We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

11

A Pseudo Stereo Vision Method using Asynchronous Multiple Cameras

Shoichi Shimizu, Hironobu Fujiyoshi, Yasunori Nagasaka
and Tomoichi Takahashi
Chubu University & Meijo University
Japan

1. Introduction

Systems of multiple baseline stereo vision (Okutomi & Kanade, 1993 and Kanade et al., 1997) have been proposed and used in various fields. They require cameras to synchronize with one another to track objects accurately to measure depth. However, inexpensive cameras such as Web-cams do not have synchronous systems.

A system of tracking human motion (Mori et al., 2001) and a method of recovering depth dynamically (Zhou & Tao, 2003) from unsynchronized video streams have been reported as approaches to measuring depth using asynchronous cameras. In the former, the system can obtain the 2D position of a contact point between a human and the floor, and the cycle of visual feedback is 5 fps on average. In the latter, the method creates a non-existing image, which is used for stereo triangulation. The non-existing image is created from the estimated time delay between unsynchronized cameras and optical flow fields computed in each view. This method can output a depth map at the moment of frame *t*-1 (one frame before the current one), not the current frame.

We propose a method of pseudo-stereo vision using asynchronous multiple cameras. Timing the shutters of cameras asynchronously has an advantage in that it can output more 3D positions than a synchronous camera system. The 3D position of an object is measured as a crossing point of lines in 3D space through the observation position on the last frame and the estimated 3D position using the previous two frames. This makes it possible for the vision system to consist of asynchronous multiple cameras. Since the 3D position is calculated at the shutter timing of each camera, the 3D position can be obtained from the number of cameras x 30 points.

This chapter is organized as follows. Section 2 describes a method of measuring the 3D position using asynchronous multiple cameras. Section 3 reports experimental result on the recovery of motion when an object is moved in virtual 3D space, and discusses the effectiveness of the proposed method. Section 4 describes some experimental setups and reports experimental results using real images of an object moving at high speed. Section 5 discusses the processing time and enhancement of n cameras. Finally, Section 6 summarizes the method of pseudo-stereo vision.

Source: Scene Reconstruction, Pose Estimation and Tracking, Book edited by: Rustam Stolkin,
ISBN 978-3-902613-06-6, pp.530, I-Tech, Vienna, Austria, June 2007

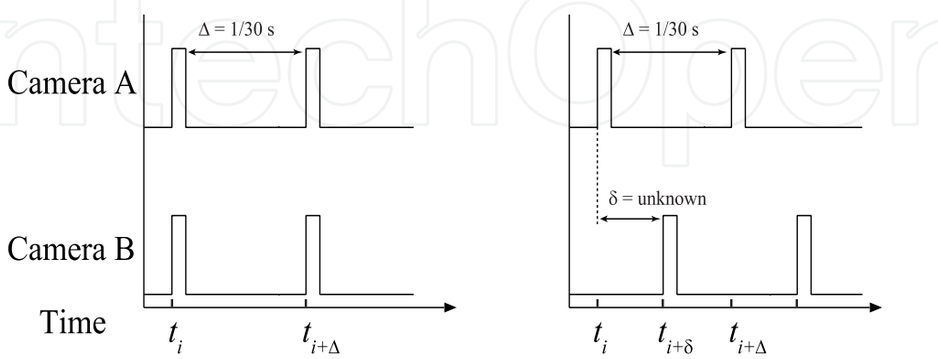


Figure 1. Two possible combinations of shutter timings

2. 3D position measurement with multiple cameras

The method of stereo vision, which measures the 3D position of an object, requires two images to be captured at the same time to reduce errors in measurement. We investigated a method of pseudo-stereo vision taking advantage of the time delay between the shutter timings of two asynchronous cameras to calculate the 3D positions of objects.

2.1 Shutter timings of two cameras

Two possible combinations of shutter timings by two cameras are outlined in Fig. 1. The first (a) is the same shutter timings, which is used in multiple baseline stereo vision, synchronized by a synchronous signal generator. In a conventional stereo vision system, the 3D positions can be obtained at a maximum of 30 fps using a normal camera with a fast vision algorithm (J. Bruce et al., 2000).

