The 23rd Conference of the Mechanical Engineering Network of Thailand
November 4 – 7, 2009, Chiang Mai

A Multi-Camera System on PC-Cluster for Real-time 3-D Tracking

Viboon Sangveraphunsiri*, Kritsana Uttamang, and Pongsakon Pedpunsri

Robotics and Advanced Manufacturing Lab, Department of Mechanical Engineering,
Chulalongkorn University
254 Phyathai, Patumwan Bangkok, 10330
Tel. +66 (0) 2218 6448, +66 (0) 2218 6585, Fax. +66 (0) 2218 6583
*E-mail: viboon.s@eng.chula.ac.th

Abstract

In this research, a real-time multi-camera system has been developed for tracking a moving object in 3-D space. Because of a large amount of data obtained from multiple cameras, a PC-cluster system, a group of linked computers, has been implemented to reduce computational load in one computer and make real-time tracking possible. The system consists of multiple cameras and a PC-cluster, four cameras and two PCs in this research. CCD camera is connected to computer via IEEE-1394 interface. Synchronization mechanism has been developed using PCI-to-PCI data movers with fiber optic connection. After camera calibration process, a 3-D position of moving object is calculated from a set of corresponding 2-D image coordinates of detected object feature using linear triangulation method. The system has been tested by tracking a spherical object with motion generated by Mitsubishi PA10 robotic arm. The experiment results show that the system can track the object in real-time with acceptable accuracy both position and velocity. As a prototypical application in reverse engineering, the developed system has been shown to be a hybrid vision/touch-probe measuring machine when moving the target object on surface needed to be 3-D reconstructed.

***Keywords**:* multiple cameras system, PC-cluster, vision-based tracking, 3-D pose estimation

1. Introduction

Multi-camera system has become an active research topic in computer vision because of its capability to deal with occlusion. As the number of camera increased, a large amount of data needs to be processed. This affects real-time performance of such system. One approach to overcome this problem is using multiple connected computers, called PC-cluster, to reduce computational load in one computer such as [1-3]. Another approach in [4] uses hardware embedded in each camera as local processing unit to process data before sending result to main processor.

In this paper, a multi-camera system based on PC-cluster has been developed. The connection between PCs has been made using PCI-to-PCI data mover card via fiber optic cable. To develop as a tracking system, five processing modules have been designed and arranged in each PC according to system architecture. The system can track an object, a spherical object in this research, in 3-D space in real-time. The detail will be described in the following sections.

2. Overview

2.1 3-D tracking process

To perform 3-D tracking, many processes have been involved such as camera calibration, feature extraction, and 3-D pose estimation. The flow of the processes is as follows:

1) Calibrate cameras.
2) Capture image from each camera synchronously.
3) Extract object feature in each image.
4) Estimate 3-D pose of object using extracted feature coordinates from step 3.
5) Display result.
6) Repeat step 2-5 for each frame.

2.2. Processing modules

The tracking process described above has been designed as five processing modules as follows:

- Camera calibration module
- Image capturing module
- 2-D image processing module
- 3-D pose estimation module
- PC to PC communication module

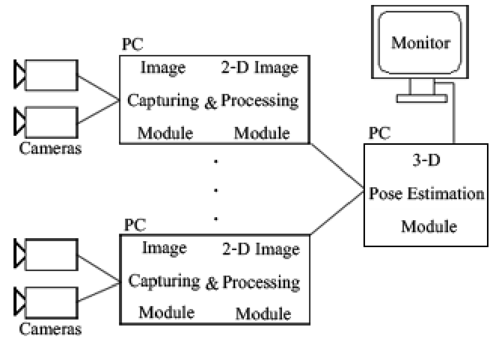

Fig. 1 The arrangement of the processing modules on PC-cluster

The arrangement of the processing modules in 3-D tracking process is shown in Fig.1. The PC to PC communication module embedded in each PC for transferring data between computers has not been shown in the figure. Before using the tracking system, all cameras need to be calibrated with camera calibration module which located in the main PC when implementing the system.

3. System Implementation

The PC-cluster system in this research has been implemented with two computers, one is local or main computer and another is remote computer. Local computer controls overall processes both contained in it and in remote computer.

