ROBOCUP2004 SYMPOSIUM,
Instituto Superior Técnico,
Lisboa, Portugal,
July 4-5, 2004.

A Method of Pseudo Stereo Vision from Images of Cameras Shutter Timing Adjusted

Hironobu Fujiyoshi[1], Shoichi Shimizu[1], Yasunori Nagasaka[2], and Tomoichi Takahashi[3]

[1] Dept. of Computer Science, Chubu University, Japan
[2] Dept. of Electronic Engineering, Chubu University, Japan
[3] Dept. of Information Science, Meijo University, Japan
hf@cs.chubu.ac.jp, shiyou@vision.cs.chubu.ac.jp,
any@nn.solan.chubu.ac.jp, ttaka@ccmfs.meijo-u.ac.jp

Abstract. Multiple cameras have been used to get a view of a large area. In some cases, the cameras are placed so that their views are overlapped to get a more complete view. 3D information of the overlapping areas that are covered with two or three cameras can be obtained by stereo vision methods. By shifting the shutter timings of cameras and using our pseudo stereo vision method, we can output 3D information faster than 30 fps. In this paper, we propose a pseudo stereo vision method using three cameras with different shutter timings. Using three cameras, two types of shutter timings are discussed. In three different shutter timings, 90 points of 3D position for a sec are obtained because the proposed method can output 3D positions at every shutter timing of three cameras. In two different shutter timings, it is possible to calculate the 3D position at 60 fps with better accuracy.

1 Introduction

In a soccer robot match, it is important to calculate the position of an object as quickly as possible in order to control the robot by visual feedback. Also, it is necessary to calculate the 3D position of the ball, not 2D position on the soccer field, because some robots have an ability of striking a loop shot [1].

As an approach to implementing a high speed vision system, a 60 fps camera has been used in small-sized robot league [2, 3]. The system processes NTSC camera images at a 60 fps rate with double buffering. However, they can't calculate a 3D position because they use a single camera. Stereo vision using multiple cameras is needed for measuring the 3D position. They require cameras to synchronize with each other for tracking an object accurately and measuring its depth.

We have proposed a pseudo stereo vision method for calculating the 3D position of an object using two unsynchronized cameras [4]. The method can obtain the 3D position of a moving object at 60 fps making use of the time lags of the shutter timing between the two cameras. In this paper, we present two kinds of vision systems based on the pseudo stereo vision method using three

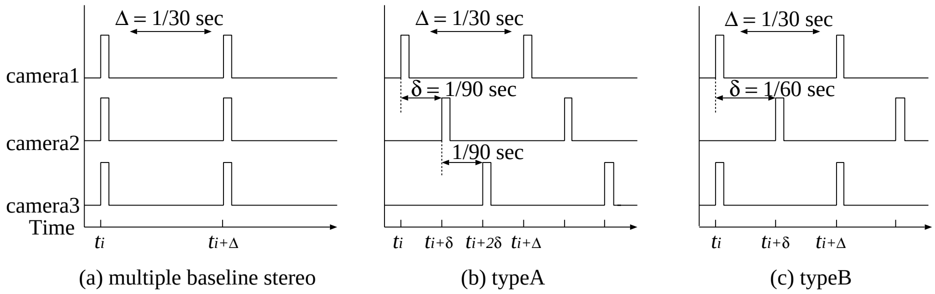


Fig. 1. Possible two combinations of shutter timing

normal cameras (that take pictures at 30 fps), which can output 3D positions at 60 fps or 90 fps by adjusting the shutter timing of each camera.

2 3D position measurement with multiple cameras

The stereo vision method which measures the 3D position of the object requires two images captured at the same time to reduce error in the measurement. Using a general stereo vision system, 3D positions can be obtained at 30 fps maximum using a normal 30 fps camera with fast vision algorithm described in [5].

Using two unsynchronized cameras for calculating the 3D positions of objects, we have proposed a pseudo stereo vision method taking advantage of time lag between the shutter timing of each camera [4]. To obtain a higher speed with better accuracy in 3D position, we investigate a vision system consisting of three cameras and a method for calculating the 3D position with two kinds of shutter timing of three cameras.

