Target Tracking and Posture Estimation of 3D Objects by Using Particle Filter and Parametric Eigenspace Method

M. Obata
YASKAWA INFORMATION SYSTEMS Corporation
Kitakyushu, Fukuoka, Japan

T. Nishida
Dept. of Mechanical and Control Engineering
Kyushu Institute of Technology
Kitakyushu, Fukuoka, Japan

H. Miyagawa
YASKAWA INFORMATION SYSTEMS Corporation
Kitakyushu, Fukuoka, Japan

F. Ohkawa
Dept. of Systems Innovation and Informatics
Kyushu Institute of Technology
Iizuka, Fukuoka, Japan

Abstract ***In this research, we propose a method executing the particle filter (PF) and the parametric eigenspace method (PEM) simultaneously. Namely, the PEM is used as a re-sampling method of the PF. Since the PEM is executed at the same frequency as the number of particles, high-speed execution of the proposed method is possible. Moreover, since the posture of the object can be estimated by using PEM, the position of the moving target object at the next time frame can be estimated in high accuracy. We apply the method to the real vision image, and examine the effectiveness and the performance.***

Keywords: particle filter, parametric eigenspace method, object detection, target tracking, posture estimation.

1 Introduction

Object tracking is required by many vision applications such as industrial automation, human interface, robot navigation, and so on. The particle filter (PF) framework is known to be effective for the target tracking problems, and it provide a robust tracking framework as they are neither limited to linear systems nor require the noise to be Gaussian. The technique for the real time object tracking on the image by the PF is variously proposed [1]-[7]. However, most of existing methods adopt only simple perceptual cues such as color histogram or contour similarity for hypothesis evaluation [1]-[3]. Moreover, concretely, the problem of the PF is that accuracy decreases when the number of particles is limited to small number. In a usual PF, the number of particles needed to guarantee approximate accuracy of probability density function increases exponential when the dimension of the state vector increases. Therefore, when the number of particles is limited to extent in which the real time tracking can be executed, a phenomenon in which density of particle is insufficient versus dimension of vector and a serious accuracy decrease are caused. Thus, when the PF is applied, the idea of decreasing the number of particles without an accuracy decrease is necessary. Furthermore, to improve the robustness and accuracy of tracking with more sophisticated hypothesis, we must devise methods for the pertinent evaluation. The importance sampling and the partitioned sampling, etc. have been proposed as such a method, and high approximate accuracy can be achieved without greatly increasing the calculation cost by using these methods [4]-[7].

On the other hand, the parametric eigenspace method [8] (PEM) has been widely used as a target detection method in actual environment. Namely, the detection and the posture estimation of a three-dimensional object can be done by the PEM onl-ine with one camera by using the dictionary database. However, the relation of the volume of the dictionary database (i.e. the recognition accuracy) and the processing speed involves trade-off. Therefore, a careful design of various parameters is necessary for the construction of the recognition processing system with the PEM.

Thus, in this research, we propose the method executing the PEM and the PF simultaneously, namely, the PEM is used as a re-sampling method of the PF. In this method, the object detection, the posture estimation, and the target tracking can be done in high-speed by executing the PEM and the PF simultaneously.

As related methods, there are methods such as combined the cascaded classifiers based on AdaBoost [9], the optical flow [10], and so on. However, there is no research that investigates the method for executing the PF together with the PEM.

In the next section, we describe about PEM, and in Sec.3, the target tracking method using PF is shown. Moreover in Sec.4, we describe the methodology of the target tracking using the information of the dynamics estimated by PEM. In Sec.5, we show several experimental results that show the effectiveness of proposed method, and we conclude in Sec.6.

2 Object Detection by Parametric Eigenspace Method

The object detection procedure using PEM can be divided by two phases, the first is dictionary database generation phase and the second is detection phase. Here, each phase is described briefly.

