Proceedings of the 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems

Object Tracking Based on Optical Flow and Depth

Ryuzo Okada, Yoshiaki Shirai and Jun Miura

Department of Mechanical Engineering for Computer-Controlled Machinery,
Osaka University
2-1, Yamadaoka, Suita, Osaka 565, JAPAN
E-mail: okada@cv.ccm.eng.osaka-u.ac.jp

Abstract

This paper describes a method to track an object based on optical flow and depth. The velocity and the depth of the target object are estimated from the histograms of the velocity and that of the disparity. A target region is extracted by Baysian inference using optical flow, disparity and the predicted target location. This method works even if tracking with either velocity data or disparity data alone may fail. Occlusion of the target can also be detected from the abrupt change of the disparity of the target region. Our method successfully tracked a moving person using a real image sequence.

1 Introduction

Visual object tracking is necessary for various applications such as autonomous vehicle navigation and human interface.

Several techniques for visual object tracking have been proposed. Some of them are based on differentiation between frames[1]. The region which has large differentiation value is extracted as an object region. This method cannot be applied to a case in which the camera moves because the background changes. In correlation based methods[2], object tracking is difficult if the appearance of the object changes. The methods using depth data [3][4][5] are proposed. The object depth is predicted from the result of the previous frame. The region which has the similar depth to the predicted object depth is determined to be the object region. This method is robust against change of the object shape. However, the object which has the similar depth to the target object cannot be separated from the target object. There are methods based on the velocity data[6][7]. The target location and velocity are predicted using the result of the previous frame. The target region is estimated as the region which has similar velocity to the predicted target velocity in the predicted target region. This method cannot work if the velocity of other objects are similar to the target object. The method based on single cue such as velocity or depth cannot separate objects with similar velocity or depth.

This paper proposes to use two cues: optical flow and disparity (see figure 1). At each frame, optical flow and disparity are calculated. The target velocity and disparity are estimated using the optical flow and the disparity inside the predicted target region. The region which has the similar velocity and disparity to the target is extracted as the target region. Occlusion of the target is detected from the abrupt disparity change in the target region. A position of occluded part of the target is estimated based on the past records of the target region. When the target is completely occluded, tracking is continued using the latest velocity of the target.

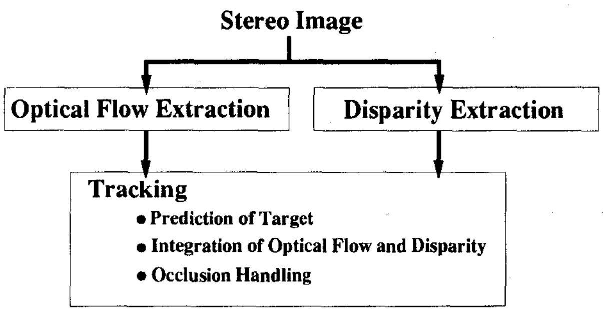


Figure 1: Our system overview

2 Optical Flow Extraction

2.1 Generalized Gradient Method based on Spatio-Temporal Filtering

Optical flow is calculated by the generalized gradient method[8] based on spatio-temporal filtering.

Let $f(x, y, t)$ denotes brightness at a point (x, y) in an image at time t. If this point moves to a point $(x + dx, y + dy)$ at time $t + dt$, the following equation holds:

$$f(x, y, t) = f(x + dx, y + dy, t + dt). \quad (1)$$

0-7803-3700-X/96 $5.00 ©1996 IEEE

Taylor expansion of the right side of the equation (1) is

$$\begin{aligned} f(x,y,t) &= f(x,y,t) + f_x(x,y,t)dx \\ &+ f_y(x,y,t)dy + f_t(x,y,t)dt + e \end{aligned} \quad (2)$$

where $f_x = \frac{\partial f}{\partial x}$, $f_y = \frac{\partial f}{\partial y}$, $f_t = \frac{\partial f}{\partial t}$, and e is the high order terms of dx,dy,dt.

Assuming that e is negligible, we obtain the next equation:

$$f_x(x,y,t)u + f_y(x,y,t)v + f_t(x,y,t) = 0 \quad (3)$$

where $u = \frac{dx}{dt}, v = \frac{dy}{dt}$, (u,v) is flow vector. This is the constraint equation of the gradient method.

