be constant. In this case, the vision system plays the role of eyes and car B (three degrees of freedom) plays the role of head.
- Automatic Loading/Unloading of Containers (special case of picking up objects placed on a ground):
  To pick up a container placed on the ground of a port, the problem is to guide the picking-up device to a position on top of a container. If we consider the picking-up device (four degrees of freedom: one rotation and three translations) as a head and the vision system mounted on it as eyes, the problem of loading/unloading containers becomes a Head-Eye coordination problem.
- On-Fly Assembly (special case of object tracking and manipulation):
  The current practice for assembly task (harddisk or PCB assembly) is to (temporarily) stop assembly parts and move them to next station after the completion of assembly. It is possible to achieve on-fly assembly while assembly parts are in motion with conveyor, if we can maintain a zero motion between assembly robot and a conveyor. This can be done by mounting robot's base on a mobile platform (considered as a Head of three degrees of freedom: one rotation and two translations) and installing a vision system (considered as eyes) on the mobile platform. Then, the problem of maintaining a zero motion between assembly robot and a conveyor becomes a Head-Eye coordination problem.

Therefore, we believe that Head-Eye coordination is an interesting research topic in robotics. In the following sections, we state the generic problem of Head-Eye coordination and present a closed-form solution. Experimental results are given to support the validity of our method.

2. Problem Statement

As illustrated by Fig.1, a Head-Eye coordination system is generally composed of a vision system (eyes) mounting on a mobile platform (head). The vision system can have one or two cameras and the mobile platform may have up to six degrees of freedom.



(a) Stereovision on mobile robot

(b) Monocular vision on robot arm

Figure 1: Generic configuration of Head-Eye coordination system that can be composed of (a) stereovision mounted on a mobile robot, (b) monocular vision mounted on a robot arm, or other configurations.

At a particular position (called Reference View), the vision system can capture an image as reference (called Reference Image). Fig.2 shows one example that a camera is at a reference view position and a reference image is taken.

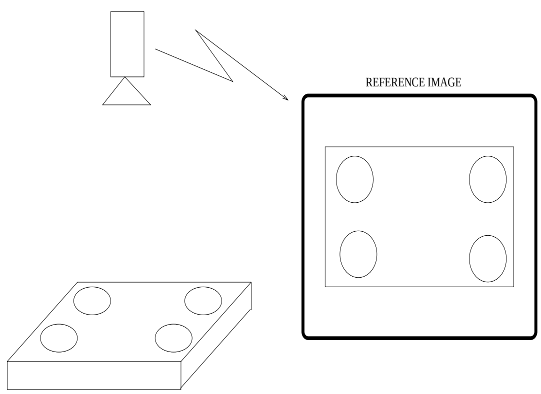


Figure 2: Reference View and Reference Image.

During real operation, the camera(s) may be at any position (as shown in Fig.3) other than the reference view position. Therefore, the observed image is different than the reference image.

Now, the problem of Head-Eye coordination can be stated as follows: *How to control the mobile platform (head) which will bring the camera(s) (eyes) to a position where the observed image is equal to the reference image ?*

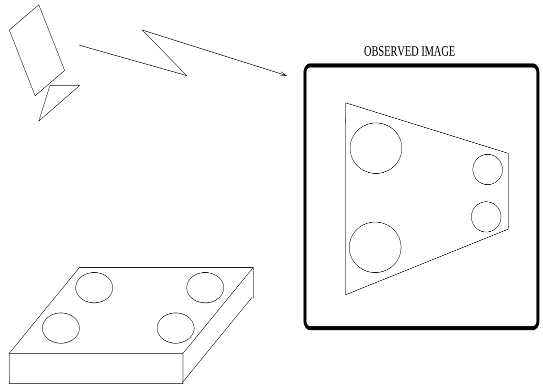


Figure 3: An Arbitrarily View and An Observed Image.