2.1 Dictionary database

Let $q_n(\boldsymbol{x}) \triangleq q_n(x,y)$, $N_t = \{n | n = 1, \cdots, |N_t|\}$ be a set of template images consisting of $M \times N$ pixels. Moreover, let $\boldsymbol{x}_n$ be its representation of MN dimensional vector in scan line order i.e.,

$$\boldsymbol{x}_n = (q_{11}, q_{12}, \cdots, q_{MN})^T \in \mathbb{R}^{MN}, \tag{1}$$

where the vectors are normalized as $\|\boldsymbol{x}_n\| = 1$. Next, the covariance matrix is generated as follows,

$$\boldsymbol{S} \triangleq E\left\{(\boldsymbol{x}_n - \boldsymbol{m})(\boldsymbol{x}_n - \boldsymbol{m})^T\right\} \in \mathbb{R}^{MN \times MN} \tag{2}$$

where $\boldsymbol{m} \in \mathbb{R}^{MN}$ is the mean vector of $\boldsymbol{x}_n$. The eigenvectors $\boldsymbol{u}_i$ $(i = 1, \cdots, MN)$ corresponding to the eigenvalue λ_i $(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{MN})$ given by the principal component analysis of $\boldsymbol{S}$. Let k is the dimension of the sub space to be used, and we register the following matrix consists of the eigenvectors $\boldsymbol{u}_k$ as a dictionary database,

$$\boldsymbol{U} \triangleq (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_k) \in \mathbb{R}^{MN \times k}. \tag{3}$$

Therefore, an input image vector $\boldsymbol{x}_n$ is mapped to the point $\boldsymbol{y}_n$ in the k-dimensional eigenspace as follows,

$$\boldsymbol{y}_n = \boldsymbol{U}^T \boldsymbol{x}_n \in \mathbb{R}^k, \tag{4}$$

and the manifold $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_n \in \mathbb{R}^k\}$. For example, 36 template images of target object rotated in units of ten degrees were prepared and these mapped into an eigenspace according to Eq.(4). This phase is finished by registering the manifold $\boldsymbol{y}$ as a dictionary database obtained by mapping the template image.

2.2 Object detection and parameter estimation

Next, the preprocessing is executed for an input image such as normalize, gray-scale conversion, and so on. The range of around the target object is extracted by the preprocessing, and the input vector $\boldsymbol{x}(t) \in \mathbb{R}^{MN}$ is constructed. Then, the input vector $\boldsymbol{x}(t)$ is mapped into the k-dimensional eigenspace, and the index of the nearest point $\boldsymbol{y}_{n_z}$ by the following expressions,

$$n_z = \underset{n \in N_t}{\mathrm{argmin}} \|\boldsymbol{U}^T \boldsymbol{x}(t) - \boldsymbol{y}_n\|. \tag{5}$$

Thus, it can be said that the parameters of the target object is most similar to that of the template image with the index n_z. However, the computational cost for the detection and the posture estimation of a fast movement target object by the PEM is comparatively high.

3 Target Tracking by Particle Filtering

The procedure flow chart of the proposed method is shown in Fig.1. First, in the prediction part, the position of the target object can be predicted from the position of the target in the frame at time t by following equations,

$$\boldsymbol{s}^{(m)}(t) \triangleq \left(s_x^{(m)}(t),\ s_y^{(m)}(t)\right)^T \tag{6}$$

$$s_x^{(m)}(t) = s_x^{(m)}(t-1) + r_x + \Delta_x, \tag{7}$$
$$s_y^{(m)}(t) = s_y^{(m)}(t-1) + r_y + \Delta_y \tag{8}$$

where $\boldsymbol{s}^{(m)}(t)$ represents the position of mth particle, $r_x = [-|r_x|, |r_x|]$ and $r_y = [-|r_y|, |r_y|]$ are random numbers, and $M_t = \{m | m = 1, \cdots, |M_t|\}$

is a set of indices of particle. Moreover, Δ_x and Δ_y are the terms to increase the robustness of tracking calculated by estimating the moved distance of the target, i.e., the amount of the movement between frames are used for them. In the observation part, PEM is executed at the position of the particle estimated by Eq.(8). Namely, $M \times N$ pixels surrounding the particles are extracted, the input vectors $\boldsymbol{x}^{(m)}(t) \in \mathbb{R}^{MN}$ are generated, and the object detection is executed by Eq.(5). Here, at first frame, PEM is executed to whole of the image and the particle positions are initialized by

$$\boldsymbol{s}^{(m)}(0) = \underset{n \in N_t, \boldsymbol{x}_l \in F_0}{\operatorname{argmin}} \|\boldsymbol{U}^T \boldsymbol{x}_l - \boldsymbol{y}_n\|. \tag{9}$$

where F_0 represents the first frame and $\boldsymbol{x}_l$ is the input vector extracted from the frame. Next, the target object detection is executed by

$$d_{\min}^{(m)} = \min_{n \in N_t, m \in M_t} \|\boldsymbol{U}^T \boldsymbol{x}^{(m)}(t) - \boldsymbol{y}_n\|. \tag{10}$$

Thus, the object detection and the target tracking are executed by following steps.

Step 1 Likelihood of particles are calculated by using $d_{\min}^{(m)}$ as follows,

$$\hat{\pi}^{(m)} = \begin{cases} 0 & d_{\min}^{(m)} > d_{\text{th}}, \\ 1 - d_{\min}^{(m)}/10 & \text{otherwise}, \end{cases} \tag{11}$$

where in the following experiments, the threshold value was set as $d_{\text{th}} = 10$.

Step 2 The values of likelihood are adjusted as

$$\pi^{(m)} := \exp\left(100 \cdot \hat{\pi}^{(m)}\right), \tag{12}$$

where $\sum_{m=0}^{|M_t|} \pi^{(m)} = 1$.

Step 3 If $\pi^{(m)} \geq \pi_{\text{th}}$, it is judged that the target object is detected at the position of the mth particle and the posture $\theta^{(m)}(t)$ is calculated by PEM. Else the particle lost sight of the object, and the particle disappears.

Here, in the following experiments, the threshold values were set as $d_{\text{th}} = 10$ and $\pi_{\text{th}} = 0.1$.

4 Robust Target Tracking by Estimating of Dynamics

Since PEM is executed at the same frequency as the number of particles, high-speed execution of the proposed method is possible. Moreover, since the

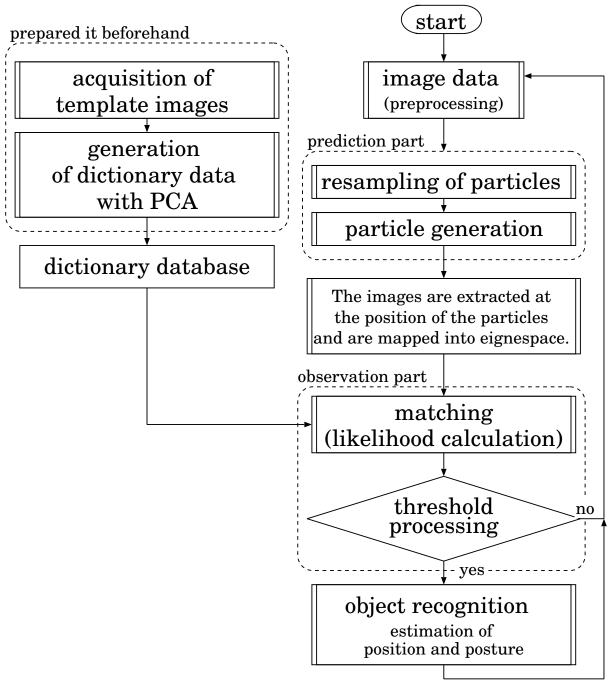


Figure 1: Flow of the processing of proposed method.

posture of the object can be estimated by using PEM, the position of the moving target object at the next time frame can be estimated by high accuracy. Namely, the particles can be generated intensively to the place where the possibility that the target moves is high by using the estimated posture $\theta^{(m)}$ as follows,

$$s_x^{(m)}(t) = s_x^{(m)}(t-1) + r_x \cos\left(\theta^{(m)}(t-1)\right) + \Delta_x,$$
$$s_y^{(m)}(t) = s_y^{(m)}(t-1) + r_y \sin\left(\theta^{(m)}(t-1)\right) + \Delta_y.$$

The outline of this processing is illustrated in Fig.2.

5 Experiments

5.1 Generation of dictionary database

First, the template images (64×64 [pixel]) of the object that changed continuously the appearance were acquired, and a part of them are shown in Fig.3(a). Moreover, the number of dimension of the eigenvector was experimentally set to be $k = 20$, and a dictionary database was generated (Fig.3(b), (c)).

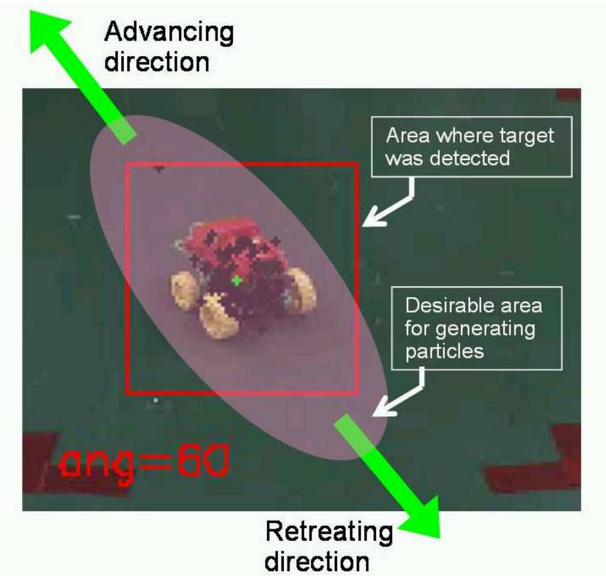


Figure 2: Method for improving robustness of detection and pursuit. The generation area of the particle is deformed according to the posture of the target.

5.2 Experimental conditions

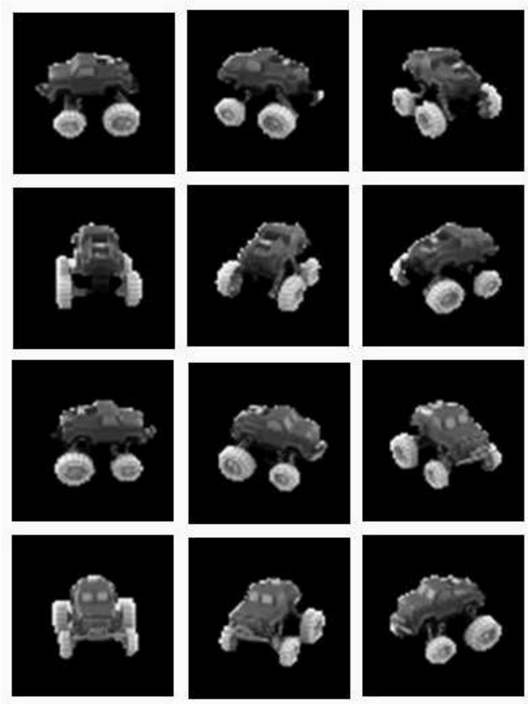
We experimented on the target detection and the tracking with a camera (160× 120 [pixel]) and a Windows PC (PentiumM 1.9 GHz). A general view of experimental environment is shown in Fig.5. The number of particles was 100.

5.3 Preprocessing of input image

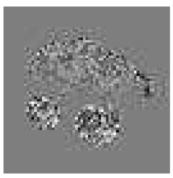
It is necessary to remove the background from the input image for highly accurate object detection by the PEM. In the following experiments, the input images are preprocessed as shown in Fig.5, i.e., gray-scaled image (Fig.5(a)) is transformed to a hue image (Fig.5(b)) by HSV transformation. Moreover, a similar hue image is generated from the image of which it captured beforehand i.e., a hue image of the background (Fig.5(c)) and the mask image is generated (Fig.5(d)). Herein, to decrease the influence of the shadow, the hue image is used without using the brightness image. At last, the target is extracted (Fig.5(e)) from a gray scale image (Fig.5(a)) by using the mask image.

5.4 Experimental results

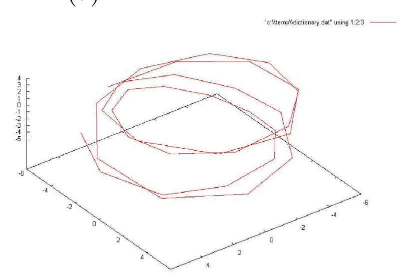
Examples of the experimental results of the detection, the posture estimation and the tracking of a movement target object are shown in Fig.6. The particle located at the center of the red rectangle (64×64 [pixel]) had the highest likelihood, and the figure in lower left shows the direction angle of the object. Moreover, the particle with a higher likelihood is represented in more vivid green, and the black particles were lower than the threshold of likelihood and disappeared in the next frame. The processing speed was 14-15 [fps], high-speed processing was achieved. From these experiment results, it is understood that the particle tracked the object, although it fails in the detection and the posture presumption of the object when the object



(a)



(b)



(c)

Figure 3: Template images for the dictionary database; (a) a part of template images rotated by 30 [deg] from 0 [deg] to 330 [deg], (b) an example of re-constructed image ($k = 20$), and (c) an example of eigenspace manifold ($k = 3$).

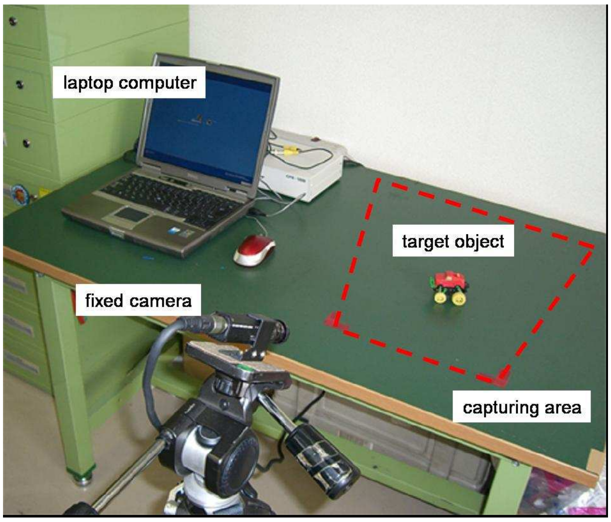
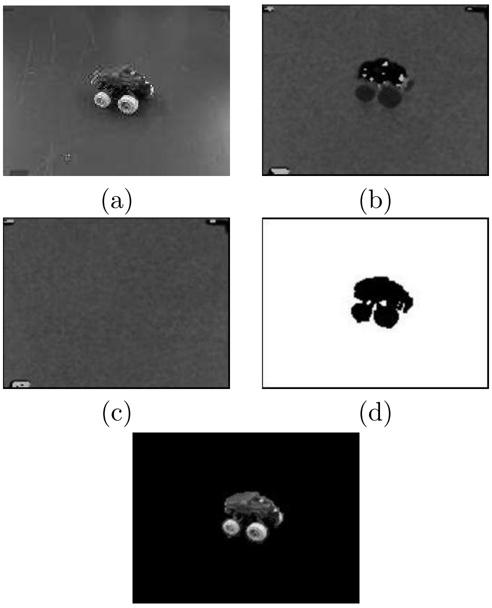


Figure 4: Over view of experimental environment.

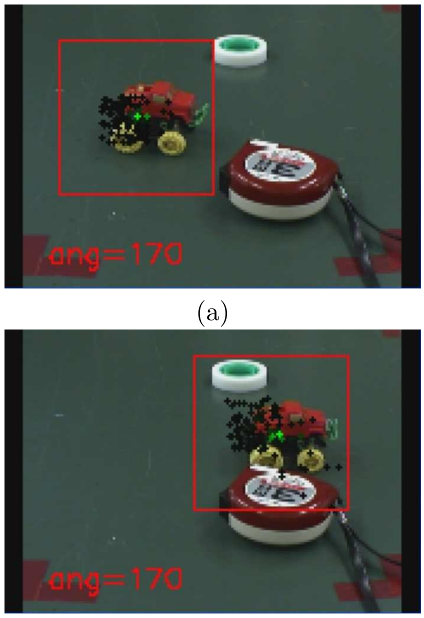


(a) (b) (c) (d) (e)

Figure 5: Flow of preprocessing to remove background from image: (a) a gray scale image, (b) a hue image generated by HSV transformation, (c) a background image captured beforehand, (d) a mask image generated by background subtraction, noise reduction and binarization are executed by using the (b) and the (c) images, and (e) extracted image of the target object by using the mask image.

moves at high speed, and the proposed method was effectively executed.

Further, Fig.7 shows the experiment result when



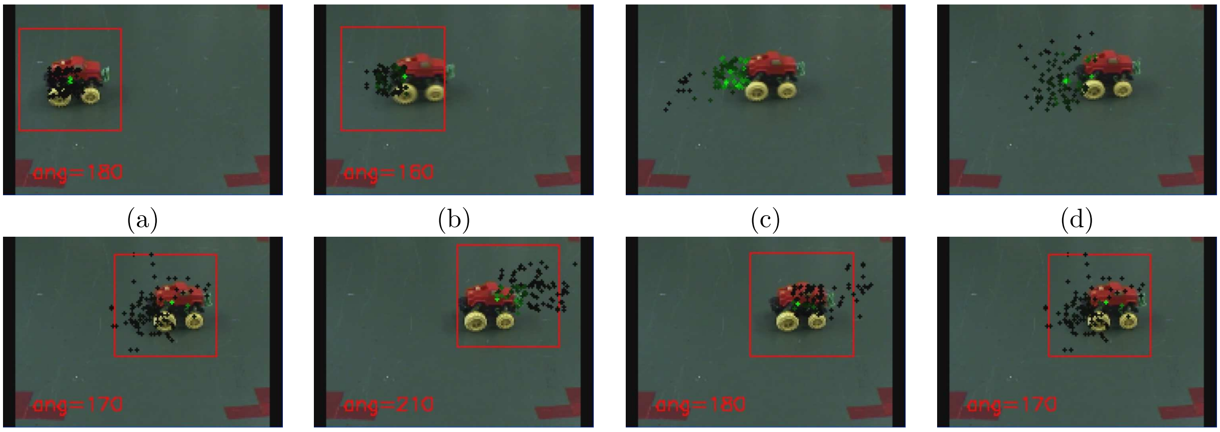
(a) (b)

Figure 7: experiment result when other objects are in surroundings of the target.

other objects are in surroundings of the target, and then it is understood that the detection only of the object was accurately executed.

6 Conclusion

In this research, we proposed the method for the robust posture estimation and the tracking of the target object in high-speed by executing PEM and PF simultaneously. However, the following problems have been left. First, since the size of appearance of the object changes according to the place of the camera image, the accuracy of the detection and the posture estimation by proposed method which uses constant size template images decreases. In general, to overcome this problem, the dictionary database generated from the template image of various sizes is used, or the input image is normalized to constant size, however, these methods increase the computational cost. Therefore, for example, it is possible to prepare the plural dictionary databases of the size of appearance of the object as a method of dealing with the problem, and it is a future work. Moreover, the propoed method is scheduled to be built in to the service robot [11, 12].



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 6: Example of result of the detection, the target tracking, and the posture estimation of a moving object. The figure in lower left shows the direction angle of the object, and the red rectangle ($64times64$[pixel]) shows the position of the detected target. Moreover, the particle with a higher likelihood is represented in more vivid green, and the particle with a low likelihood is represented in black.

References

[1] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking," International Journal of Computer Vision, Vol. 29, No. 1, pp. 5–28, 1998.

[2] M. Isard and A. Blake, "Condensation: Unifying low-level and high-level tracking in a stochastic framework," Proc. of ECCV98, pp. 893–908, 1998.

[3] A. Blake and M. Isard, Active Contours, Springer, 1998.

[4] J. MacCormick and M. Isard, " Partitioned sampling, articulated objects, and interface-quality hand tracking," Proc. of ECCV2000, pp. II–3–19, 2000.

[5] K. Nummiaro, E. Koller-Meier and L. V. Gool, "An adaptive color-based particle filter," Image and Vision Computing, Vol. 21, No. 1, pp. 99–110, 2003.

[6] A. Treptow, G. Cielniak and T. Duckett, "Real-time people tracking for mobile robots using thermal vision," Robotics and Autonomous Systems, Vol. 21, No. 54, pp. 729–739, 2006.

[7] T. Bando, T. Shibata, K. Doya and S. Ishii, "Switching particle filters for efficient visual tracking," Robotics and Autonomous Systems, Vol. 21, No. 54, pp. 873–884, 2006.

[8] H. Murase, S. K. Nayar, "Three-dimensional object recognition from appearance – parametric eigenspace method," Systems and computers in Japan, Vol. 26, No. 8, pp. 45–54, 1995.

[9] K. Okuma, A. Taleghani and N. D. Freitas, "A Boosted Particle Filter: Multi target Detection and Tracking," European Conference on Computer Vision, Vol. 3021 of LNCS, pp. 28–39, 2004.

[10] C. Shan, T. Tan and Y. Wei, "Real-time hand tracking using a mean shift embedded particle filter," Pattern Recognition, Vol. 40, Issue 7, pp. 1958–1970, 2007.

[11] T. Nishida, Y. Takemura, Y. Fuchikawa, S. Kurogi, S. Ito, M. Obata, N. Hiratsuka, H. Miyagawa, Y. Watanabe, F. Koga, T. Suehiro, Y. Kawamura, Y. Kihara, T. Kondo and F. Ohkawa, "Development of an Outdoor Service Robot," Proc. of SICE-ICCAS 2006, pp. 2052-2057, 2006.

[12] T. Nishida, Y. Takemura, Y. Fuchikawa, S. Kurogi, S. Ito, M. Obata, N. Hiratsuka, H. Miyagawa, Y. Watanabe, T. Suehiro, Y. Kawamura and F. Ohkawa, "Development of Sensor System for Outdoor Service Robot," Proc. of SICE-ICCAS 2006, pp. 2687-2691, 2006.