3.1 Hardware

The two PCs of the system consist of the main PC which has Pentium IV 3.2 GHz as processor and 2GB of RAM installed and the remote PC which uses Pentium IV 2.0 GHz with 512 MB of RAM. They are connected to each other via fiber optic cable with dataBLIZZARD PCI-to-PCI data mover card. Each PC has 2 CCD cameras, PixeLINK PL-A741-BL, with 16mm lens connected through IEEE-1394 ports. The system has been shown in Fig. 2.

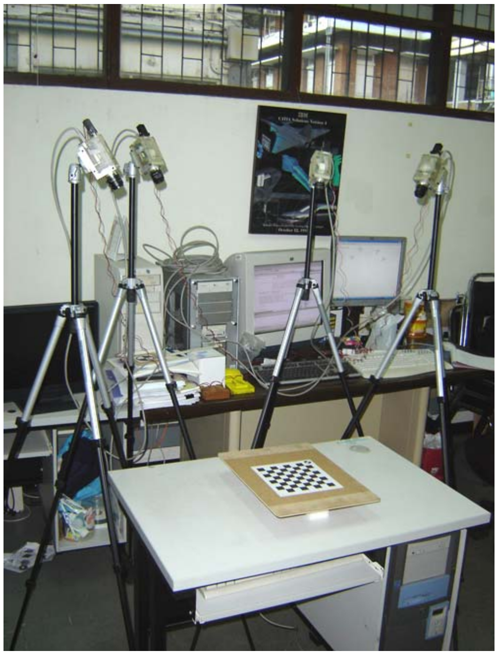

Fig. 2 The developed system with four cameras and two computers

3.2 Software

The architecture of cluster system consists of two types of application. Control application operates on local computer and service application operates on remote computer. Its schematic diagram is shown in Fig.3.

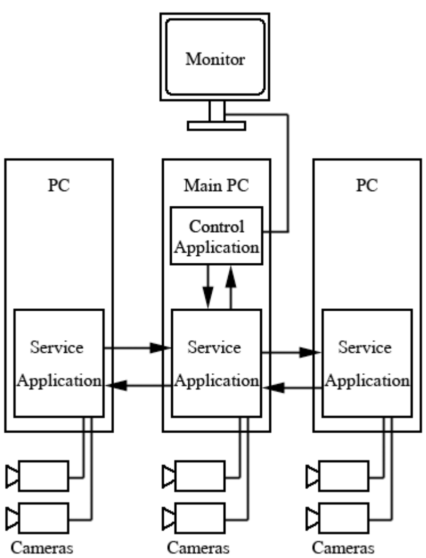

Fig. 3 The software architecture of PC-cluster

3.2.1 Service application

Service application is a Windows-based application which runs in the background of Windows operating system. In each PC, a service application will be responsible for image capturing and 2-D image processing, so an image capturing module and a 2-D image processing module are contained in it. Moreover, the service application has been developed to perform communication between PCs in PC-cluster system via PCI-to-PCI data mover card. It also provides interfaces to other Windows application.

3.2.2 Control application

For the developed system, one PC has been used as main PC. It controls remote PCs for tracking the target object. In the main PC, control application has been developed on the top of service application described earlier by using Component Object Model (COM) architecture. The control application receives 2-D image coordinates of the target from service application and estimates 3-D position of the target in the world coordinates. So, the 3-D pose estimation module is contained in this application. Moreover, the control application also provides application user interfaces. User can control tracking system through this application.

The architecture of the PC-cluster with processing modules described in section 2.2 is shown in Fig.4. The PC to PC communication module is implemented using message passing. It wraps data into programming data structure called message and pushes it to message queue to waiting for processing. The local computer contains the camera calibration module and the 3-D pose estimation module in addition to the three modules for remote computer.

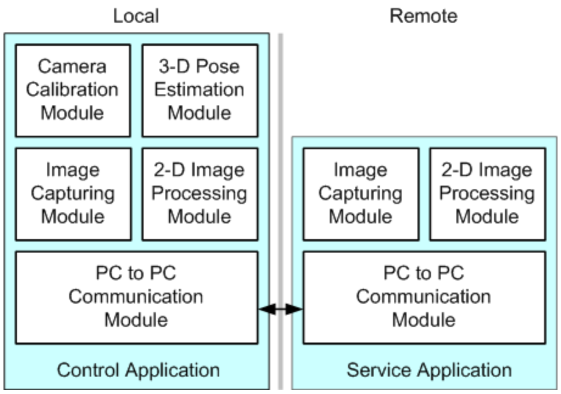

Fig. 4 The architecture of PC-cluster with processing modules

The application software has been implemented using Microsoft Visual Studio 2005 with C++ language on Windows XP SP2 operating system. The control application can display tracking results in real-time on graphical screen. OpenGL is used as graphics library. The control application is shown in Fig.5.

