MVA'98 IAPR Workshop on Machine Vision Applications, Nov. 17-19, 1998, Makuhari, Chiba, Japan

12—3

Optical Flow-Based Person Tracking by Multiple Cameras

Hideki Tsutsui, Jun Miura, and Yoshiaki Shirai
Department of Computer-Controlled Mechanical Systems, Osaka University

Abstract

This paper describes optical flow-based person tracking using multiple cameras in an indoor environment. There are usually several objects in indoor environments which may obstruct a camera view. If we use only one camera, tracking may fail when the target is occluded by the objects. By using multiple cameras, this problem can be solved. In our method, each camera tracks the target person independently. By exchanging information among cameras, the three dimensional position and the velocity of the target are estimated. When a camera loses the target by occlusion, the target position and the velocity in the image are estimated using information from other cameras which are tracking the target.

1 Introduction

To track a moving object from an image sequence is one of the most important problems in computer vision. Visual object tracking is useful for various applications such as visual surveillance and gesture recognition.

In a usual indoor environment, the target object is often occluded by other object; in such a situation tracking becomes difficult.

Yamamoto et al.[2] proposed a method of multiple object tracking based on optical flow. By tracking all moving objects, the method can track a person even when he is occluded by another person. However, since object extraction uses optical flow, only moving objects can be extracted. If the target is occluded by a stationary object, the method cannot track him.

Rao and Durrant-Whyte [3], and Utsumi et al.[4] proposed methods of multiple persons tracking using multiple cameras. In these work, an object region in the image is extracted by subtraction of the background image from the current image. Thus, when multiple objects overlap in one image, the system cannot distinguish them.

We propose an optical flow-based tracking method using multiple cameras. Using optical flow is an effective way to distinguish multiple moving objects. By using multiple cameras, the system can track the target even if the target is not observed by several cameras.

In this paper, we call a camera which is tracking the target a *tracking* camera and a camera which loses the target a *lost* camera.

Usually each camera tracks the target independently. When a camera loses the target due to occlusion, it searches for the target in the image using information of the target position and velocity obtained by other *tracking* cameras. This method of exchanging information between cameras realizes a robust person tracking in a cluttered environment.

2 Tracking Moving Object by Single Camera

Optical flow is calculated based on the generalized gradient method using multiple spatial filters [1]. We assume that an object region in the image has almost uniform flow vectors. An object is tracked by updating a rectangular window which circumscribes the object region.

We assume that the moving object which comes into the view first is the target, and set a window as the initial region where an enough number of flow vectors are obtained. We call it the *tracking window*. Once the target is obtained, it is tracked by the following procedure (see Figure 1):

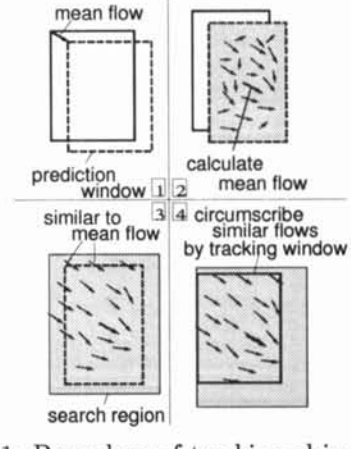

Figure 1: Procedure of tracking object region

1. A window is set where the tracking window is shifted by the mean flow of the previous frame. We call it the *prediction window*. In the initial frame, the prediction window is set at the initial region.
2. The mean flow is calculated in the prediction window.
3. Pixels whose flow vectors are similar to the mean flow are searched for in the prediction

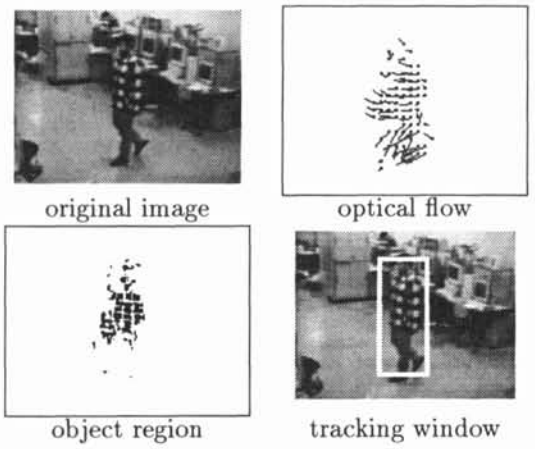

Figure 2: Extraction of object region

window and its neighborhood. The object region is generated as the set of such pixels.

4. The tracking window is set to circumscribe the object region.

Figure 2 shows a result of optical flow-based person tracking. As shown in this figure, this tracking method often fails to extract the feet because the velocity of the feet is different from that of the body.

3 Tracking Target

3.1 Target Position Estimation by Multiple Cameras

We model a person with a vertical cylinder and its height and radius are set to the height and the width of the person.

Let $\boldsymbol{X} = [X, Y, Z, 1]^t$ denote a point in the world coordinate system, and $\boldsymbol{x} = [x, y, 1]^t$ denote the projected point of $\boldsymbol{X}$ to the image. The following equation is satisfied:

$$h\boldsymbol{x} = \boldsymbol{C}\boldsymbol{X} \tag{1}$$

where h denote a scale factor and $\boldsymbol{C}$ denote the camera parameter:

$$\boldsymbol{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{bmatrix}. \tag{2}$$

If the object region is correctly extracted, the center of gravity of the object region $\boldsymbol{x}$ is determined. The corresponding three dimensional position of the target $\boldsymbol{X}$ satisfy the equation (1). The target position is obtained as the intersection of projection lines from at least two *tracking* cameras. However, since most part of the feet is often excluded from the object region as shown in Figure 2, the vertical position of the center of gravity in the image is not reliable. Thus we use only the horizontal position of the center of gravity to estimate the target position.

We consider the plane which contains a projection line and is perpendicular to the floor. We call it a *target plane*. The axis of the target cylinder is obtained on this plane. A target plane is given by the following equation:

$$\boldsymbol{a} \begin{bmatrix} X \\ Y \end{bmatrix} = b \tag{3}$$

where

$$\boldsymbol{a} = \begin{bmatrix} (C_{11} - C_{31}x)(C_{23} - C_{33}y) - (C_{13} - C_{33}x)(C_{21} - C_{31}y) \\ (C_{12} - C_{32}x)(C_{23} - C_{33}y) - (C_{13} - C_{33}x)(C_{22} - C_{32}y) \end{bmatrix}^t \tag{4}$$

$$b = (C_{13} - C_{33}x)(C_{24} - C_{34}y) - (C_{14} - C_{34}x)(C_{23} - C_{33}y) \tag{5}$$

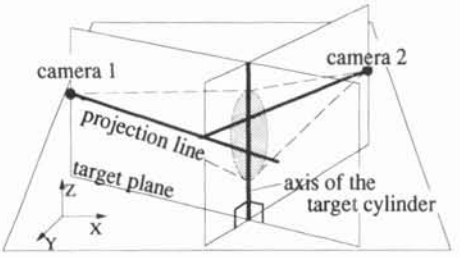

Figure 3: Estimating of target position

When there are at least two *tracking* cameras, the axis of the target cylinder is obtained as the intersection line of such target planes (Figure 3). If we have more than two target planes, the axis position is calculated by the least squares method as follows:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = (\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B} \tag{6}$$

where $\boldsymbol{A} = [\boldsymbol{a}_1\boldsymbol{a}_2\boldsymbol{a}_3 \cdots \boldsymbol{a}_m]^t$ $\boldsymbol{B} = [b_1 b_2 b_3 \cdots b_m]^t$, and $\boldsymbol{a}_i$ b_i denote parameters of the target plane for the ith camera, m is the number of *tracking* cameras.

There are two cases where the above method does not work. One is the case where there is only one *tracking* camera.

The other case is explained as follows. If the rank of the matrix $\boldsymbol{A}$ is less than two, equation (6) cannot be solved. This happens when all target planes coincide. Thus, when the target planes make small angles (when $|\boldsymbol{A}^t\boldsymbol{A}|$ is small), the estimated position is not reliable.

The above two cases are described in section 3.2.

3.2 Target Area Estimation by One Camera

By considering the size of the object region in the image and the human model, the target area is estimated by using the information from only one *tracking* camera.

Estimating Range of Depth As described in section 2, a part of the target is extracted and circumscribed by the tracking window. The top of the head is not lower than that of the tracking window, and the bottom of the foot is not higher than that of

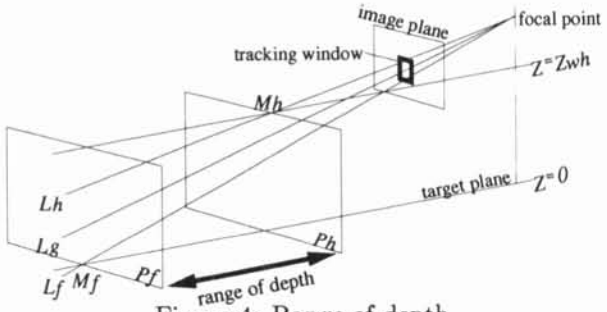

Figure 4: Range of depth

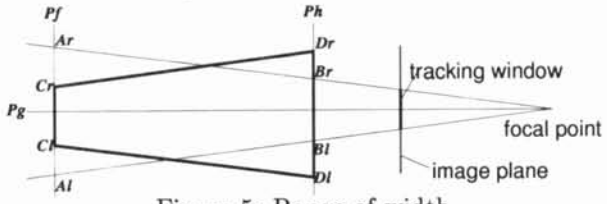

Figure 5: Range of width

the tracking window. This fact is used to estimate the possible depth range of the target.

In the target plane (Figure 4), let L_h and L_f denote the projection line through the top and the bottom of the tracking window, respectively.

First, we consider the plane $Z = Z_{wh}$, which indicates the height of the target person, and calculate the intersection M_h of this plane and L_h. Since the top of the head is not lower than that of the tracking window in the image, the person should be farther than M_h. Thus, plane P_h, which includes M_h and is perpendicular to both the target plane and the floor, represents one constraint on the target depth. Similarly, considering the condition of the foot position, we can obtain the other constraining plane P_f from line L_f and plane $Z = 0$. The target exists between P_h and P_f.

Estimating Range of Width Since the tracking window circumscribes a part of the target, the leftmost and the rightmost edge of the target never be inside the tracking window. Figure 5 shows the top view of figure 4. When the rightmost edge of the target is on line A_rB_r, the axis of the human model is on line C_lD_l, such that these lines are in parallel and the distance between the lines is equal to the radius of the model. Line C_lD_l represents the left boundary. Similarly, we can obtain line C_rD_r as the right boundary.

The target area which is constrained by the ranges of depth and width becomes a tetragon $C_rC_lD_lD_r$. Figure 6(a) is an example of target area estimation.

3.3 Determination of Prediction Window

For a *lost* camera, we estimate the target position in the image using the target position on the floor which is estimated by *tracking* cameras.

When the target position is estimated by the method of section 3.1, the human model at the target position is projected to the image of a *lost* camera by equation (1). The projected region is considered as the prediction window, which is searched for the target.

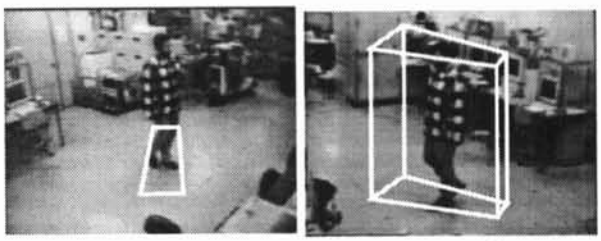
camera A camera B

Figure 6: Target area and its projection

When the target area is estimated by the method of section 3.2, the cylinder at the every possible position inside the target area is projected to the image of the *lost* camera. Then the union of all projected region is considered as the prediction window. Figure 6(b) shows the projected region for camera B which is generated from the target area estimated by camera A.

3.4 Target Identification by Target Velocity

A *lost* camera searches the prediction window for the target. However, if there is another moving object in this window, the *lost* camera may track him as the target. To discriminate moving objects, we use the target velocity.

3.4.1 Estimation of Target Velocity

The velocity in an image is derived by differentiating equation (1):

$$\boldsymbol{v} = -\frac{\dot{h}}{h^2}\boldsymbol{CX} + \frac{1}{h}\boldsymbol{CV} \tag{7}$$

where $\boldsymbol{V} = \dot{\boldsymbol{X}} = [U, V, W]^t$ denote a three dimensional velocity, $\boldsymbol{v} = \dot{\boldsymbol{x}} = [u, v]^t$ denote a velocity in the image.

When there are at least two *tracking* cameras, a set of the constraint equations about $\boldsymbol{V}$ is derived by substituting each mean flow which is estimated by *tracking* cameras for $\boldsymbol{v}$ at equation (7). The target velocity $\boldsymbol{V}$ is calculated by applying the least squares method to the equations.

When there is only one *tracking* camera, the target velocity $\boldsymbol{V}$ cannot be calculated by the above method. In this case, by assuming that the target have no vertical velocity, $\boldsymbol{V} = [U, V, 0]$ is calculated by solving equation (7).

3.4.2 Estimation of Target Velocity in Image

For *lost* cameras, the target velocity $\hat{\boldsymbol{v}}$ in the image is estimated from $\boldsymbol{V}$ using equation (7). Each *lost*

camera searches for flow vectors which are similar to $\hat{v}$. If another object is moving in the prediction window at a different velocity, the camera can discriminate it from the target.

Figure 7 is an example of effective use of target velocity. In this figure, "+" shows the center of gravity of the object region. A solid window shows a tracking window, and a dashed window shows a prediction window in a *lost* camera.

In camera C, the target is occluded by an obstacle. However, an appropriate position is searched by using the projected area which is estimated by other cameras. At the same time, there is another person with a leftward velocity in the image. Camera C correctly judges that the object is not the target because the projected velocity $\hat{V}_c$ is rightward.

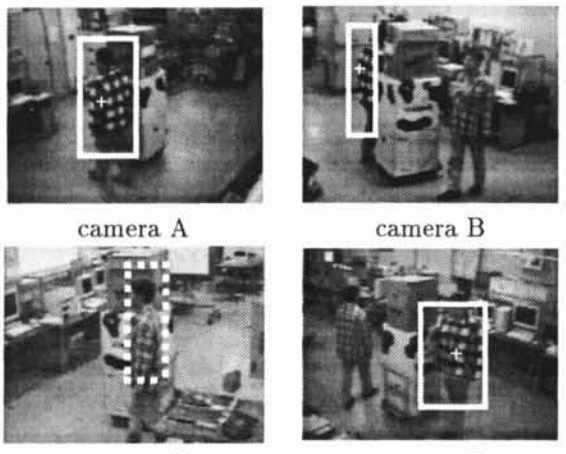

Figure 7: Tracking result

4 Experiments

Figure 8 shows a sequence of views from camera A. At the 50th frame and the 230th frame, camera A could not observe the target. However, the system did not lose the target by using images from other cameras. Figure 9 shows the calculated trajectory of the target on the floor.

We assume that the horizontal position of the center of gravity of the object region is reliable. However, if the width of the object region is not completely extracted (for example, a half of the target is occluded), the target position becomes less reliable.

5 Conclusion and Future Work

We proposed a method of optical flow-based person tracking by multiple cameras. The experiments show that a person can be tracked in a cluttered environment even if objects may obstruct the camera view.

One future work is to implement this method in a realtime image processor. Another future work is to extend the method so that multiple persons are tracked at the same time.

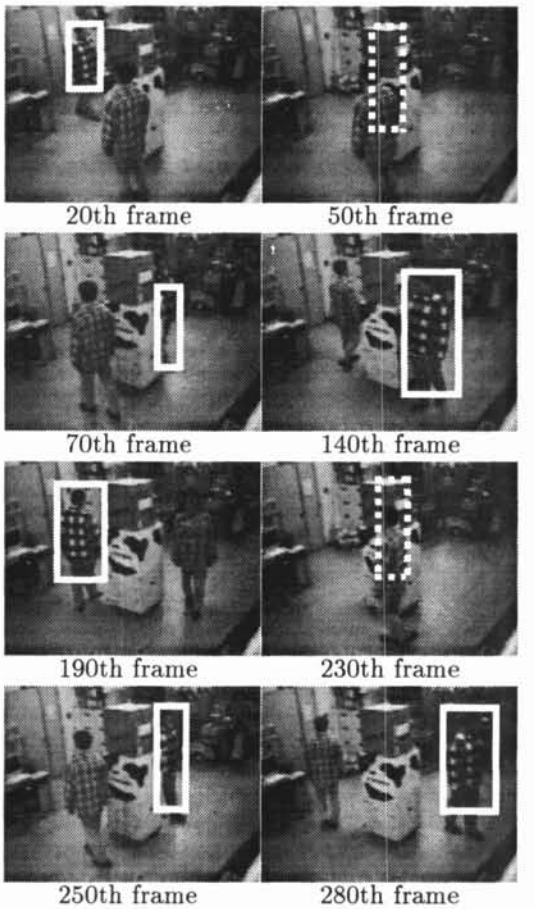

Figure 8: Tracking sequence of Camera A

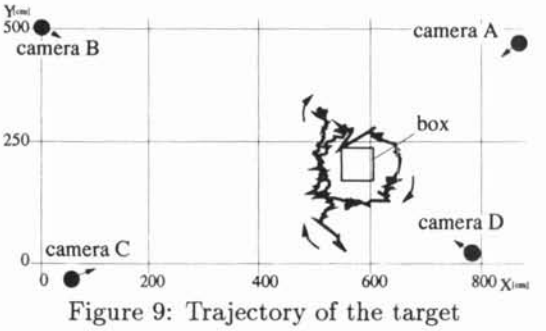

Figure 9: Trajectory of the target

References

[1] H. Chen et al.: Detecting Multiple Rigid Image Motions from an Optical Flow Field Obtained with Multi-Scale, Multi-Orientation Filters. IEICE Trans. Vol. E76-D No. 10, pp. 1253-1262 (1993).

[2] S. Yamamoto et al.: Realtime Multiple Object Tracking Based on Optical Flows, Proc. Int. Conf. on Robotics and Automation, pp. 2328-2333 (1995).

[3] B. S. Rao and H. Durrant-Whyte: A Decentralized Bayesian Algorithm for Identification of Tracked Targets, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 6, (1993).

[4] A. Utsumi et al.: Multiple-Human Tracking using Multiple Cameras. Proc. Third IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp. 498-503 (1998).