Figure 1 (b) outlines the other type of shutter timing using asynchronous cameras where there is a time delay of δ. When an object moves fast, the stereo vision with this shutter timing calculates the 3D position from corresponding points with the time delay. Therefore, the estimated 3D position, $\hat{\boldsymbol{P}}_{t+1}$, has error, as shown in Fig. 2. This chapter focuses on this kind of shutter timing, and proposes a method of calculating the 3D position taking time delay δ into account, which is unknown. Since our method calculates the 3D position at each shutter timing, it is possible to output the 3D position from the number of cameras x 30 points per second.

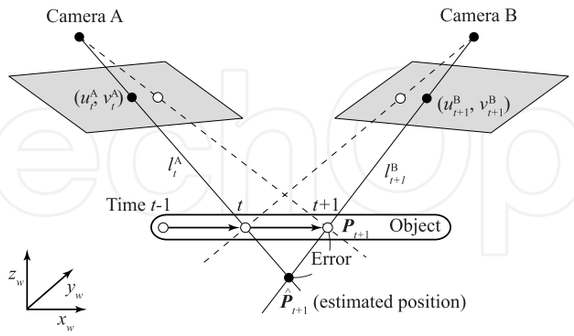


Figure 2. Error in stereo vision with time delay

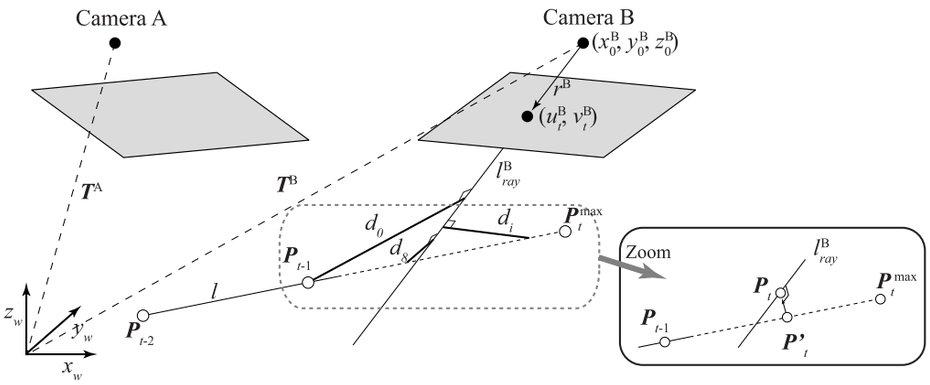


Figure 3. Estimate of 3D position of last frame

2.2 Estimate 3D position of last frame

The 3D position, $\boldsymbol{P}_t = [x_t, y_t, z_t]$, of the last frame, t, is estimated from positions $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$ in the previous two frames as shown in Fig. 3. Where the image in last frame t is captured by camera B, the algorithm to calculate the 3D position in last frame t is described as follows:

1. Given the 3D positions in the previous two frames, $\boldsymbol{P}_{t-1} = [x_{t-1}, y_{t-1}, z_{t-1}]$ and $\boldsymbol{P}_{t-2} = [x_{t-2}, y_{t-2}, z_{t-2}]$, straight line l is calculated by:

$$l = \frac{x - x_{t-1}}{x_{t-2} - x_{t-1}} = \frac{y - y_{t-1}}{y_{t-2} - y_{t-1}} = \frac{z - z_{t-1}}{z_{t-2} - z_{t-1}}. \quad (1)$$

2. The instant in time when the images are captured to estimate the 3D positions $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$, are unknown due to asynchronous cameras being used. The maximum number of frames for a normal camera is generally 30 fps. Since the maximum time delay is assumed to be 1/30 s, the range of locations in 3D position can be estimated by:

$$\begin{aligned}\boldsymbol{P}_t^{\max} &= \max_{(0<\delta<1/30)}\left(\frac{(\boldsymbol{P}_{t-1}-\boldsymbol{P}_{t-2})}{\delta}(\Delta-\delta)+\boldsymbol{P}_{t-1}\right)\\ &= \max_{(0<\delta<1/30)}\left(\frac{(\boldsymbol{P}_{t-1}-\boldsymbol{P}_{t-2})\Delta}{\delta}-(2\boldsymbol{P}_{t-1}-\boldsymbol{P}_{t-2})\right),\end{aligned} \tag{2}$$

where Δ is the frame rate (1/30 s).