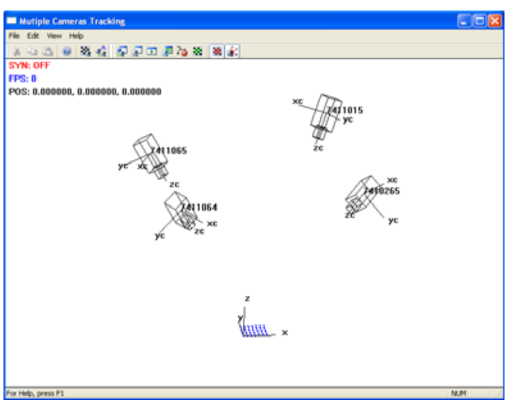

Fig. 5 The software implementation with graphical user interface

4. System Operations

4.1 Camera calibration

Camera calibration is the process of estimating of intrinsic and extrinsic parameters of camera model which is a matrix mapping between world and image coordinates [5, 6]. These parameters are important for 3-D reconstruction. There are many methods for calibrating camera with its pros and cons. In this research, two methods have been implemented, Tsai's [7] and Zhang's [8] method. For Zhang's method, radial distortion model has been modified to reduce computational time in reconstruction process, see [9] for more details. Chessboard calibration pattern used for calibration process has been design as 8x8 rectangles with 20 mm in width and height as shown in Fig. 6. The 49-point correspondences have been used in order to compute camera matrix.

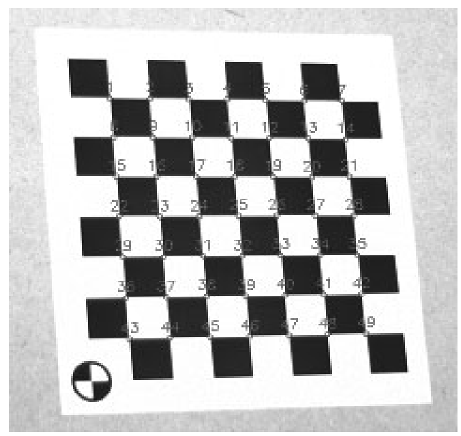

Fig. 6 The chessboard pattern for calibration

Circular mark in the calibration pattern is used for automatic identification of pattern orientation in the captured image.

4.2 Feature extraction

The accurate detection of image features is required in applications of 3-D reconstruction. In this paper, the spherical object whose image always circle has been used as tracking target. The circular Hough transform has been employed to detect center point and radius of the object.

4.3 Pose estimation

Consider the multiple cameras system which has n cameras, given $\mathbf{P}_i$ is the i-th camera matrix, and $\mathbf{x}_i$ is point in the i-th image of the 3-D world point $\mathbf{X}$ and corresponding to camera matrix $\mathbf{P_i}$, then we have $\mathbf{x_i} = \mathbf{P_i X}$. By using relation of vector cross product:

$$\mathbf{x_i} \times (\mathbf{P_i X}) = 0 \qquad (1)$$

Then

$$\begin{aligned} x_i\left(\mathbf{p}_i^{3T}\mathbf{X}\right)-\left(\mathbf{p}_i^{1T}\mathbf{X}\right)&=0\\ y_i\left(\mathbf{p}_i^{3T}\mathbf{X}\right)-\left(\mathbf{p}_i^{2T}\mathbf{X}\right)&=0\\ x_i\left(\mathbf{p}_i^{2T}\mathbf{X}\right)-y_i\left(\mathbf{p}_i^{1T}\mathbf{X}\right)&=0 \end{aligned} \qquad (2)$$

where

$\mathbf{p}_i^{jT}$ is the j-th row of camera matrix $\mathbf{P_i}$

From each camera, only 2 equations of (2) are independent, let choose the first two equations from each camera. A matrix $\mathbf{L}$ can be obtained by stacking up equation (2) as in [5, 10]:

$$\mathbf{LX} = \begin{bmatrix} x_1\mathbf{p}_1^{3T} - \mathbf{p}_1^{1T} \\ y_1\mathbf{p}_1^{3T} - \mathbf{p}_1^{2T} \\ x_2\mathbf{p}_2^{3T} - \mathbf{p}_2^{1T} \\ y_2\mathbf{p}_2^{3T} - \mathbf{p}_2^{2T} \\ \vdots \\ x_n\mathbf{p}_n^{3T} - \mathbf{p}_n^{1T} \\ y_n\mathbf{p}_n^{3T} - \mathbf{p}_n^{2T} \end{bmatrix} \mathbf{X} = \mathbf{0} \qquad (3)$$

Because of noise embedded in the captured data, we cannot obtain the exact solution of the above equation. The solution for $\mathbf{X}$ can be obtained by least-square solution (LS) of equation (3). LS gives an accurate result but it has no geometrical meaning. The other method is the bundle adjustment with Levenberg-Marquardt optimization (LM). This method tries to minimize geometric image distance between measured image point and reprojected image point of the estimated 3-D point. LM gives a better accuracy solution but slower. This is because LM is an iteration-based method. In contrast, LS is faster than LM but the error is also more than LM. Both methods have been implemented in this research.

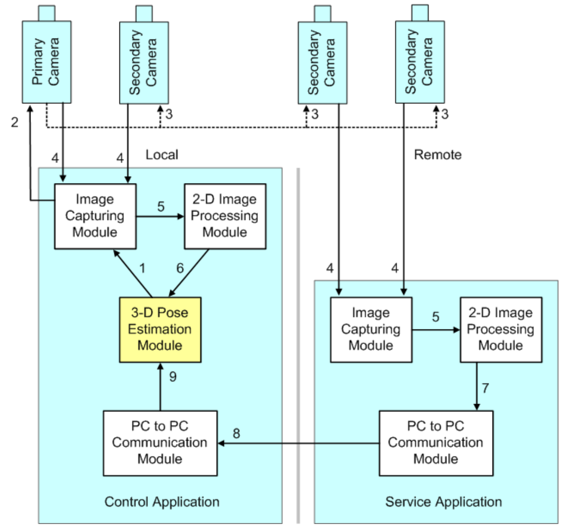

Fig. 7 3-D pose estimation process

Before performing 3-D pose estimation, all cameras have to be calibrated and image capturing module need to be set for synchronous image capturing. One camera has been chosen as primary camera to send hardware trigger to other cameras, secondary cameras. In 3-D pose estimation process, the 3-D pose estimation module in control application performs operations as shown in Fig.7.

1) 3-D pose estimation module sends software type trigger signal to local image capturing module.
2) Local image capturing module sends software trigger to primary camera only.
3) Primary camera sends hardware trigger signal to other cameras.
4) Both primary and secondary cameras capture images and send them back to image capturing module which attaches to them.
5) Image capturing module passes the captured images to 2-D image processing module to find 2-D image position of object.

6) In local computer, 2-D image processing module sends 2-D image positions to 3-D pose estimation module.
7) In remote computer, 2-D image processing module sends 2-D image positions to PC to PC communication module.
8) Remote PC to PC communication module transforms 2-D image positions into transferring message and transfers it to local PC to PC communication module.
9) Local PC to PC communication module receives transferring message and transform it back to 2-D image positions. Then 2-D image positions are sent to 3-D pose estimation module.

5. Experiments

5.1 3-D tracking

The developed system can detect the moving target, which is a spherical ball, attached to Mitsubishi PA10 robot arm as shown in Fig. 8. The motion for the robot arm has been generated and the system tracks the moving target. The results have been shown in Fig 9-14.