By applying two different spatial filters g,h to the input image $f(x,y,t)$, the following two constraint equations are derived.

$$\begin{pmatrix} (g*f)_x & (g*f)_y \\ (h*f)_x & (h*f)_y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} (g*f)_t \\ (h*f)_t \end{pmatrix} \quad (4)$$

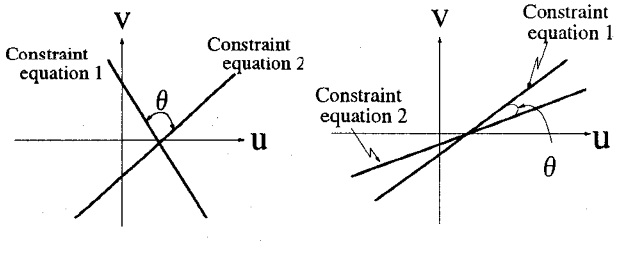
where $*$ represents convolution. The flow vector (u,v) is obtained by solving these simultaneous equations.

2.2 Reliability of Optical Flows

We define the reliability of the flow by the angle between two lines corresponding to equations (4) as shown in figure 2. The reliability R_v is given as:

$$R_v = |sin\theta| = \frac{|\mathbf{G} \times \mathbf{H}|}{|\mathbf{G}||\mathbf{H}|} \quad (5)$$

where $\mathbf{G} = ((g*f)_x, (g*f)_y)$ and $\mathbf{H} = ((h*f)_x, (h*f)_y)$. Note that the right side of this equation is the absolute value of the determinant of the coefficient matrix of the flow vector normalized by the spatial gradient of brightness. If the reliability R_v is small, the flow vector is not calculated.



(a) high reliability

(b) low reliability

Figure 2: reliability of optical flow

If the contrast is small, the flow vector is not reliable, either. The reliability R'_v of the contrast is given as:

$$R'_v = (g*f)_x^2 + (g*f)_y^2 + (h*f)_x^2 + (h*f)_y^2 \quad (6)$$

If the reliability R'_v is small, the flow vector is not calculated at the point.

3 Disparity Extraction

3.1 Calculation of Disparity

The disparity is derived from one-dimensional optical flow between a pair of stereo images. One-dimensional optical flow is calculated based on the gradient method as stated in subsection 2.1. In case of calculating disparity, the constraint equation is as follows:

$$\frac{f^R{}_x(x,y) + f^L{}_x(x,y)}{2} d + f^L(x,y) - f^R(x,y) = 0 \quad (7)$$

where f^L and f^R denote the left and the right image respectively, and d denotes disparity.

If the target is close to the observer, the scope of the scene visible from both cameras is small. We use the vergence motor of active camera head (turning the right camera inside) to enlarge the scope. Let ϕ denotes a rotation angle between optical axes of the right and the left camera (see figure 3), the epipolar constraint equation is as follows:

$$y_R = \frac{sin\phi}{f} x_R + Y_L cos\phi \quad (8)$$

where (x_R, y_R) denotes the coordinate of right image, and Y_L denotes the y-component of a point A on left image, and f denotes the focal length. The right image is transformed so that epipolar lines are horizontally aligned. Thus we can calculate the disparity based on equation (8).

3.2 Reliability of disparity

If the contrast is small for a point, then the disparity of the point is not reliable. We define the reliability of the disparity as follows:

$$R_d = (f^R{}_x)^2 + (f^L{}_x)^2 \quad (9)$$

If R_d is small for a point, then the disparity of the point is not calculated.

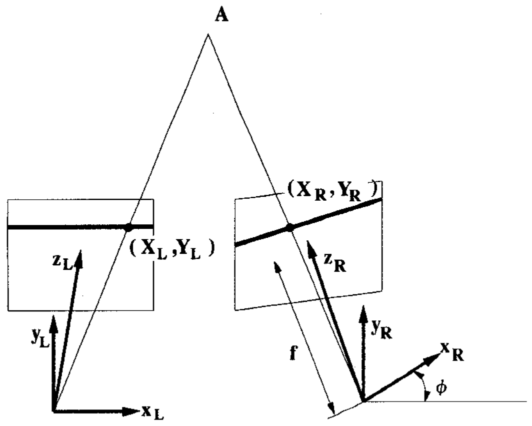


Figure 3: coordinate system

4 Tracking

Initially a target object is given to the system by circumscribing the target region in the image by a rectangle. This rectangle is called a *target window.* A candidate region of the target is predicted from the target window of the previous frame. Initially, however, the candidate region is the initial target window itself.