3. Methodology of Deterministic Approach

Let's consider a set of 3D line segments (a 3D line segment is equivalent to a pair of two 3D points on a rigid object) observed by one mobile camera moving at View One and View Two, respectively. Then, the problem of Head-Eye coordination becomes how to determine View Three (ie., the reference view) if the motion transformation between View One and View Two is known. Fig.4 shows the situation. If we are able to determine View Three and move the camera to View Three, the observed image at View Three will be equal to the reference image.

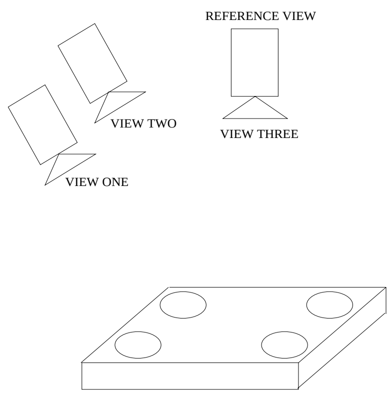


Figure 4: Principle for deterministic Head-Eye coordination: Determine a third unknown view (REFERENCE VIEW) from two known views (VIEW ONE, VIEW TWO).

Suppose that the motion between View One and View Two is known. This condition is realistic because one can move the camera from a current position (View One) to an intermediate position (View Two) with a known motion of the mobile head. If using calibrated stereovision, we can directly consider left camera as View One and right camera as View Two. In order to determine (unknown) View Three (ie., the reference view), the problem is how to determine the motion transformation between View Two (or View One) and View Three by using a set of line segments extracted from a set of three images: the image taken at View One, the image taken at View Two and the reference image (the reference image is memoried by the vision system at the initial stage of Head-Eye coordination). To solve this problem, the eight-points algorithm developed in [5] is not applicable because it did not allow to completely determine a motion transformation, nor the approaches based on three-dimensional correspondences [1] because a reference image does not provide any three-dimensional information. Then, one may ask the question of whether a solution exists ? In the next section, we present a novel and closed-form solution to affirmatively answer the question.

4. Determination of Third Unknown View From Two Known Views

Here, we present a general solution for Head-Eye coordination system composed of one camera mounted on a mobile head having six degrees of freedom. If one uses two cameras and a mobile head having less than six degrees of freedom, the solution is still valid and will obviously become simpler (because there will be less motion parameters to be estimated).

We model a camera by a perspective projection. Let $OXYZ$ be a 3D coordinate system associated to a camera and oxy be a 2D coordinate system associated to its image plane. If (X, Y, Z) represent the coordinates of a 3D point and (x, y) represent the coordinates of its 2D image point, under perspective projection, we have the following relationship:

$$x = X/Z \quad \text{and} \quad y = Y/Z \tag{1}$$

Under View Three, the projection of a 3D line segment onto the 2D image plane yields a 2D line segment which can be described by the equation of its supporting line like below:

$$a_{v3}\, x_{v3} + b_{v3}\, y_{v3} + c_{v3} = 0. \tag{2}$$

where the suffix $v3$ simply means the observation by the camera at View Three. By applying Eq.1 to the above equation, we obtain the following equation describing the projection plane passing through the 3D line segment:

$$a_{v3}\, X_{v3} + b_{v3}\, Y_{v3} + c_{v3}\, Z_{v3} = 0. \tag{3}$$

If we denote $S_{v3} = (a_{v3}, b_{v3}, c_{v3})$ and $P_{v3} = (X_{v3}, Y_{v3}, Z_{v3})$, the above equation can be written in a matrix form as follows:

$$S_{v3} \bullet P^t_{v3} = 0. \tag{4}$$

S_{v3} is the parameter vector of a 2D line segment extracted from the reference image (which is known), and P_{v3} are the coordinates of a point on a 3D line segment observed by the camera at View Three (which is unknown). Let P_{v2} be the coordinates of the same point observed by the camera at View Two. Then, the following relation holds:

$$P^t_{v3} = R_{v2v3} \bullet P^t_{v2} + T_{v2v3}. \tag{5}$$

where $(R_{v2v3}|T_{v2v3})$ is the unknown motion transformation between View Two and View Three.