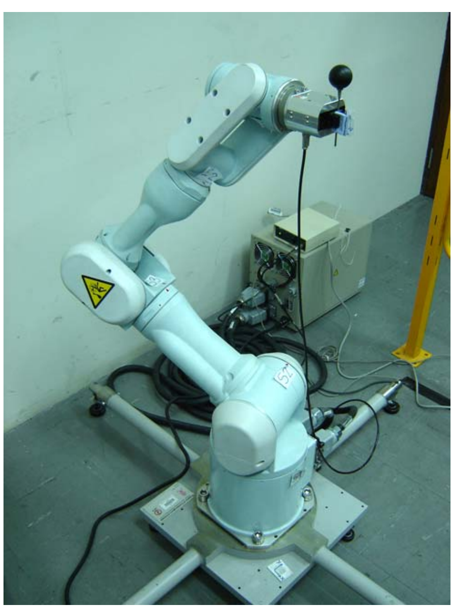

Fig. 8 The robot arm holds a spherical target

Fig. 9, 10, and 11 show the comparison of the result from the measurements and the robot arm move in XY-plane, YZ-plane, and ZX-plane, respectively. The speed of the robot arm is set to 0.05 rad/sec. Fig 11, 12, and 13 show the system detecting the target moving helically in Z-direction, Y-direction, and X-direction, respectively with the same velocity.

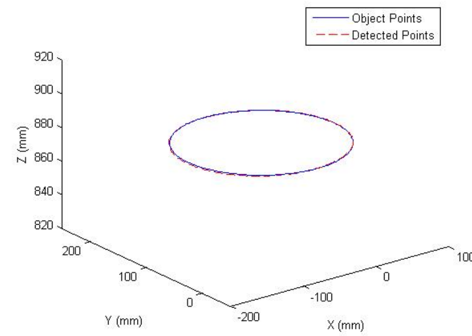

Fig. 9 The target moves in the XY-plane

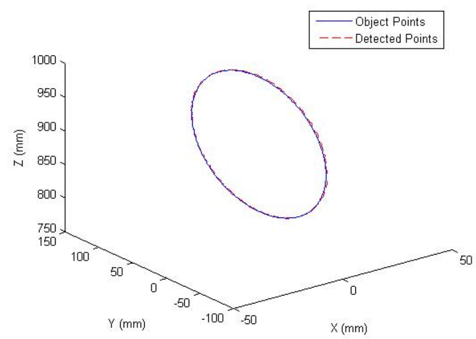

Fig. 10 The target moves in the YZ-plane

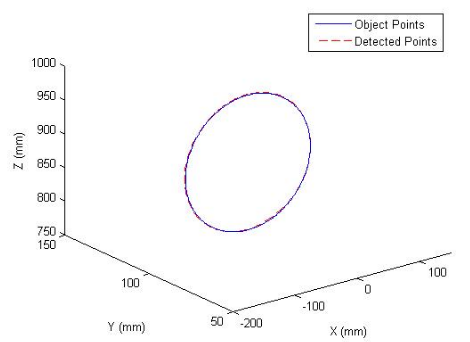

Fig. 11 The target moves in the ZX-plane

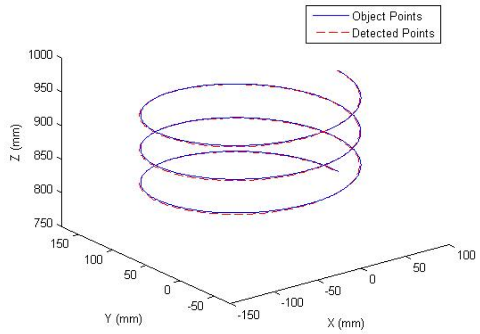

Fig. 12 Helical motion in z-direction

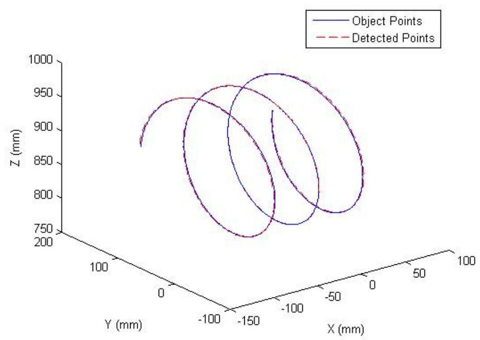

Fig. 13 Helical motion in X-direction

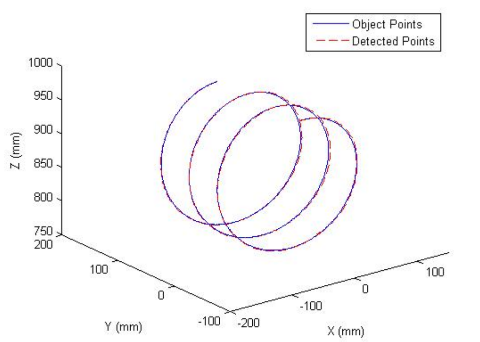

Fig. 14 Helical motion in Y-direction

The target object motion with s-curve velocity profile has been tested as shown in Fig. 15. The robot arm is programmed to move in y-direction with acceleration and deceleration set equal to 20 mm/s^2. The total distant is 300 mm with 40 mm/s constant velocity. The result shows that the system can track the target very well. We can improve the result by using faster cameras and tracking in a well-defined environment to reduce exposure time of the camera.