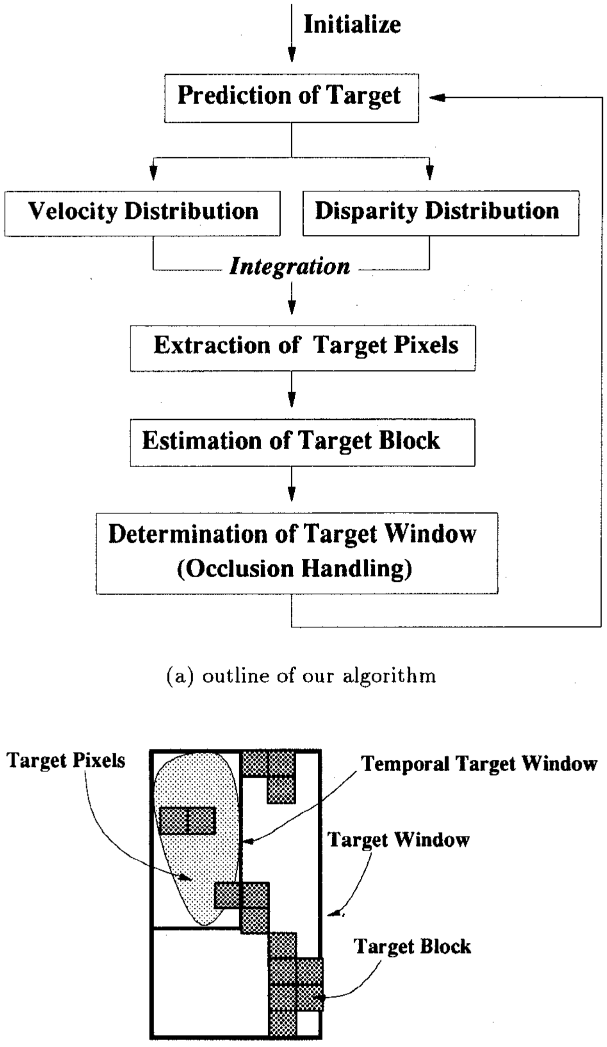
We use histograms of velocity u, v and disparity d in the target window to estimate the distribution of the target velocity and that of the target disparity. The histograms of u and v are weighted by the reliability of optical flow (equation (5)). The histogram of d is weighted by the reliability of disparity (equation (9)).The peak of the estimated distribution is determined to be the representative target velocity (u_{obj},v_{obj}) and disparity d_{obj}, respectively.

The probability of the pixel belonging to the target is calculated based on the distributions of the target velocity, that of the target disparity and the predicted candidate region of the target. The pixels with high probability are extracted as the points belonging to the target. These pixels are called *target pixels.* The rectangle circumscribing the target pixels is called a *temporal target window* (see figure 4(a)).

The target pixels are not reliable because the extracted optical flow and disparity are noisy. The target window is divided into small fix-sized rectangles at the initial frame. This rectangle is called a *block.* The block moves at the same velocity as that of the target window. If enough target pixels are extracted in a block in many frames, the block is determined to be a part of the target and called *target block.* The rectangle circumscribing the target pixels and the target blocks is a new *target window.* In the next frame, the candidate target region is predicted as the target window shifted by the representative target velocity (u_{obj},v_{obj}).

If the mean disparity in the target block becomes abruptly large, the region is determined to be occluded and its position is estimated by shifting the previous position by the representative target velocity. This block is called an *occluded block.* If the mean disparity in a occluded block is similar to the target disparity d_{obj}, this block becomes target block. Since there may be an occluding object in the target window, the histograms of the target velocity and disparity are affected by the occluding object. The histograms are therefore made in the temporal target window shifted by the representative target velocity (u_{obj},v_{obj}).

If the target is completely occluded, the target window is estimated assuming the target moves with a constant velocity so that the target may be tracked again when it comes out of the occluding object.



(a) outline of our algorithm

(b) terminological definition

Figure 4: tracking algorithm

4.1 Prediction of the target location

If the frame interval is short enough, the target moves with nearly constant velocity between two consecutive frames. The candidate target region at the current frame is predicted by shifting the previous target window by the previous representative target velocity (u_{obj},v_{obj}) and enlarging it (see figure 5). We define the prior probability of the pixel belonging to the target inside the candidate target region as follows. The points around the center of the candidate target region have higher probability of the points being on the target than that of the points near the edge of the candidate target region. The center region has the probability of 1.0, and the probability is reduced linearly toward the edge of the candidate target region.