2.1 Shutter timings of three cameras

Three combinations of cameras might be considered as shown in Figure 1 by adjusting the shutter timings of the cameras. One of them is the case of same shutter timings which are used in multiple baseline stereo as shown in Figure 1(a). We focus on two cases where the shutter timing of each camera is different as shown in Figure 1(b) and (c). In the case of type A, the shutter timing of each camera is shifted for 1/90 second. Since the 3D position is calculated at every shutter timing of each camera, the 3D position can be obtained at 90 fps. In the case of type B, the shutter timings of camera1 and camera3 are synchronized, and the 3D position is calculated using stereo vision. The shutter timing of camera2 is shifted for 1/60 sec from the shutter timing of camera1 and camera3. The 3D position can be obtained at 60 fps, and we can obtain better accuracy. The methods for estimating the 3D position of two kinds of shutter timing are described as follows.

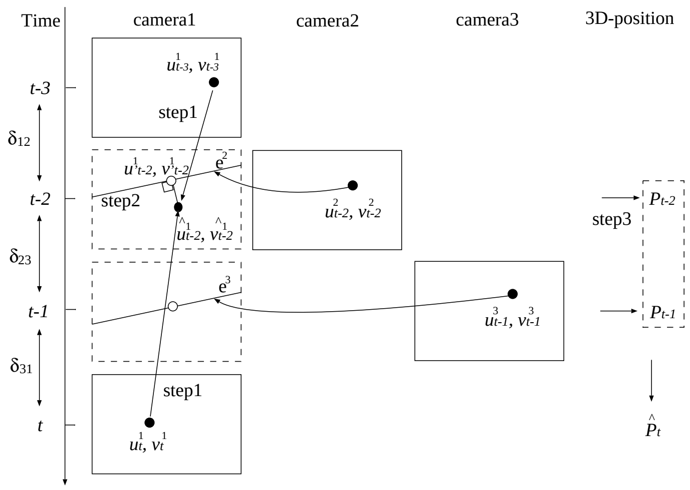


Fig. 2. Proposed calculation method of corresponding position

2.2 TYPE-A: Algorithm for three different shutter timings(90fps)

The 3D position in the last frame is estimated by using the time interval δ between the shutter timings of each camera and the results from the previous two frames. The procedure of 3D position measurement is as follows:

Step1 Calculation of 3D positions in the previous two frames
Step2 Linear prediction of the 3D position
Step3 Prediction using constraints from ray information

Calculation of 3D positions in the previous two frames
In order to obtain an accurate 3D position from the current frame by linear prediction, it is necessary to accurately calculate the previous 3D positions $\boldsymbol{P}_{t-1}$, $\boldsymbol{P}_{t-2}$. The algorithm of the 3D position calculation at $t-2$ is described as follows:

Step1 Using two observed points in the frame t and $t-3$, a pseudo-corresponding point from camera1 on frame $t-2$ $(u^{\hat{1}}_{t-2},\, v^{\hat{1}}_{t-2})$ is interpolated by the following equation:

$$u^{\hat{1}}_{t-2} = \frac{\delta_{12} u^1_t + (\delta_{23} + \delta_{31}) u^1_{t-3}}{\delta_{12} + \delta_{23} + \delta_{31}}, \quad v^{\hat{1}}_{t-2} = \frac{\delta_{12} v^1_t + (\delta_{23} + \delta_{31}) v^1_{t-3}}{\delta_{12} + \delta_{23} + \delta_{31}} \tag{1}$$

Step2 Calculate the epipolar line e^2 on the image from camera1 using the corresponding point (u^2_{t-2}, v^2_{t-2}) from the image from camera2. Then, the nearest point $(u'^1_{t-2},\, v'^1_{t-2})$ from the interpolated point, calculated by step1, can be set as corresponding point for $(u^2_{t-2},\, v^2_{t-2})$.
Step3 We can measure the 3D position $\boldsymbol{P}_{t-2}$ using triangulation as a crossing point of the two lines in 3D space.