3. Let $\boldsymbol{T}^{\mathrm{B}}=[T_x^{\mathrm{B}}\ T_y^{\mathrm{B}}\ T_z^{\mathrm{B}}]$ be the translation vector from the origin of the world coordinate to the focus point of camera B, and $\boldsymbol{r}^{\mathrm{B}}=[\lambda\ \mu\ \nu]^T$ be the vector that denotes the direction of the viewing ray, l_{ray}^{B}, passing through the position on image coordinate $(u_t^{\mathrm{B}}, v_t^{\mathrm{B}})$ and the focus point of the camera B. Then, viewing ray l_{ray}^{B} is defined by

$$l_{ray}^{\mathrm{B}}=\frac{x-T_x^{\mathrm{B}}}{\lambda}=\frac{y-T_y^{\mathrm{B}}}{\mu}=\frac{z-T_z^{\mathrm{B}}}{\nu}. \tag{3}$$

The distance, *d*, is calculated between points $\boldsymbol{P}_i=[x_i, y_i, z_i]$ on straight line *l* and viewing ray l_{ray}^{B} using:

$$d(x_i,y_i,z_i)=\sqrt{\left\{\mu(x_i-T_x^{\mathrm{B}})-\lambda(y_i-T_y^{\mathrm{B}})\right\}^2+\left\{\nu(y_i-T_y^{\mathrm{B}})-\mu(z_i-T_z^{\mathrm{B}})\right\}^2+\left\{\lambda(z_i-T_z^{\mathrm{B}})-\nu(x_i-T_z^{\mathrm{B}})\right\}^2}. \tag{4}$$

Then, the 3D position, $\boldsymbol{P}_i$, which produces the minimum distance, is selected as $\boldsymbol{P}'_t$ by calculating:

$$\boldsymbol{P}'_t=\underset{(\boldsymbol{P}_{t-1}\le\boldsymbol{P}_i\le\boldsymbol{P}_t^{max})}{\arg\min}\ [d(\boldsymbol{P}_i)]. \tag{5}$$

4. The 3D position may not exist on the viewing ray l_{ray}^{B}, as shown in Fig. 3, because of prediction error. To solve this problem, 3D position $\boldsymbol{P}_t$ is calculated as the nearest point on viewing ray l_{ray}^{B} by:

$$\boldsymbol{P}_t=\frac{(\boldsymbol{P}'_t-\boldsymbol{T}^{\mathrm{B}})\cdot\boldsymbol{r}^{\mathrm{B}}}{\left|\boldsymbol{r}^{\mathrm{B}}\right|^2}\boldsymbol{r}^{\mathrm{B}}+\boldsymbol{T}^{\mathrm{B}}. \tag{6}$$

If the last frame is camera A, 3D position $\boldsymbol{P}_t$ can be calculated by changing the suffix.

2.3 Calculation of 3D positions of previous two frames

It is necessary to calculate the previous 3D positions, $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$, accurately to obtain the current frame using the method described in section 2.2. Where frame *t*-1 is camera B, the position of an object using the image coordinates from camera A at frame *t* does not have a

corresponding point from camera B at the same time, which is needed to calculate the 3D position using stereo vision. The predicted point of camera B, $(u'^{B}_{t-1}, v'^{B}_{t-1})$, corresponding to observed point $(u^{A}_{t-1}, v^{A}_{t-1})$ is generated by a basic interpolation technique, as shown in Fig.4. The 3D position can be measured by stereo vision using observed point $(u^{A}_{t-1}, v^{A}_{t-1})$ and pseudo-corresponding point $(u'^{B}_{t-1}, v'^{B}_{t-1})$. The algorithm to calculate the 3D position at $t-1$ is described as follows:

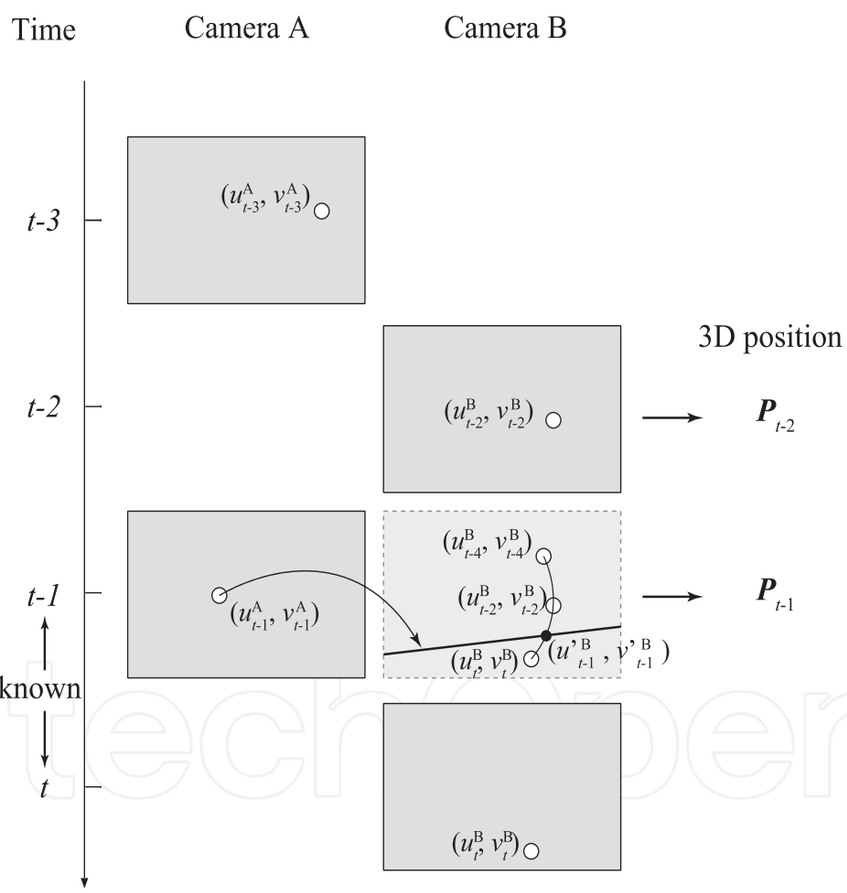


Figure 4. 3D position is estimated using spline curve

Estimation of 3D position using spline curve

The object's trajectory is estimated by spline-curve fitting (de Boor, 1978). The pseudo-corresponding point is obtained as the intersection between the trajectory and epipolar line, as shown in Fig. 4. The spline curve on the camera image can be calculated from three observed points, $(u_t^{\mathrm{B}}, v_t^{\mathrm{B}})$, $(u_{t-2}^{\mathrm{B}}, v_{t-2}^{\mathrm{B}})$, and $(u_{t-4}^{\mathrm{B}}, v_{t-4}^{\mathrm{B}})$ by:

$$u(s) = \sum_{i=0}^{N-1} \alpha_i B_{i,K}(s), \quad v(s) = \sum_{i=0}^{N-1} \beta_i B_{i,K}(s), \tag{7}$$

where s is a parameter uniquely determined by (u, v), $u = u(s)$ and $v = v(s)$ are determined by s, and $B_{i,K}$ means the $(K - 1)$ dimensional B spline. Here, parameters α and β are solved from some input point (u, v). The spline curve can be constructed using parameters α and β. Then, the epipolar line (Faugeras, 1993) on camera B is calculated from observation point $(u_{t-1}^{\mathrm{A}}, v_{t-1}^{\mathrm{A}})$ on camera A. Let $\boldsymbol{F}$ denote a fundamental matrix of 3x3 which is defined by:

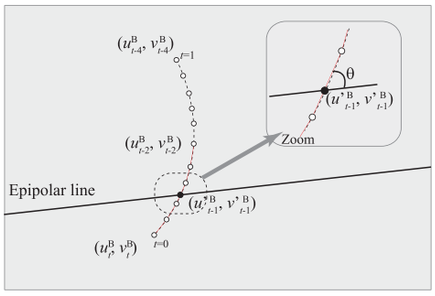


Figure 5. Calculate intersecting point of spline curve and epipolar line

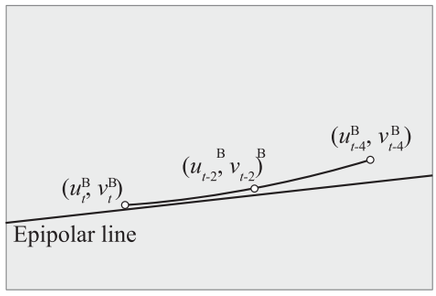


Figure 6. Error in the intersecting point

$$\boldsymbol{m}_1 \, \boldsymbol{F} \, \boldsymbol{m}_2^T = 0, \tag{8}$$

where $\boldsymbol{m}_1 = \begin{bmatrix} u^{\mathrm{A}}{}_{t-1} & v^{\mathrm{A}}{}_{t-1} & 1 \end{bmatrix}$, $\boldsymbol{m}_2 = \begin{bmatrix} u'^{\mathrm{B}}_{t-1} & v'^{\mathrm{B}}_{t-1} & 1 \end{bmatrix}$, $F = \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix}$. The epipolar line on camera B is given by:

$$v = -\frac{\boldsymbol{m}_1 \begin{bmatrix} f_1 & f_4 & f_7 \end{bmatrix}^T u + \boldsymbol{m}_1 \begin{bmatrix} f_3 & f_6 & f_9 \end{bmatrix}^T}{\boldsymbol{m}_1 \begin{bmatrix} f_2 & f_5 & f_8 \end{bmatrix}^T}. \tag{9}$$

The spline curve has various node points from $(u_t^{\mathrm{B}}, v_t^{\mathrm{B}})$ to $(u_{t-4}^{\mathrm{B}}, v_{t-4}^{\mathrm{B}})$, and the intersecting point is determined as pseudo-corresponding point $(u'^{\mathrm{B}}_{t-1}, v'^{\mathrm{B}}_{t-1})$ on camera B as shown in Fig. 5. Then, the 3D position $\boldsymbol{P}_{t-1}$ is calculated from pseudo-corresponding point $(u'^{\mathrm{B}}_{t-1}, v'^{\mathrm{B}}_{t-1})$ and observation point $(u^{\mathrm{A}}_{t-1}, v^{\mathrm{A}}_{t-1})$ on camera A using stereo vision.

3D position $\boldsymbol{P}_{t-2}$ at frame *t*-2 is calculated in the same way as above, i.e., as the intersecting point of the spline curve of *t*-1, *t*-3, *t*-5 frames, and the epipolar line from image coordinate $(u^{\mathrm{B}}_{t-2}, v^{\mathrm{B}}_{t-2})$ of camera B.

Reliability of estimated position

Angle θ of the intersecting position Fig. 5 is calculated to measure how reliable the intersecting position is. There is a case where the spline curve becomes parallel to the epipolar line, as shown in Fig. 6. For this reason, the object moves along the epipolar plane. Here, the estimated 3D position includes a large amount of error. Therefore, it is necessary to reject the outlier causing the error in the 3D position using angle θ.

3. Simulation experiments

3.1 Recovery of object motions

We evaluated the method we propose by simulating the recovery of object's movement with uniform and non-uniform motion in 3D space (3,000 x 2,000 x 2,000 mm). In the experiment, we assumed that n (n= 2, 3) cameras would be mounted at a height of 3,000 [mm]. The conventional approach and the proposed method were evaluated using two kinds of motion.

- Uniform motion (spiral) : An object moves in a spiral with a radius of 620 mm at a velocity of 3,000 mm/s at center (*x*, *y*) = (1,000, 1,000)

- Non-uniform motion : An object falls from a height of 2,000 mm, then describes a parabola (gravitational acceleration: g = 9.8 m/s^2)

The trajectory of the object is projected to the virtual image planes of each camera. A 3D position is estimated with the proposed method described in Section 2 using the point projected on the virtual image plane (*u*, *v*) of each camera.