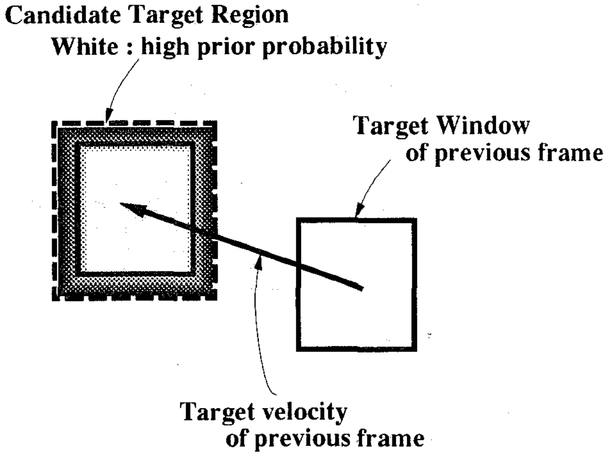


Figure 5: candidate target region and prior probability (The higher the intensity is, the higher the prior probability is.)

4.2 Estimation of the distribution of target velocity and disparity

Weighted histograms We use histograms of velocity u, v and the disparity d to estimate the distribution of the target velocity and that of the target disparity. These three histograms are made in the previous temporal target window shifted by the representative target velocity of previous frame because the target window may include the occluded target region. The histograms of u and v are weighted by the reliability of optical flow (equation (5)). The histogram of d is weighted by the reliability of disparity (equation (9)).

Representative velocity and disparity of the target The peak of the histograms is determined to be the representative target velocity (u_{obj},v_{obj}) and the target disparity d_{obj} respectively.

Estimation of distribution The distributions are estimated based on the weighted histograms. Since the weighted histograms are noisy, the histograms are smoothed. Since we approximate the target pixels with the rectangle temporal target window and make histograms in the shifted temporal target window, the histograms are affected by the background and other objects. The part of the histogram corresponding to the target is extracted to estimate the distributions of the target velocity and that of disparity (see figure 6). The position of the local minimum on each side of the peak u_{obj} are searched for. Let u_- and u_+ denote the velocity of the local minimum less than u_{obj} and greater than u_{obj} respectively. If there is the local minimum u_+, the distribution of the target for the velocity greater than u_+ is estimated by the line P_+P_0 (extention of the line $P_{obj}P_+$), and the distribution between u_{obj} and u_+ is estimated by the histogram. If there is not local minimum u_+, we use the histogram for the distribution between u_{obj} and u_{max}. So with the velocity of local minimum u_-. The integral of the modified histogram is normalized to 1.

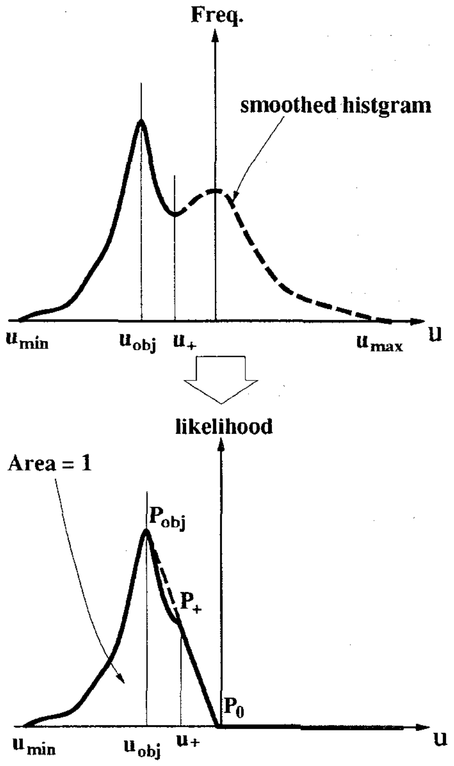


Figure 6: distribution of target velocity u

4.3 Extraction of target pixels

The target pixels are extracted in and around the temporal target window shifted by the representative target velocity of the previous frame. Let $\mathbf{F}(x,y)$ be the observation vector at (x,y), and is defined as follows:

$$\mathbf{F}(x,y) = (u(x,y), v(x,y), d(x,y)). \tag{10}$$

Let $P(obj|\mathbf{F})$ denote the probability of the pixel (x,y) being on the target object. $P(obj|\mathbf{F})$ is calculated as follows:

$$P(obj|\mathbf{F}) = \frac{P(\mathbf{F}|obj)P(obj)}{P(\mathbf{F})}. \tag{11}$$

$P(\mathbf{F})$ is constant. Assuming u, v and d are independent of each other, $P(\mathbf{F}|obj)$ is represented as follows:

$$P(\mathbf{F}|obj) = P(u|obj)P(v|obj)P(d|obj) \tag{12}$$

where $P(u|obj)$, $P(v|obj)$, $P(d|obj)$ are likelihoods with respect to the elements of the observation vector. We define the distributions (described in section 4.2) of the velocity and the disparity as the likelihoods. $P(obj)$ is the prior probability of the pixel (x,y) belonging to the target object (described in section 4.1).

The pixels which have high probability are extracted as the point belonging to the target object. These pixels are target pixels.

4.4 Estimation of the Target Block

The target pixels extracted only from one frame are not reliable because the extracted optical flow and disparity are noisy. The target window is divided into small fix-sized rectangles at the initial frame. This rectangle is called *block*. The blocks move at the same velocity as the target window's velocity. Each block has the following states.

1. A block belongs to the target, and is not occluded. (*Target*)
2. A block belongs to the target, and is occluded. (*Occluded*)
3. A block does not belong to the target. (*Not_Target*)

All the initial states of blocks are *Not_Target*. If there are enough target pixels in a block, the block gets one vote. An enough number of votes changes the state *Not_Target* to *Target*. If the target pixels are not extracted in a block whose state is *Target*, the average disparity in the block is calculated. If the average disparity is much larger than the representative target disparity d_{obj}, the state *Target* changes to *Occluded* because the block belongs to the occluded target region. If the average disparity is not much larger than the representative target disparity d_{obj}, the block gets one negative vote. An enough number of negative votes changes the state *Target* to *Not_Target*. If the average disparity in a block of state *Occluded* becomes similar to the representative target disparity d_{obj}, the state of the block changes again to *Target*.

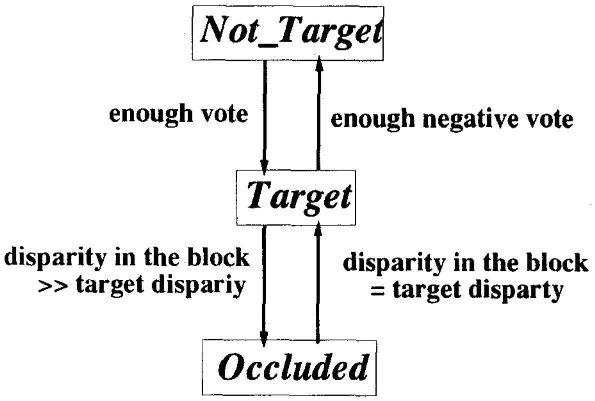

Not_Target
enough vote
enough negative vote
Target
disparity in the block >> target dispariy
disparity in the block = target disparty
Occluded

Figure 7: state transition of block

4.5 Determination of the target window

The rectangle circumscribing the target blocks, occluded blocks and the target pixels is the target window unless the target is completely occluded.

We decide that the target is completely occluded when either of the following two conditions is satisfied.

- The number of the target pixels is too small and there are the occluded blocks.
- The current representative target disparity d_{obj} is much larger than that of the previous frame.

Assuming that the occluded target moves with a constant velocity and the shape of the target do not change, the representative target velocity (u_{obj},v_{obj}) and the tracking window does not change. If the disparity extracted in the current target window, which is the disparity of the occluding object, is much smaller than that of the previous frame, we decide that the target appears again.

5 Experimental result

Figure 8 shows the experimental result of our method. The white regions are the target pixels, and the black rectangles represent the target blocks and occluded blocks. The white rectangle represents the target window.

Two persons walk from right to left in the scene. The target person is in the center of the image No. 1. The person on the right side in the image No. 1 is nearer to the camera than the target person. As he walks a little faster than the target person, he overtakes the target person. In this situation, tracking using only the optical flow will fail because of similarity of the velocity. The target person is occluded from image No. 25 to No. 40. After the target object is overtaken, he passes by the standing person. Since the distances of the target object and the standing person are similar, the tracking using only depth data will fail.

Since the system cannot so far control the active camera head synchronizing with the target person because of the hardware limitation, we take the image sequence by rotating the active camera head with a constant angular velocity. The number of the images in the sequence is 100. The duration of the image sequence is about 6.7 seconds. The resolution of an image is 160 × 120 pixels.

The target person was successfully tracked in our method. In the 28th frame, the system detects that the target person is completely occluded, and appears in the 39th frame. The feet of the target person are not extracted as the target region. This is because we assume that the target velocity is uniform in the target region.

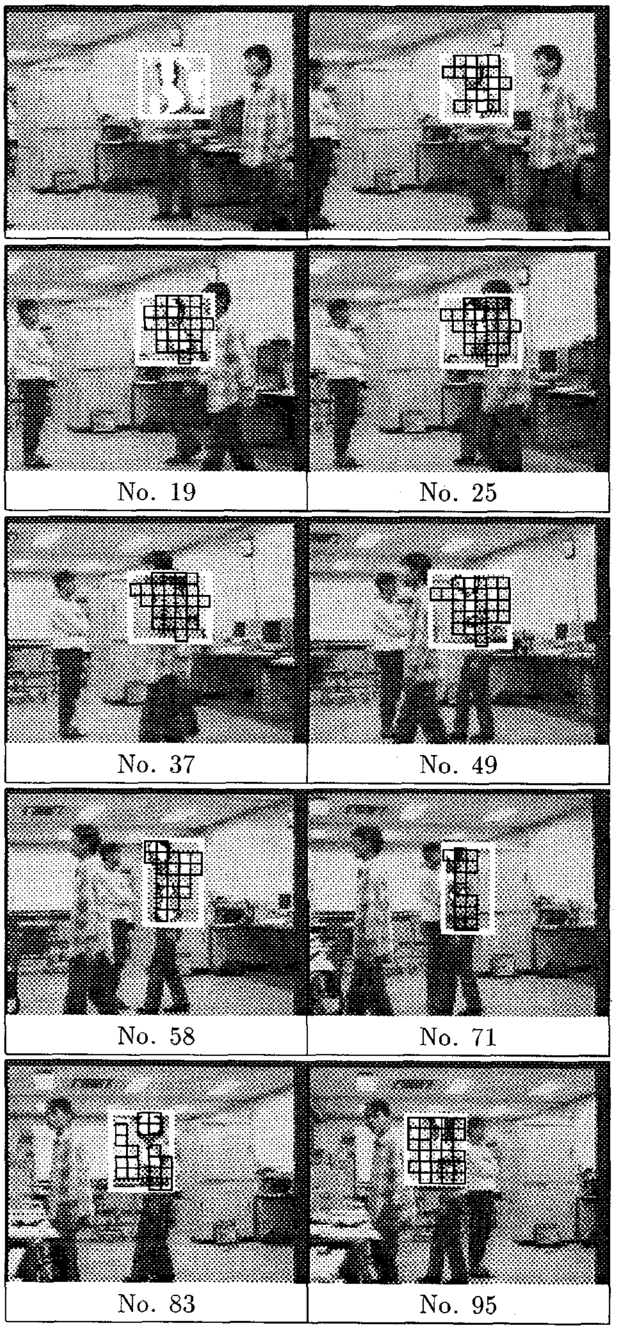
6 Realtime tracking

The hardware configuration of our system is illustrated in figure 9. The image processor processes the sequence of stereo images taken by the stereo active camera head, and put the result of tracking out to the monitor. The image processor also sends the motor commands to the active camera head controller.

The image processor has many DSPs (Digital Signal Processor) . Each DSPs is mounted on a DSP-board which has memories and interface for data transfer. The programs of the DSPs and the connection between the each DSP-boards can be changed as the implemented algorithm.

The processing of the each DSPs and the connection between the DSP-boards on our system is shown in figure 10. The optical flow is calculated in DSP1, 2, 3, 4, 5 as described in section 2. The disparity is calculated in DSP6, 7, 8, 9, 10 as described in section 3. DSP11 gathers the optical flow and the disparity, and sends them to the nest DSP-board. DSP12 carries out the tracking described in section 4, sends the motor command to the active camera head controller.

The tracking algorithm based on only optical flow is so far implemented on the image processor[6]. This system can track target objects in realtime(15 Hz).



No. 19 No. 25

No. 37 No. 49

No. 58 No. 71

No. 83 No. 95

Figure 8: experimental result

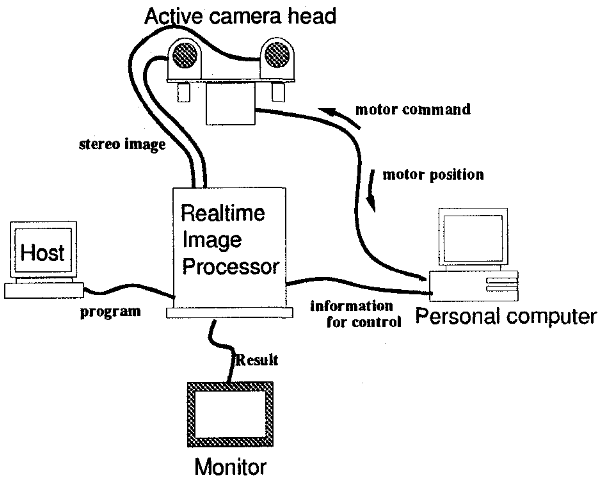


Figure 9: hardware configuration for realtime tracking

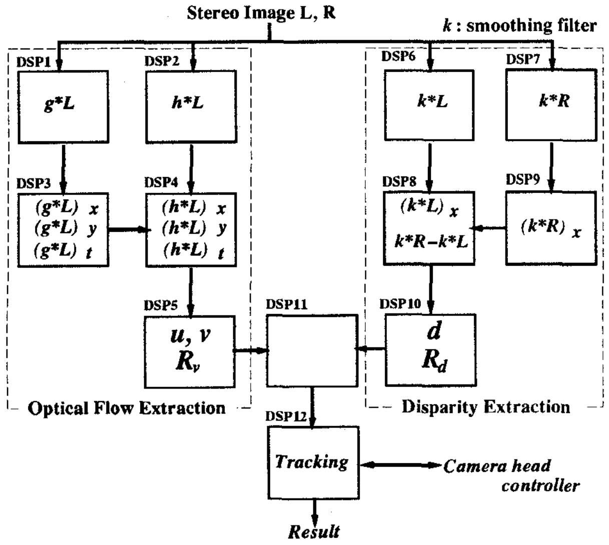


Figure 10: processing and connection of DSP-boards

7 Conclusion

In this paper, we have proposed a method to track an object by integrating optical flow and disparity. The probability distributions of the target velocity and that of disparity are calculated from the optical flow and the disparity in the predicted target region. Using these distributions along with the prior probability obtained from the predicted target position, the target region is extracted using Baysian inference. Occlusion of the target can be also detected from the abrupt change of the disparity in the target region. Our method successfully tracked a moving person using the real image sequence.

In our method, there is so far a assumption that the optical flow is uniform inside the target region. It is a future work to extend our method to be applicable to objects with various motion such as: articulated objects, objects moving towards an observer, and rotating objects.

References

[1] M.Yachida, M.Asada, and S.Tsuji. Automatic analysis of moving image. IEEE Trans. Pattern Anal. Mach. Intell. Vol.PAMI-3, No.1, pp.12-20, 1981.

[2] H.Inoue, T.Tachkawa, and M.Inaba. Robot vision system with a correlation chip for real-time tracking, optical flow and depth map generation. Proc. IEEE International Conference on Robotics and Automation, pp.1621-1626, 1992.

[3] D.Coombs and C.Brown. Real-time smooth pursiut tracking for a moving binocular robot. Proc. CVPR'92 pp.23-28, 1992.

[4] N.Kita, S.Rougeaux, Y.Kuniyoshi, and S.Sakane. Throuth zdf-based localization for binocular tracking. IAPR Workshop on Machine Vision Applications, pp.190-195, 1994.

[5] A.Maki, T.Uhlin, and J.-O.Eklundh. Disparity selection in binocular pursuit. IAPR Workshop on Machine Vision Applications, pp.190-195, 1994.

[6] S.Yamamoto, Y.Mae, Y.Shirai, and J.Miura. Realtime multiple object tracking based on optical flows. Proc. Robotics and Automation, Vol.3 pp.2328-2333, 1995.

[7] H.J.Chen, Y.Shirai, and M.Asada. Detecting multiple rigid image motions from an optical flow field obtained with multi-scale, multi-orientation filters. IEICE Trans. Inf. & Syst., Vol.E76-D, No.10, 1993.

[8] M.V.Srinivasan. Generalized gradient schemes for the measurement of two-dimensional image motion. Biolgical Cybernetics, Vol.63, pp.421-431, 1990.