The application of (5) to (4) yields:

$$S_{v3} \bullet R_{v2v3} \bullet P^t_{v2} + S_{v3} \bullet T_{v2v3} = 0. \tag{6}$$

Eq.6 is a linear system that relates to each other the 2D information S_{v3} (extracted from the reference image), the 3D information P_{v2} (can be calculated from View One and View Two) and the unknown motion transformation $(R_{v2v3}|T_{v2v3})$. Eq.6 is the first key equation of our method.

Note that P_{v2} is any point on a 3D line segment that can be thoroughly determined from two known views: View One and View Two (see [7] for a detailed solution). If we take a pair of two points (P'_{v2}, P''_{v2}) on a 3D line segment observed by the camera at View Two, we can build the following system:

$$\begin{cases} S_{v3} \bullet R_{v2v3} \bullet P'^t_{v2} + S_{v3} \bullet T_{v2v3} & = 0. \\ S_{v3} \bullet R_{v2v3} \bullet P''^t_{v2} + S_{v3} \bullet T_{v2v3} & = 0. \end{cases} \tag{7}$$

The subtraction of the two equations in Eq.7 yields:

$$S_{v3} \bullet R_{v2v3} \bullet (P'^t_{v2} - P''^t_{v2}) = 0. \tag{8}$$

If we define:

$$D_{v2} = (P'_{v2} - P''_{v2}) / \|P'_{v2} - P''_{v2}\|$$

which is an unit vector that represents the direction vector of a 3D line segment, Eq,8 becomes:

$$S_{v3} \bullet R_{v2v3} \bullet D_{v2}^{t} = 0. \tag{9}$$

The direction vector of a 3D line segment can be directly determined from the two known views: View One and View Two (see [7]). It is interesting to point out that Eq.9 does not involve the unknown T_{v2v3}. Therefore, we can first estimate the rotation matrix and then the translation vector of the motion transformation between View Two and View Three. Eq.9 is the second key equation of our method.

There are three independent unknown parameters in R_{v2v3} and three unknown parameters in T_{v2v3}. Hence, a set of three line segments are enough to fully determine R_{v2v3} and T_{v2v3} with Eq.9 and Eq.6. In practice, one may prefer to work with linear system and use Least-Squares estimation method. In order to do so, we can consider that there are eight unknown parameters in R_{v2v3} (factorize out, for instance, the parameter of row 3 and column 3 from R_{v2v3}, and recover it later on by using the orthogonality constraint that the matrix R_{v2v3} has to meet). In this way, a set of eight line segments allow to establish a linear system for the estimation of the matrix R_{v2v3}. Once we know R_{v2v3}, T_{v2v3} can be estimated from a linear system with a set of three line segments only.

So far, we can summarize our Head-Eye coordination algorithm (called "two-step" control algorithm) as follows:

- Step 0: At View One, capture one image and estimate a set (Set A) of line segments from the image.
- Step 1: Move the camera from View One to View Two with a known motion, capture one image and estimate a set (Set B) of line segments from the image.
- Step 2: Establish the correspondence between the above two sets (Set A, Set B) of line segments and calculate the corresponding set of 3D line segments. Establish the correspondence between Set B and the set (Set C) of line segments extracted from the reference image. Choose a set of eight pairs of matched line segments (between Set B and Set C) and estimate the motion transformation R_{v2v3} and T_{v2v3}. Finally, move the camera to View Three (ie., the reference view).

"Two-step" control algorithm is applicable to the configuration of one camera mounted on a mobile platform (eg., robot arm) (as in Fig.1b). It is easy to notice that we can have a "one-setp" control algorithm if using a calibrated stereovision system mounted on a mobile platform (as in Fig.1a). In this case, left camera is treated as View One and right camera is treated as View Two.

5. Experimental Results

The implementation of our method is straightforward if we already have algorithms for performing edge detection, line segment estimation, temporal/spatial matching and 3D reconstruction of line segments (see [3] for details). Due to space limitation, we give one example of experimental results:

This example shows an experiment with real system. We have a stereo cameras mounted on an indoor mobile robot (as in Fig.1a). A box is placed in front of the mobile robot. The goal here is to maintain a fixed view of the box by the mobile robot. At one time instant, the vision system captures a pair of stereo images. We move the box with a motion involving both rotation and translation. Moving the box is equivalent to move the reference view away from robot's current position. Then, the robot automatically computes the motion which is necessary to bring the robot again to the reference view position. Fig.5 shows the experimental results. The image size is 512×512. The estimated motion transformation between View Two and View Three (ie., reference view) is:

1. The rotation matrix R^*:

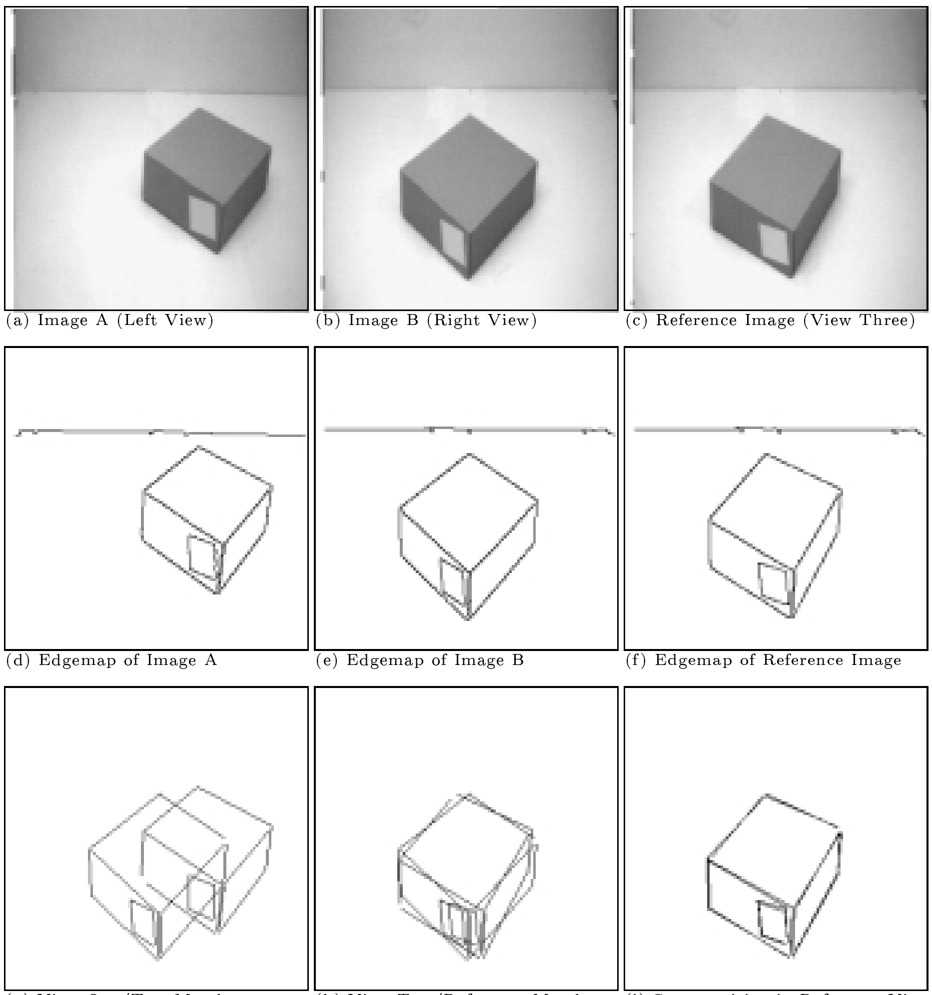
$$R^* = \begin{pmatrix} 0.980591 & 0.107180 & -0.164174 \\ -0.104814 & 0.994225 & 0.023033 \\ 0.165694 & -0.005378 & 0.986162 \end{pmatrix}. \tag{10}$$

   with its axis of rotation to be $(-0.072265, -0.839053, -0.539229)$ and the angle of rotation to be 11.336561 (degree).