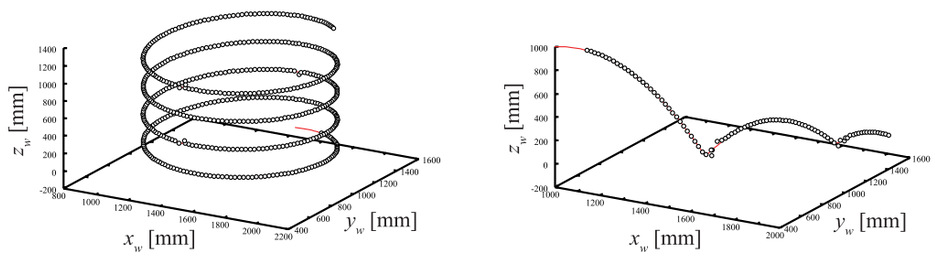
3.2 Simulation results

Table 1 lists the averages for estimation error in the simulation experiments, and Fig. 7 has examples of the recovery of spiral and non-uniform motion using three cameras. The "constant" in Table 1 means the time delay of the shutter timing is the same (δ = 1/60 s when there are two cameras, and 1/90 sec when there are three), and "random" in Table 1 means the time delays at every frame are not the same (0 < δ < 1/30 s). The "stereo (asynchronous)" shows the results for general stereo vision with time delay. It is clear that the proposed method provides more accurate estimates than stereo vision. This is because it can estimate the 3D position using the pseudo-corresponding points at the same time. Using three cameras provides more accurate results than using two cameras. This is why time delay δ is short, and the accuracy of a linear prediction is improved.

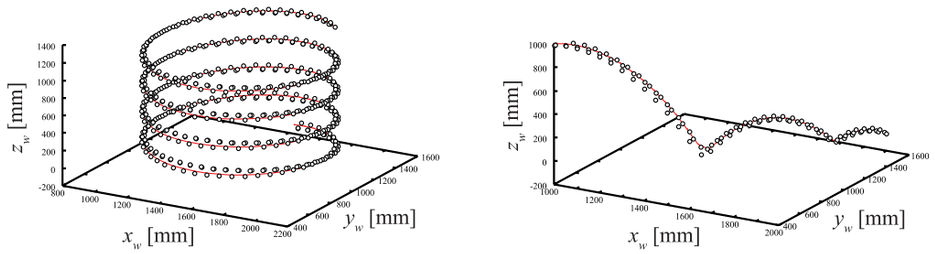
Motion	Camera	Proposed method	Stereo(asynchronous)

		Constant	Random	Constant	Random
Spiral	2	0.86	2.89	13.68	13.68
	3	0.32	2.17	12.54	13.72
Non-uniform	2	2.30	3.61	11.07	11.92
	3	1.23	2.51	9.87	12.54

Table 1. Average of absolute errors in 3D positions [mm]



(a) Proposed method (left: spiral, right: non-uniform motion)



(b) Stereo vision (left: spiral, right: non-uniform motion)

Figure 7. Example of recovery of 3D position

4 Experiments using real cameras

4.1 Configuration of vision system

Figure 8 shows how we placed the three cameras in our vision system. They were mounted at a height of 2,800 mm, and each viewed an area of 2,000 x 3,000 mm. Each was calibrated using corresponding points of world coordinates (x_w, y_w, z_w) and image coordinate (u, v) based on Tsai's camera model (Tsai, 1987). The shutter timing for all three cameras was controlled by a TV-signal generator. Three frame grabbers for the three cameras were installed on a PC. Our hardware specifications were:

<Hardware Specifications>
- PC (DELL PRECISION 530)
  - CPU (XEON DUAL PROCESSOR 2.2 GHz)
  - MEMORY (1.0 GB)
- CAMERA (SONY XC-003) x 3
- FRAME GRABBER (ViewCast Osprey-100) x 3
- TV SIGNAL GENERATOR (TG700 + BG7)

Process-1, process-2, and process-3 in Fig. 8 analyze images from camera A, camera B, and camera C every 1/30 s. Analyzed results such as object position in image coordinate (u_t, v_t) and the instant at which the analyzed image is captured are sent via the UDP interface to process-4, which outputs the 3D positions of the object using the procedure described in Section 2. There is negligible delay due to communications between processes because this work is done on the same computer.

