Proc. ICRA-2009 Workshop on Person Detection and Tracking,
Kobe, Japan, May 2009.

Robust Stereo-Based Person Detection and Tracking for a Person Following Robot

Junji Satake and Jun Miura
Department of Information and Computer Sciences
Toyohashi University of Technology

***Abstract*— This paper describes a stereo-based person detection and tracking method for a mobile robot that can follow a specific person in dynamic environments. Many previous works on person detection use laser range finders which can provide very accurate range measurements. Stereo-based systems have also been popular, but most of them have not been used for controlling a real robot. We propose a detection method using depth templates of person shape applied to a dense depth image. We also develop an SVM-based verifier for eliminating false positive. For person tracking by a mobile platform, we formulate the tracking problem using the Extended Kalman filter. The robot continuously estimates the position and the velocity of persons in the robot local coordinates, which are then used for appropriately controlling the robot motion. Although our approach is relatively simple, our robot can robustly follow a specific person while recognizing the target and other persons with occasional occlusions.**

***Index Terms*— Person detection and tracking, Mobile robot, Stereo.**

I. Introduction

Following a specific person is an important task for service robots. Visual person following in public spaces entails tracking of multiple persons by a moving camera.

There have been a lot of works on person detection and tracking using various image features and classification methods [1], [2], [3], [4], [5]. Many of them, however, use a fixed camera. In the case of using a moving camera, foreground/background separation is an important problem.

This paper deals with detection and tracking of multiple persons for a mobile robot. Laser range finders are widely used for person detection and tracking by mobile robots [6], [7], [8]. Image information such as color and texture is, however, sometimes necessary for person segmentation and/or identification. Omnidirectional cameras are also used [9], [10], but their limited resolutions are sometimes inappropriate for analyzing complex scenes.

Stereo is also popular in moving object detection and tracking. Beymer and Konolige [11] developed a method of tracking people by continuously detecting people using distance information obtained from a stationary stereo camera.

Howard et al. [12] proposed a person detection method which first converts a depth map into a polar-perspective map on the ground and then extracts regions with largely-accumulated pixels. Calisi et al. [13] developed a robot system that can follow a moving person. It makes an appearance model for each person using stereo in advance. In tracking, the robot extracts candidate regions using the model and confirms it using stereo. Occlusions between people are not handled in these works.

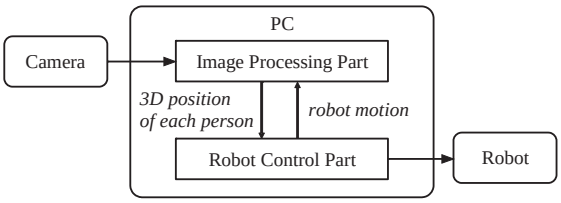


Fig. 1. Configuration of our system.

Ess et al. [14], [15] proposed to integrate various cues such as appearance-based object detection, depth estimation, visual odometry, and ground plane detection using a graphical model for pedestrian detection. Although their method exhibits a nice performance for complicated scenes, it is still costly to be used for controlling a real robot.

In this paper, we propose a person tracking method using stereo. We prepare several *depth templates* to be used for dense depth images and detect person regions by template matching, followed by a support vector machine (SVM)-based verifier. Depth information is very effective in data association with adjusting template size and values as well as occlusion handling. Person detection results are input to Extended Kalman Filter-based trackers. The robot continuously estimates the position and the velocity of persons in the robot local coordinates to appropriately control its motion. Fig. 1 shows the configuration of our system. The main contribution of the paper is to show that a simple depth template-based approach, combined with EKF and an SVM-based verifier, realizes a robust person following by a mobile robot.

II. Stereo-based person detection and tracking

To track persons stably with a moving camera, we use *depth templates*, which are the templates for human upper body in depth images (see Fig. 2); we currently use three templates with different direction of body. We made the templates from the depth images where the target person was at $2\,[m]$ away from the camera. A depth template is a binary template, the foreground and the background value are adjusted according to status of tracks and input data.

A. Tracking

For a person being tracked, his/her predicted scene position is available from the corresponding EKF-tracker (see

Fig. 2. Depth templates.

Sec. III-B). We thus set the foreground depth of the template to the predicted depth of the head of the person and search a region around the predicted image position for the person.

Concerning the background depth, since it may change as the camera moves, we estimate it on-line. We make the depth histogram of the current input depth image and use the Kth percentile as the background depth (currently, $K = 90$).

For a depth template $T(x, y)$ of $H \times W$ pixels ($x \in [-W/2, W/2]$, $y \in [-H/2, H/2]$) and the depth image $I_D(x, y)$, the 2D image position (x^*, y^*) is given as the position which minimizes the following SSD (sum of squared distances) criterion:

$$\sum_{p=-W/2}^{W/2} \sum_{q=-H/2}^{H/2} \left[T(p, q) - I_D(x + p, y + q)\right]^2 . \quad (1)$$

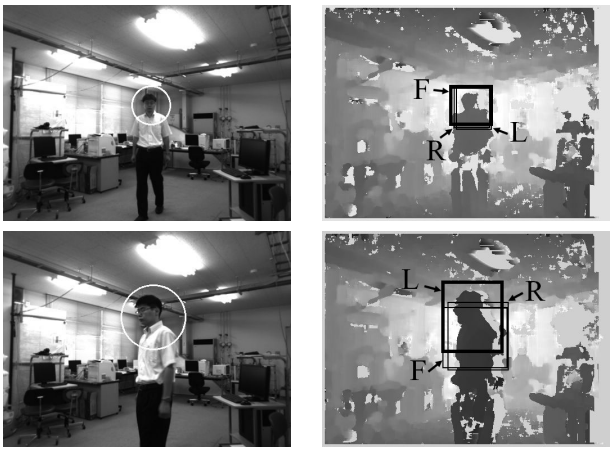
We use the three templates simultaneously and take the one with the smallest SSD value as the detection result if that value is less than some threshold.

Each template has the *position of the head* and the median value of the neighboring region of that position is used as the depth from the camera of the detected person. The accuracy of the depth value is empirically estimated as about one percent when a person is at about $3\,[m]$ distance.

B. Detection

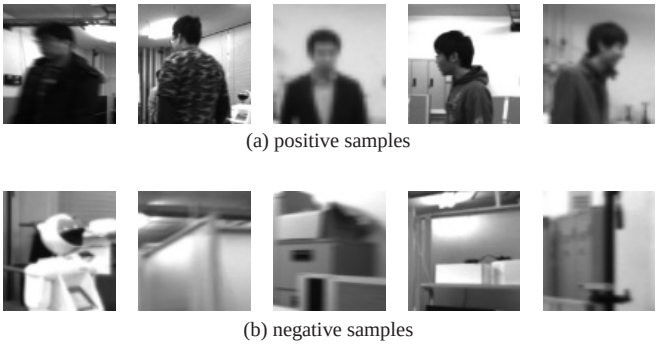
We continuously check if a new person appears in the image. In this case, we do not have any prediction and basically search the entire image. The foreground depth is set to the depth of each image position and the background one is set as in the same way as tracking.

We use the same SSD criterion (see eq. (1)) for judging if a person exists at an image position. Since applying this SSD calculation to the entire image is costly, we examine the three boundary points, on the left of, on the right of, and above the head position, and only when the depth values of the points are one-meter farther than the depth of the head, the SSD value is calculated and evaluated. We also set a *detection volume* to search in the scene; its height range is $0.7 \sim 2.0\,[m]$ and the range of the depth from the camera is $0.5 \sim 5.5\,[m]$. In addition, if the image position under consideration is in an already-detected person region, and unless the its depth is at least one-meter smaller than the depth of the region, the detection there is skipped. These techniques can reduce the search cost largely. After collecting pixels with qualified SSD values, we extract the mass centers of all connected regions as the positions of newly detected persons.



(a) Input images (b) Depth images

Fig. 3. Detection examples using depth templates.



(a) positive samples

(b) negative samples

Fig. 4. Training samples for the SVM-based verifier.

Figure 3 shows examples of detection using the depth templates. Three rectangles in each depth image are detection results with the three templates, and the one with the highest evaluation value is shown in bold line. Even when the direction of the body changed, it is possible to detect a person stably by using multiple templates.

C. Intensity-based false detection elimination

A simple template-based detection is effective in reducing the computational cost but at the same time may produce many false detections for objects with similar silhouette to person. To cope with this, we use an SVM-based person verifier using intensity images.

We collected many person candidate images detected by the depth templates, and manually examined if they are correct. Fig. 4 shows some of positive and negative samples. We used 438 positive and 146 negative images for training. The size of the sample images is normalized to 20×20. The SVM is the one with RBF kernel ($K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp(-\gamma||\boldsymbol{x}_1 - \boldsymbol{x}_2||^2)$, $\gamma = 8.0$). We use an OpenCV implementation of SVM.

We examine the performance of the SVM-based verifier using three image sequences, which had not used for training. The numbers of persons appearing in the sequences are zero, one, and two, respectively. We used the image regions

TABLE I

PERFORMANCE SUMMARY OF THE SVM-BASED VERIFIER.

# of persons	results		
0		judged to exist	judged not to exist
	exist	—	—
	not exist	0	126
1		judged to exist	judged not to exist
	exist	414	5
	not exist	0	75
2		judged to exist	judged not to exist
	exist	391	31
	not exist	0	491

detected using the depth templates. Table I summarizes the results. It is noted that the rate of eliminating false positives is 100%. This is very important because a simple depth template-based person detection tends to produce many false positives. On the other hand, the verifier sometimes eliminates actual person regions; the false negative rate is about six percent. The EKF-based tracker can usually cope with such an occasional failure of person detection.

III. PERSON TRACKING AND ROBOT CONTROL

A. Configuration of our system

Figure 5 illustrates the coordinate systems attached to our mobile robot and stereo system. The relation between the robot and the camera coordinate system is given by

$$Z_c \left[\begin{array}{cccc} x & y & 1 \end{array} \right]^T = \boldsymbol{A} \left[\begin{array}{c|c} \boldsymbol{R} & \boldsymbol{T} \end{array} \right] \left[\begin{array}{cccc} X_r & Y_r & Z_r & 1 \end{array} \right]^T, \quad (2)$$

where $\boldsymbol{A}$, $\boldsymbol{R}$, and $\boldsymbol{T}$ show the intrinsic parameters matrix, the rotation matrix, and the translation vector, respectively.

B. Estimation of 3D position using EKF

1) State equation: In the robot coordinate system, the person's position at time t is defined as (X_t, Y_t, Z_t). The state variable $\boldsymbol{x}_t$ is defined as

$$\boldsymbol{x}_t = \left[\begin{array}{ccccc} X_t & Y_t & Z_t & \dot{X}_t & \dot{Y}_t \end{array} \right]^T,$$

where $\dot{X}_t$ and $\dot{Y}_t$ denote velocities in the horizontal plane.

We first consider the case where the robot does not move. The system equation is given by

$$\boldsymbol{x}_{t+1} = \boldsymbol{F}_t \boldsymbol{x}_t + \boldsymbol{G}_t \boldsymbol{w}_t \quad (3)$$

where $\boldsymbol{w}_t$ is the process noise and

$$\boldsymbol{F}_t = \left[\begin{array}{ccccc} 1 & 0 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array} \right], \quad \boldsymbol{G}_t = \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \Delta t & 0 \\ 0 & \Delta t \end{array} \right],$$

$$\boldsymbol{Q}_t = Cov(\boldsymbol{w}_t) = E\left[\boldsymbol{w}_t \boldsymbol{w}_t^T\right] = \sigma_w^2 \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right].$$

We then consider the case where the robot moves. Figure 6 shows how a wheeled mobile robot moves. The distance of two wheels is denoted as $2d$. When each wheel rotates with

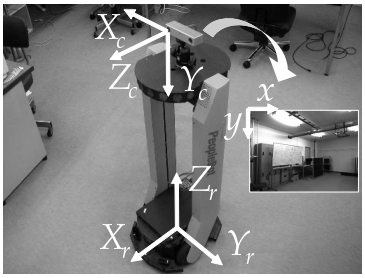


Fig. 5. Definition of coordinate systems.

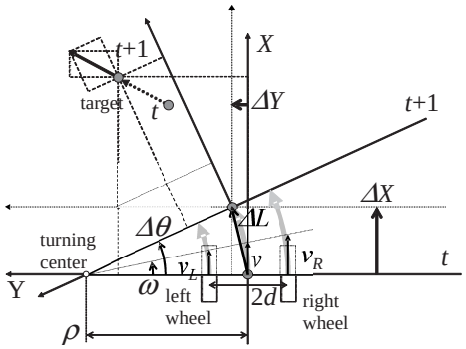


Fig. 6. Control of wheeled mobile robot.

speed v_L and v_R, the velocity v, the angular velocity ω, and the turning radius ρ of the robot have the following relations:

$$v = (v_R + v_L)/2, \quad \omega = (v_R - v_L)/2d,$$
$$\rho = d(v_R + v_L)/(v_R - v_L).$$

The rotation angle $\Delta\theta$ and the moved distance ΔL during time Δt are obtained respectively as

$$\Delta\theta = \omega \Delta t, \quad \Delta L = 2\rho \sin(\Delta\theta/2).$$

In addition, the robot movement ΔX and ΔY seen from the robot position at time t are obtained respectively as

$$\Delta X = \Delta L \cos(\Delta\theta/2), \quad \Delta Y = \Delta L \sin(\Delta\theta/2).$$

We then have the relationship between the position and the velocity of a person before and after the coordinate transformation from the robot coordinate at time t to that at time $t+1$ as follows:

$$X^{(t+1)} = (X^{(t)} - \Delta X)\cos\Delta\theta + (Y^{(t)} - \Delta Y)\sin\Delta\theta,$$
$$Y^{(t+1)} = -(X^{(t)} - \Delta X)\sin\Delta\theta + (Y^{(t)} - \Delta Y)\cos\Delta\theta,$$
$$\dot{X}^{(t+1)} = \dot{X}^{(t)}\cos\Delta\theta + \dot{Y}^{(t)}\sin\Delta\theta - v,$$
$$\dot{Y}^{(t+1)} = -\dot{X}^{(t)}\sin\Delta\theta + \dot{Y}^{(t)}\cos\Delta\theta.$$

By the combination of these equations and eq. (3), the state equation that considers the robot movement $\boldsymbol{u}_t = [v_L \ v_R]^T$ is expressed as

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}_t(\boldsymbol{x}_t, \boldsymbol{u}_t) + \boldsymbol{G}_t \boldsymbol{w}_t, \quad (4)$$

where

$$\boldsymbol{f}_t(\boldsymbol{x}_t, \boldsymbol{u}_t) = \begin{bmatrix} (X_t + \Delta t\dot{X}_t - \Delta X)\cos\Delta\theta + (Y_t + \Delta t\dot{Y}_t - \Delta Y)\sin\Delta\theta \\ -(X_t + \Delta t\dot{X}_t - \Delta X)\sin\Delta\theta + (Y_t + \Delta t\dot{Y}_t - \Delta Y)\cos\Delta\theta \\ Z_t \\ \dot{X}_t\cos\Delta\theta + \dot{Y}_t\sin\Delta\theta - v \\ -\dot{X}_t\sin\Delta\theta + \dot{Y}_t\cos\Delta\theta \end{bmatrix}.$$

2) Observation equation: The observed person's position in the robot coordinate system is denoted as $\boldsymbol{y}_t$. The observation equation is expressed as

$$\boldsymbol{y}_t = \boldsymbol{H}_t\boldsymbol{x}_t + \boldsymbol{v}_t, \quad (5)$$

where $\boldsymbol{v}_t$ is the observation noise and

$$\boldsymbol{y}_t = \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix}, \quad \boldsymbol{H}_t = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{R}_t = Cov(\boldsymbol{v}_t) = E\left[\boldsymbol{v}_t\boldsymbol{v}_t^T\right] = \sigma_v^2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

3) Extended Kalman filter: The Extended Kalman filter (EKF) are formulated using the the state eq. (4) and the observation equation (5). The EKF can estimate the position and the velocity of a person with their uncertainty estimates.

C. Data association and occlusion handling

3D position information is effective in data association. We use the predicted 3D position to adjust the size and the foreground depth of the depth templates to be used (see Section. II-A). If a person is detected, then its 3D position is tested with the Mahalanobis distance to see if the matching can be made between the detected person and the corresponding track.

3D information is also used for occlusion handling. In the case where an occlusion relation is reliably predicted between two persons, if an occluding one is correctly detected, only the prediction step in EKF is performed for the occluded person. Possible occlusion relationships are enumerated by examining the predicted 3D positions of tracks.

In an ordinary situation, persons pass each other with keeping a certain distance (say, one meter) between them. In our current setting, this distance difference can be detected as long as they are within about four meters from the camera; this is enough for the robot to correctly recognize the person motion in a local region around the robot.

Figure 7 shows an example of correctly tracking two persons under occlusion and depth change. In the middle row of the image, the person behind is completely occluded and only the prediction step in EKF is performed. After the occlusion, the track continues correctly.

D. Tracking algorithm

The image processing part (see Fig. 1) works as follows:
1) Stereo processing: The depth image is made with a stereo camera.
2) Person tracking: Each person is tracked by using the EKF described in Section III-B.

Fig. 7. Correctly tracking two persons in an occlusion case.

2.1) Prediction: The 3D position and its uncertainty at the current time t are predicted from the state variable at the previous time $t-1$. They are then projected to 2D image by eq. (2). The projected uncertainty region is used for determining the predicted region.
2.2) Observation: The predicted region is searched for the person by the method described in Section II-A. The templates used for search are made based on the depth to the person. After the search, the person's 3D position $\boldsymbol{y}_t$ is calculated by eq. (2) based on image coordinates (x, y) and distance from camera $Z_c = D$.
2.3) Data association: Correspondences are made between tracks and observations by the procedure described in Section III-C.
2.4) Update: The state variable is updated, if an observation is obtained.
3) Detection: The persons who appear newly in image are detected with depth templates.
4) Communication: The estimated position is sent to the robot control part, and the rotational speeds of the left and right wheels are received.

IV. Control to follow a specific person

The robot with two-wheel drive follows a circular trajectory from the current to the target position (path A in Fig. 8). In this case, the speeds for the wheels to move the robot at velocity v is calculated as follows. From the equation:

$$\rho^2 = \left\{(X/2)^2 + (Y/2)^2\right\} + \left\{(X/2)^2 + (\rho - Y/2)^2\right\},$$

we have

$$\rho = (X^2 + Y^2)/2Y.$$

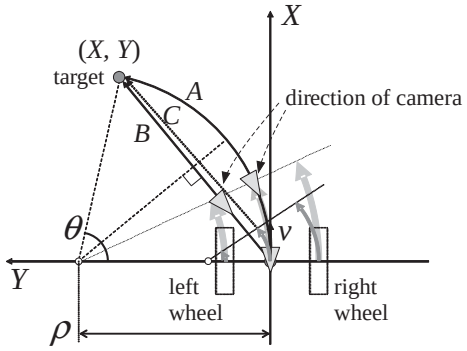


Fig. 8. Path to target position.

Then we can calculate the velocities as:

$$\begin{aligned} v_L &= v\left(1-\tfrac{d}{\rho}\right) = v\left(1-\tfrac{2dY}{X^2+Y^2}\right), \\ v_R &= v\left(1+\tfrac{d}{\rho}\right) = v\left(1+\tfrac{2dY}{X^2+Y^2}\right). \end{aligned}$$

When the robot follows this circular path, however, since the turning rate of robot orientation is relatively slow, the target person tends to go out of the field of view. On the other hand, the robot first turns and then moves straight toward the target like path B, the robot movement is not smooth. We thus use the one like path C, on which the robot turns to the target while moving ahead. In this case, the velocity of each wheel is adjusted as follows:

$$v_L = v\left(1-k\tfrac{2dY}{X^2+Y^2}\right), \quad v_R = v\left(1+k\tfrac{2dY}{X^2+Y^2}\right).$$

This means the turning radius ρ is reduced to ρ/k.

V. Experimental result

A. Experimental setup

We have implemented the proposed method on a PeopleBot (by Mobile Robots) with a Bumblebee2 stereo camera (by PointGrey Research) for the experiments (see Fig. 5). A note PC (Core2Duo, 2.6GHz) performs all processes including stereo calculation, person detection and tracking, and robot motion control. The processed image size is 512×384 and the processing time is about $90\,[msec/frame]$. Table II shows the breakdown of processing time; our system can process about eleven frames per second.

We implemented the software modules for person detection and tracking, motion planning, and robot control as *RT components* in the *RT-middleware* environment [16] for easier development and maintenance.

TABLE II
Breakdown of processing time.

Processing	Time
1) Image acquisition & stereo processing	40 [ms]
2) Person tracking (in the case of two persons)	20 [ms]
3) Person detection	10 [ms]
4) Communication, display, and save data	20 [ms]
Total	90 [ms]

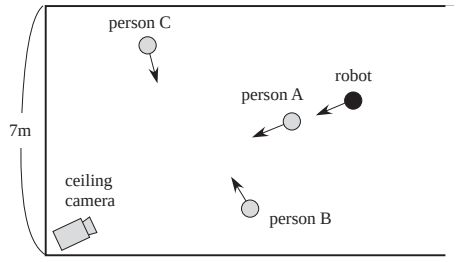


Fig. 10. Initial positions of the robot and the persons.

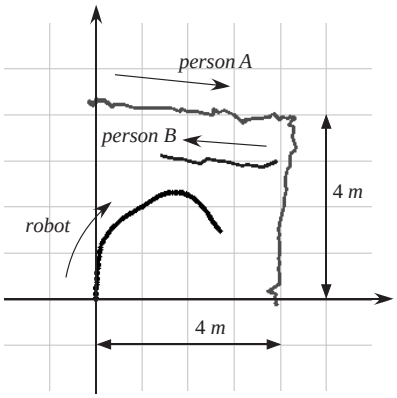


Fig. 11. Trace of two persons and the robot.

B. Person following experiments

Figure 9 shows a result of tracking. The left row images are the results of person detection. Each circle in the image shows the result of observation with depth templates, and each small point shows the 3D head position estimated using EKF. The right row images show the positions of the robot and the persons taken by a ceiling camera. In addition, the curves in the final frame (#156) shows the traces of the robot and the persons.

Figure 10 shows the initial positions of the robot and the persons. The robot moved toward person A who was detected first and considered the target. Even when person B and C passed between the robot and person A, the target person was correctly tracked.

C. Evaluation of person position estimation

We evaluated the quality of the person position estimation. Figure 11 shows the traces of the robot and two persons in the robot initial coordinates. Person A moved on two edges of a $4 \times 4\,[m]$ square drawn on the floor. Person B moved so that it temporarily occluded person A.

The robot followed person A while estimating the positions of the persons. The averaged and the maximum error in position estimation for person A were $125\,[mm]$ and $336\,[mm]$, respectively. This result shows that the position estimation is accurate enough for the robot to follow a specific person.

Fig. 9. Experimental result with one person to follow and the other two.

VI. Conclusions and Future Work

This paper has described a method of detecting and tracking multiple persons for a mobile robot by using distance information obtained by stereo. We presented an EKF-based formulation by which the robot continuously estimates the position and the velocity of persons. Distance information is effectively utilized for robust person detection, data association, and occlusion handling. We realized a robot that can robustly follow a specific person while recognizing the target and other persons with occasional occlusions.

The current algorithm does not consider the case where multiple persons are too close to be separated by depth information. To cope with such cases, it would be necessary to use other visual information such as color and texture. It is also necessary to manage static obstacles such as furniture as well as an effective path planning to realize a person following robot that can operate in more complex environments.

Acknowledgment

The authors would like to thank Yuki Ishikawa for his help in implementing the system. This work is supported by NEDO (New Energy and Industrial Technology Development Organization, Japan) Intelligent RT Software Project.

References

[1] P. Viola, M.J. Jones, and D. Snow. Detecting Pedestrians Using Patterns of Motion and Appearance. *Int. J. of Computer Vision*, Vol. 63, No. 2, pp. 153–161, 2005.

[2] N. Dalal and B. Briggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of 2005 IEEE Conf. on Computer Vision and Patttern Recognition*, pp. 886–893, 2005.

[3] B. Han, S.W. Joo, and L.S. Davis. Probabilistic Fusion Tracking Using Mixture Kernel-Based Bayesian Filtering. In *Proceedings of the 11th Int. Conf. on Computer Vision*, 2007.

[4] D.M. Gavrila. A Bayesian, Exempler-Based Approach to Hierarchical Shape Matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 8, pp. 1408–1421, 2008.

[5] S. Munder, C. Schnorr, and D.M. Gavrila. Pedestrian Detection and Tracking Using a Mixture of View-Based Shape-Texture Models. *IEEE Trans. on Intelligent Transportation Systems*, Vol. 9, No. 2, pp. 333–343, 2008.

[6] C.-Y. Lee, H. Gonzàlez-Baños, and J.-C. Latombe. Real-Time Tracking of an Unpredictable Target Amidst Unknown Obstacles. In *Proceedings of the 7th Int. Conf. on Control, Automation, Robotics and Vision*, pp. 596–601, 2002.

[7] D. Schulz, W. Burgard, D. Fox, and A.B. Cremers. People Tracking with a Mobile Robot Using Sample-Based Joint Probabilistic Data Association Filters. *Int. J. of Robotics Research*, Vol. 22, No. 2, pp. 99–116, 2003.

[8] N. Bellotto and H. Hu. Multisensor Data Fusion for Joint People Tracking and Identification with a Service Robot. In *Proceedings of 2007 IEEE Int. Conf. on Robotics and Biomimetics*, pp. 1494–1499, 2007.

[9] H. Koyasu, J. Miura, and Y. Shirai. Realtime Omnidirectional Stereo for Obstacle Detection and Tracking in Dynamic Environments. In *Proceedings of the 2001 IEEE/RSJ Int. Conf. on Intelligent Robots and Sysetms*, pp. 31–36, 2001.

[10] M. Kobilarov, G. Sukhatme, J. Hyams, and P. Batavia. People Tracking and Following with Mobile Robot Using Omnidirectional Camera and a Laser. In *Proceedings of 2006 IEEE Int. Conf. on Robotics and Automation*, pp. 557–562, 2006.

[11] D. Beymer and K. Konolige. Real-Time Tracking of Multiple People Using Continuous Detection. In *Proceedings of the 7th Int. Conf. on Computer Vision*, 1999.

[12] A. Howard, L.H. Matthies, A. Huertas, M. Bajracharya, and A. Rankin. Detecting Pedestrians with Stereo Vision: Safe Operation of Autonomous Ground Vehicles in Dynamic Environments. In *Proceedings of the 13th Int. Symp. of Robotics Research*, 2007.

[13] D. Calisi, L. Locchi, and R. Leone. Person Following through Appearance Models and Stereo Vision using a Mobile Robot. In *Proceedings of VISAPP-2007 Workshop on Robot Vision*, pp. 46–56, 2007.

[14] A. Ess, B. Leibe, and L.V. Gool. Depth and Appearance for Mobile Scene Analysis. In *Proceedings of the 11th Int. Conf. on Computer Vision*, 2007.

[15] A. Ess, B. Leibe, K. Schindler, and L.V. Gool. A Mobile Vision System for Robust Multi-Person Tracking. In *Proceedings of the 2008 IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[16] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku, and W.-K. Yoon. RT-Middleware: Distributed Component Middleware for RT (Robot Technology). In *Proceedings of 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3555–3560, 2005.