2. The translation vector T^*:

$$T^* = (22.871370, -3.478399, 1.009764)(cm). \tag{11}$$

(The motion transformation is expressed in the left camera's coordinate system)



(a) Image A (Left View) (b) Image B (Right View) (c) Reference Image (View Three)
(d) Edgemap of Image A (e) Edgemap of Image B (f) Edgemap of Reference Image
(g) View One/Two Match (h) View Two/Reference Match (i) Superposition in Reference View

Figure 5: Head-Eye Coordination With Stereovision. The left camera is treated as View One and the right camera is treated as View Two. After taking a reference image, the box in the scene is moved away. Then, the Head-Eye coordination system computes the reference view from stereo images. (a) Image A taken at View One; (b) Image B taken at View Two; (c) The memorized reference image; (d) The edgemap of image A; (e) The edgemap of image B; (f) The edgemap of reference image; (g) The matched line segments extracted from image A and image B; (h) The matched line segments extracted from image B and reference image; (i) the superposition between the line segments extracted from the image taken at the estimated reference view and the line segments extracted from the original reference image.

6. Conclusions

The important conclusion that can be derived from this work is the following principle: *Given a set of eight (or three) non-collinear line segments, we can determine a third unknown view from two known views by using a linear estimation process (or non-linear estimation process if using only three non-collinear line segments).* This principle is directly applicable to solve the Head-Eye coordination problem, and thus ensures a closed-form solution. The "two-step" or "one-step" control algorithm developed under the presented principle complements existing iterative approaches of Head-Eye coordination. This in return deepens our understanding of Head-Eye coordination which is an intelligent human behavior and a relevant topic in Robotics.

References

[1] Blostein, S. D. and Huang, T. S. (1988). Algorithms for Motion Estimation Based on Three-Dimensional Correspondences. *Motion Understanding: Robot and Human Vision*, Editors: Martin, W. N. and Aggarwal, J. K., 328-350.

[2] Corke, P. I. (1994). Visual Control of Robot Manipulators: A Review. *Visual Servoing*, Editor: Hashimoto, K., World Scientific, 1-31.

[3] Faugeras, O. (1993). Three Dimensional Vision: A Geometrical Viewpoint. MIT Press.

[4] Hager, G. D, Chang, W. C. and Morse, A. S. (1995). Robot Hand-Eye Coordination Based on Stereo Vision. *IEEE Control Systems Vol.15*, 30-39.

[5] Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, Vol. 293.

[6] Rives, P. Chaumette, F and Espiau, B. (1991). Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing. *Proc. IEEE Int. Conf. Robotics and Automation*, 2248-2253.

[7] Xie, M and Thonnat, M. (1992). A Theory of 3D Reconstruction of Heterogeneous Edge Primitives from Two Perspective Views. *In Proceedings of the European Conference on Computer Vision*, Italy, pp 715-719.

[8] Xie, M, Trassoudaine, L, Alizon, J and Gallice, J. (1994). Road Obstacle Detection and Tracking By an Active and Intelligent Sensing Strategy. *Machine Vision and Applications*, Vol.7, pp165-177.

[9] Xie, M. (1995). Trinocular Vision For AGV Guidance: Path Localisation and Obstacle Detection. *Computers & Electrical Engineering*, Vol.21, No.6, pp441-452.

[10] Xie, M. (1995). A Single Camera Based Object Tracking System: Motion Stereo or Visual Servoing. *International Symposium on Industrial Robots*, Singapore, pp399-404.