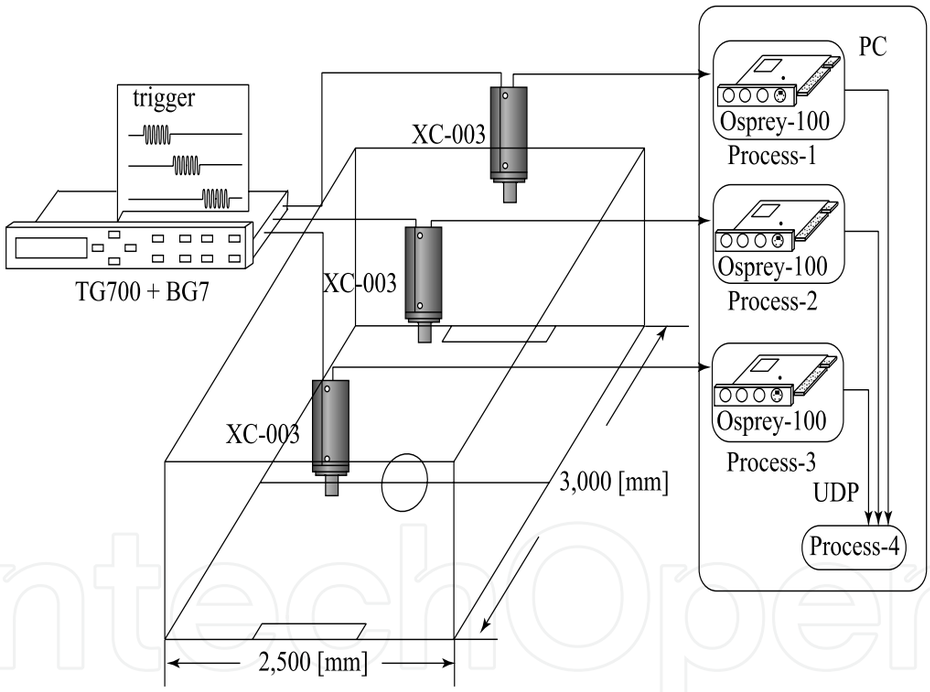


Figure 8. Overview of our vision system

4.2 Recovery of uniform circular motion

We used a turntable and a ball, as shown in Fig. 9, to evaluate the accuracy of the estimated 3D positions. A ball attached to the edge of a ruler (1,000 mm in length) makes a uniform

circular motion with a radius of 500 mm. The turntable is placed on a box at a height of 500 mm, and the ball is 660 mm above the floor. The turntable rotates at a speed of 45 rpm, and its rotation speed per second is (45x2)/60 = 0.478 radian.

Table 2 lists the averages for estimation error. The pixel resolution at the floor is 4 mm. The average of the positions using stereo vision with synchronous cameras was measured within 4 mm; this error is an appropriately result from the viewpoint of resolution. It is clear that our method is better than the stereo vision with asynchronous shutter timing. However, the error with our method is approximately 6 mm larger than with stereo vision with synchronous shutter timing. This is because the pseudo-corresponding points have the error. However, it is possible to reject 3D positions using angle θ, which was described in Section 2.3.

Proposed method	Stereo	
	Synchronous	Asynchronous
9.2	3.7	266.0

Table 2. Average and variance in estimation error [mm]

(a) Image from camera A

(b) Image from camera B

(c) Image from camera C

Figure 9. Captured images of turntable and ball

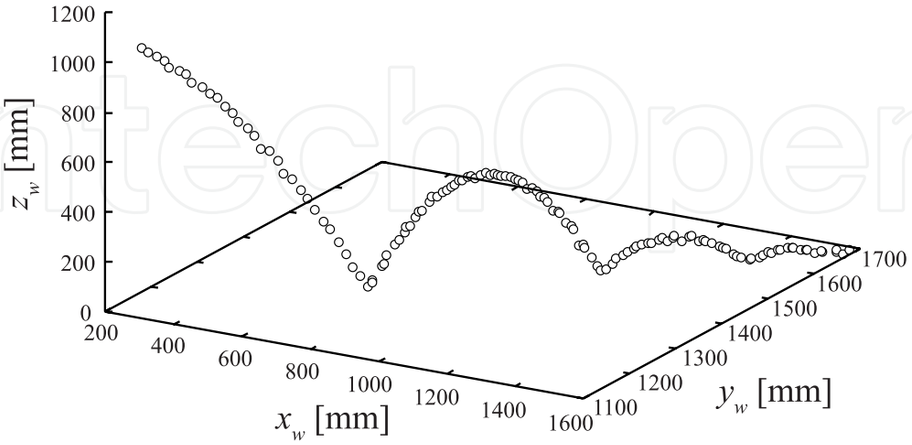


Figure 10. Recovery of bounding motion

4.3 Recovery of bounding ball

We performed an experiment to recover a bounding object to evaluate the proposed method since the recovery of uniform circular motion was at a constant speed. Figure 10 has an example of the motion of a hand-thrown ball bounced for about 1.5 s. We can see 135 plotted points. This indicates that the speed is the same as a 90 fps camera. Therefore, the proposed method can obtain the 3D position at 90 fps, when we use three normal cameras (30 fps) when the time delay of each shutter timing is 1/90 s.