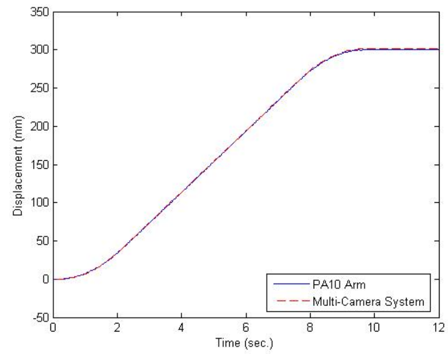

Fig. 15 Linear motion with s-curve velocity profile

5.2 Application for reverse engineering

The developed system can be used as a 3-D coordinate measuring machine for reverse engineering application. In Fig. 16, the spherical target moves on part that surface need to be reconstructed like a probe. The triangular mesh created from the measurement data has been shown in Fig. 17.

Fig. 16 The spherical target moving on surface

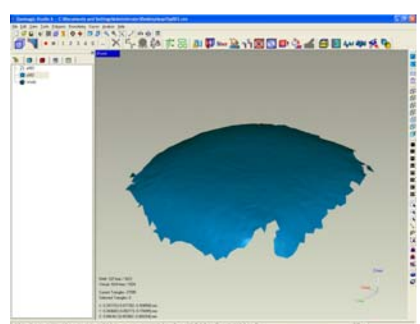

Fig. 17 The triangular mesh created from the measurement points

6. Conclusion

In this paper, a real-time 3-D tracking system using multiple cameras has been implemented on PC-cluster. Two types of application, control and service application, with five processing modules have been designed and implemented to each PC. The system can track a spherical object moved by Mitsubishi PA-10 manipulator with acceptable accuracy. The better result can be obtained if using better cameras. To apply for reverse engineering task, the system can track the spherical object moving on target surface. With the touch probe improvement, the system can be a vision/touch-probe measuring machine with flexibility and scalability.

7. References

[1] Arita, D., Yonemoto, S., and Taniguchi, R. (2000). Real-Time Computer Vision on PC-Cluster and Its Application to Real-Time Motion Capture. *5th IEEE International Workshop on Computer Architectures for Machine Perception*, pp. 205-214.

[2] Yoshimoto, H., Date, N., and Yonemoto, S. (2003). Vision-Based Real-Time Motion Capture System using Multiple Cameras. *Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 30 July-1 August 2003, pp. 247-251.

[3] Kurihara, K., Hoshino, S., Yamane, K. and Nakamura, Y. (2002). Optical Motion Capture System with Pan-Tilt Camera Tracking and Realtime Data Processing. *Proceedings of the IEEE International Conference on Robotics and Automation*, 2: pp.1241-1248.

[4] Garcia, R., Batlle, J. and Salvi, J. (2002). A New Approach to Pose Detection using a Trinocular Stereovision System, *Real-Time Imaging*, pp.73-93.

[5] Hartley, R., and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*, 2nd edition, Cambridge University Press, Cambridge, UK.

[6] Trucco, E. and Verri, A. (1998). *Introductory Techniques for 3-D Computer Vision*, Prentice-Hall, New Jersey, USA.

[7] Tsai, R. (1987). A Versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology using Off-The-Shelf TV Cameras and Lenses, *IEEE Journal of Robotics and Automation*, vol. 3. no.4, pp.323-344.

[8] Zhang, Z. (2000). A Flexible New Technique for Camera Calibration, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22. no. 11, pp.1330-1334.

[9] Uttamang, K., and Sangveraphunsiri, V. (2007). Multi-Camera System using PC-Cluster for Real-Time 3-D Pose Estimation, *Thammasat Int. J. Sc. Tech.* 12, April-June 2007, pp. 52-63.

[10] Faugeras, O. and Luong, Q. (2001). *The Geometry of Multiple Images*, MIT Press, Cambridge, MA.