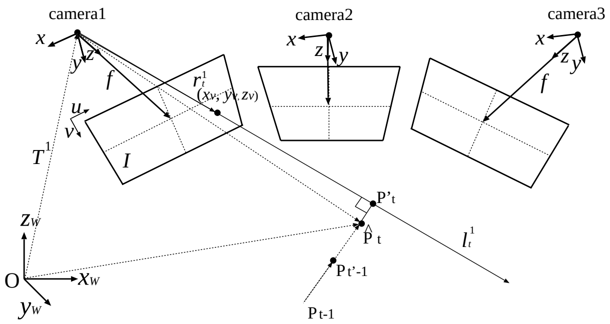


Fig. 3. Constraints in estimating 3D position

Step 1, 2, 3 are repeated in the frame $t-1$. The two previous 3D positions, at $t-1, t-2$ are calculated from previous and following frames and the epipolar constraint. These two previous points, $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$, will be used and decrease the prediction error at the next step of linear prediction.

Linear prediction of 3D position

As shown in Figure 3, the predicted position $\hat{\boldsymbol{P}}_t = [x_w, y_w, z_w]^T$ of the last frame t is calculated by the following equation using the already measured positions $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$ in the previous two frames.

$$\hat{\boldsymbol{P}}_t = \boldsymbol{P}_{t-1} + \frac{\delta_{31}(\boldsymbol{P}_{t-1} - \boldsymbol{P}_{t-2})}{\delta_{12}} \tag{2}$$

Note that Equation (2) is based on the analyzed image of the last frame from camera1. For the position estimate, the Kalman filter [6, 7] and spline curve fitting have been proposed.

Prediction using constraints from ray information

In order to decrease the prediction error, the 3D position is calculated once more using the constraint of a viewing ray in 3D space obtained from the current image. Let $\boldsymbol{T}^1 = [T_x, T_y, T_z]$ be the translation matrix from the origin of the world coordinate to the focus point of camera1, and $\boldsymbol{r}_t^1 = [x_w, y_w, z_w]^T$ be the vector which denotes the direction of the viewing ray, $\boldsymbol{l}_t^1$, passing through the position on the image coordinate (u_t^1, v_t^1) and the focus point of the camera. The viewing ray shown in Figure 3 can be expressed by

$$\boldsymbol{l}_t^1 = k\boldsymbol{r}_t^1 + \boldsymbol{T}^1 \tag{3}$$

where k is a real number. Although Equation (2) gives a good 3D position prediction, the position may not exist on the viewing ray $\boldsymbol{l}_t^1$ as shown in Figure 3 because of its prediction error. In order to solve this problem, the 3D position $\boldsymbol{P'}_t$ is calculated by the following equation as the nearest point on the ray $\boldsymbol{l}_t^1$.

$$\boldsymbol{P}'_t = \frac{(\hat{\boldsymbol{P}}_t - \boldsymbol{T}^1) \cdot \boldsymbol{r}_t^1}{|\boldsymbol{r}_t^1|^2} \boldsymbol{r}_t^1 + \boldsymbol{T}^1 \tag{4}$$

Finally, $\boldsymbol{P}'_t$ will be the 3D position of the object at the last frame. In case camera2 and camera3 are the latest frame, the image of the 3D position is calculated in the same way as mentioned above.

2.3 TYPE-B: Algorithm for two different shutter timings(60fps)

In order to estimate better 3D position by linear prediction, it is important to calculate 3D positions in the previous two frames. In the case of Figure 1(c), the shutter timings of two cameras(camera1 and camera3) are synchronized so that 3D position is calculated by stereo vision. Therefore, $\boldsymbol{P}'_t$ is estimated by linear prediction using $\boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-3}$, which are obtained by stereo. Futhermore, 3D position of camera2 $\boldsymbol{P}'_t$ is calculated by constraint from ray information using the same algorithm as type A.

Therefore, 3D positions, which are calculated by stereo from two synchronized cameras and estimated by constraint from ray information of a single camera are obtained respectively. In this case, the total number of points can be obtained for a second is 60 points, which is less than type A (90 points).

3 Simulation experiments

3.1 Recovery of object motions

We evaluated the proposed method by simulation of recovering the object's motion with uniform and non-uniform motion in 3D space (3,000 $\times$ 2,000 $\times$ 2,000 mm). In the simulation, we assumed that three cameras would be mounted at the height of 3,000 [mm].The proposed method is evaluated by following three motions.

- uniform motion(straight): An object moves to (x, y, z) = (3,000, 1,200, 0) from (x, y, z) = (0, 1,200, 2,000) at velocity of 3,000 mm/sec
- uniform motion(spiral) An object moves in a spiral by radius of 620 mm at velocity of 3,000 mm/sec at center (x, y) = (1,000, 1,000)
- non-uniform motion: An object falls from the height of 2,000mm, then an object 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 by the proposed method described in section 2.1 using the projected point on the virtual image plane (u, v) of each camera.

3.2 Simulation results

Table 1 shows averages of estimation error with simulation experiments. The unsynchronized method in Table 1 shows the result of stereo vision by corresponding points in time delay using two cameras, and the synchronized method shows the result of general stereo vision with no time delay. In the case of using two cameras, it is clear that the proposed method(type A) has a better result

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

	method	fps	uniform		non-uniform
			straight	spiral	
2cameras	unsynchronized	60	23.2	21.4	16.2
	synchronized	60	0.03	0.14	0.12
	type A	60	1.1	2.2	1.8
3cameras	type A	90	1.1	2.0	1.7
	type B	60	0.2	0.5	1.5
	linear prediction	60	0.2	1.4	4.4

than the unsynchronized method, and its accuracy is close to the synchronized method.

Linear prediction in Table 1 shows the result of linear prediction using the past two positions calculated by stereo vision. Comparing type B to linear prediction, it is clear that type B has better accuracy because constraint from ray information decreases the error generated by linear prediction. In the simulation experiment of non-uniform motion, type A has better accuracy compared to linear prediction even though the shutter timings of the three cameras are different. This is why the time interval of the shutter timing is small ($\delta = 1/90$ sec).

4 Experiments using real cameras

We evaluated our method using real data in the same way as the simulation experiments.

4.1 Configuration of vision system

Figure 4 shows the camera placement of our vision system that uses three cameras, camera1, camera2 and camera3. These cameras are mounted at a height of 2,800 mm, and each camera has a view of an area of 2,000 × 3,000 mm. Each camera is calibrated using corresponding points of world coordinates (x_w, y_w, z_w) and image coordinate (u, v) [8]. The shutter timing of each camera is controlled by a TV signal generator. Three frame grabbers for the three cameras are installed on a PC. Our hardware specifications are described as follows:

Process-1, process-2 and process-3 analyze images from camera1, camera2 and camera3 at every 1/30 second respectively. The analyzed results such as (u^i, v^i) and the time instant at which the analyzed image was captured are sent via UDP interface to process-4 that calculates the 3D positions of the object. There is negligible delay due to communications among processes because this work is done on the same computer.

4.2 Experiments

Figure 6 shows results of recovering the motion of a hand-thrown ball for about 1.5 sec. Figure 6(a) shows that the numbers of plotted points is 135. This indi-

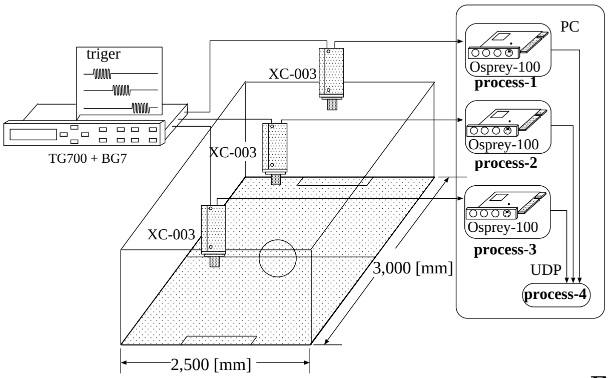


Fig. 4. Overview of our vision system

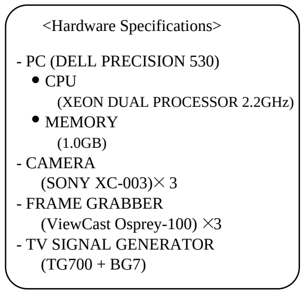
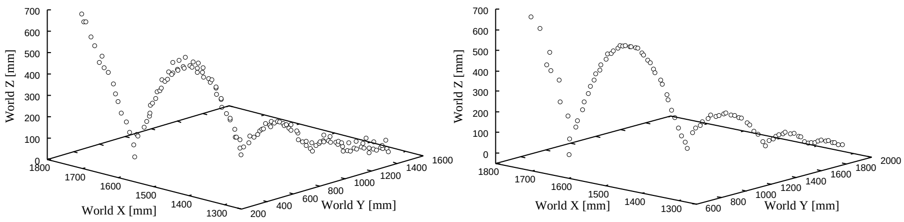


Fig. 5. Hardware specifications

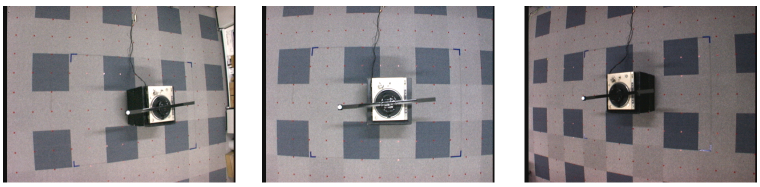


(a) typeA (b) typeB

Fig. 6. Results of 3D position measurement

cates that the speed is the same as 90 fps camera. Figure 6(b) shows that the numbers of plotted points is 90, which is same as 60 fps camera.

As an evaluation for the accuracy of estimated 3D positions, we used a turntable and a ball as shown in Figure 7. A ball attached on the edge of ruler(1,000 mm length) makes a uniform circular motion with a radius of 500 mm. The turntable is placed on a box at the height of 500 mm, and the ball's height from the floor is 660 mm. The turntable rotates at a speed of 45 rpm, and its rotation speed per second is $(45 \times 2\pi)/60 = 0.478$ radian.



(a) image from camera1 (b) image from camera2 (c) image from camera3

Fig. 7. Captured images of turntable

Table 2. Average and variance of z values of 3D positions

	average[mm]	variance
type A	664.5	2143.7
type B	662.3	112.0

Table 2 shows the average and variance of the 3D position on the z_w axis for both types. The average of the positions from the two methods was measured within 5 mm from the actual height of 660 mm. We see that variance of type B is smaller than type A, which is the same result as the simulation experiments.

5 Discussion and Conclusion

We proposed a pseudo stereo vision method using cameras with different shutter timings. The method can output 3D position at 60 fps or 90 fps by adjusting the shutter timing of three cameras. In three different shutter timings (type A), 90 points of 3D position for a sec are obtained because the proposed system can output 3D positions at every shutter timing of the three cameras. In two different shutter timings (type B), it is possible to calculate the 3D position at 60 fps with better accuracy.

In RoboCup small-size league, some teams have used multiple cameras to get the robot's position with better precision than one camera. From 2004, the soccer field will become larger than the size that one camera can cover. Using our method, high speed and 3D information of the overlapped area can be obtained.

References

1. Muratec FC.
http://www.muratec.net/robot/
2. R. D'Andrea, et al. Detailed vision documentation.
http://robocup.mae.cornell.edu/
3. S. Hibino, Y. Kodama, Y. Nagasaka, T. Takahashi, K. Murakami and Tadashi Naruse. Fast Image Processing and Flexible Path Generation System for RoboCup Small Size League, RoboCup2002, pp.53-64, 2002.
4. S. Shimizu, H. Fujiyoshi, Y. Nagasaka and T. Takahashi. A Pseudo Stereo Vision Method for Unsynchronized Cameras. ACCV2004, vol.1, pp.575-580, 2004.
5. J. Bruce, T. Balch and M. Veloso. Fast and Inexpensive Color Image Segmentation for Interactive Robots, IROS-2000, vol.3, pp.2061-2066, 2000.
6. B. Browning, M. Bowling and M. Veloso. Improbability Filtering for Rejecting False Positives, In Proc. IEEE International Conference on Robotics and Automation, pp.120-200, 2002.
7. K. Horiuchhi, S. Kaneko and T. Honda. Object Motion Estimation based on Multiple Distributed Kalman Filters, In the IEICE, Vol.J79-D-II, Num.5, pp.840-850, 1996.
8. R. Y. Tsai. 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.