5. Processing time

Our vision system was implemented on a PC with dual Xeon processors 2.2 GHz dual Xeon processors. It took 2.7 ms for the vision process to calculate the positions of colored objects from an image, and 1.8 ms for the transmission process via UDP with one PC in our implementation. Therefore, this system can determine the 3D positions of an object in 4.5 ms from the time the analyzed image was captured. It is possible to use n cameras with our system when the time delay from each of them is δ, if $\delta >$ n x (processing time) is satisfied.

6. Discussion and Conclusion

We proposed a method of pseudo-stereo vision method using cameras with different shutter timings, where the previous two frames were calculated using a spline curve. The method can output a 3D position at 90 fps using three cameras, and using multiple cameras is expected to enhance the output of the 3D positions.

We confirmed that our method was better than stereo vision with the time delay in simulation experiments. The error was 9.2 mm in experiments using real cameras. However, the error was within a useful range, because the object's radius was 20 mm. Moreover, it is clear that our method is better than the stereo vision with asynchronous shutter timing.

7. References

M. Okutomi.; & T. Kanade. (1993). A Multiple Baseline Stereo, IEEE Trans. PAMI, Vol.15, No. 4, pp. 353-363, 1993.

T. Kanade.; H. Kano.; S. Kimura.; E. Kawamura; A. Yoshida.; & K. Oda. (1997). Development of a Video-Rate Stereo Machine, Journal of the Robotics Society of Japan, Vol. 15, No.2, pp.261-267, 1997

H. Mori.; A. Utsumi.; J. Ohya. & M. Yachida. (2001). Human Motion Tracking Using Non-synchronous Multiple Observations. In the IEICE, Vol.J84-D-II, Num. 1, pp. 102-110, 2001.

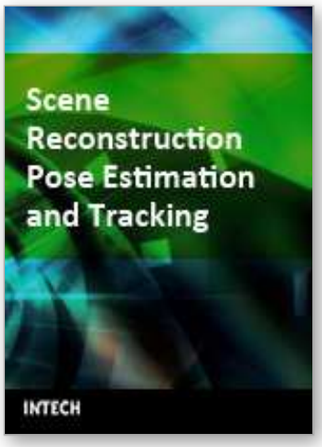
C. Zhou. & H. Tao. (2003). Dynamic Depth Recovery from Unsynchronized Video Streams. In Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 351-358, 2003.

J. Bruce.; T. Balch. & M. Veloso. (2000). Fast and Inexpensive Color Image Segmentation for Interactive Robots, IROS-2000, Vol. 3, pp. 2061-2066, 2000.

O.D. Faugeras. (1993). Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press, Cambridge, MA, 1993.

de. Boor. (1978). A Practical Guide to Splines. Splinger-Verlag, New York, 1978.

R. Y. Tsai. (1987). A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses, In IEEE Journal of Robotics and Automation, Vol.RA-3, Num.4, pp.323-344, 1987.



Scene Reconstruction Pose Estimation and Tracking

Edited by Rustam Stolkin

ISBN 978-3-902613-06-6

Hard cover, 530 pages

Publisher I-Tech Education and Publishing

Published online 01, June, 2007

Published in print edition June, 2007

This book reports recent advances in the use of pattern recognition techniques for computer and robot vision. The sciences of pattern recognition and computational vision have been inextricably intertwined since their early days, some four decades ago with the emergence of fast digital computing. All computer vision techniques could be regarded as a form of pattern recognition, in the broadest sense of the term. Conversely, if one looks through the contents of a typical international pattern recognition conference proceedings, it appears that the large majority (perhaps 70-80%) of all pattern recognition papers are concerned with the analysis of images. In particular, these sciences overlap in areas of low level vision such as segmentation, edge detection and other kinds of feature extraction and region identification, which are the focus of this book.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen