2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Robust Recognition and Pose Estimation of 3D Objects Based on Evidence Fusion in a Sequence of Images

Sukhan Lee, Seongsoo Lee, Jeihun Lee, Dongju Moon, Eunyoung Kim and Jeonghyun Seo

Abstract— **A sequence of images in multiple views rather than a single image from a single view is of great advantage for robust visual recognition and pose estimation of 3D objects in noisy and visually not-so-friendly environments (due to texture, occlusion, illumination, and camera pose). In this paper, we present a particle filter based probabilistic method for recognizing an object and estimating its pose based on a sequence of images, where the probability distribution of object pose in 3D space is represented by particles. The particles are updated by consecutive observations in a sequence of images and are converged to a single pose. The proposed method allows an easy integration of multiple evidences such photometric and geometric features as SIFT, color, 3D line, 2D square, etc. The integration of multiple evidences, including photometric and geometric features, in space and time makes the proposed method robust to various not-so-friendly visual environments. The experimental results with a single stereo camera show the validity of the proposed method in an environment containing both textured and texture-less objects.**

I. Introduction

The object recognition has been one of the major problems in computer vision and intensively investigated for several decades. In particular, the object recognition has played an important role for manipulation and SLAM in robotics field.

Many researchers proposed the various 3D object recognition approaches. Among them, the model-based recognition method is the most general one for recognizing the shape and object. It recognizes the objects by matching features extracted from the scene with stored features of the object [1][2]. The representative model-based object recognition studies are as follows.

The method proposed by Fischler and Bolles [3] uses RANSAC to recognize objects. It projects points of all models in the scene and determines if projected points are close to those of detected scene and recognizes the object through this. This method performs hypothesis and verification tasks several times thus making computational cost to be high. Olson [4] proposed the pose-clustering method for the object recognition. This method recognizes an object by producing the pose space discretely and finding cluster including the object to search. As for disadvantages of this method, data size is quite big because the pose space is 6-dimentional and pose cluster can be detected only when the accurate pose is generated. In the next, David et al. [5] proposed the approach that the recognition and pose estimation are solved simultaneously by minimizing energy function. But it may not be converged to minimum value in functional minimization method due to high non-linearity of the cost function. In addition, the spin-image based recognition algorithm in cluttered 3D scenes was suggested by Johnson and Herbert [6] and Andrea Frome et al. [7] compared the performance of the 3D shape context with the spin-image. Jean Ponce et al. [8] introduced the 3D object recognition approach using affine invariant patches. However, these methods are mostly tested with several scenes which have enough and accurate depth data. Most model-based 3D object recognition algorithms [9][10][11] including above mentioned have used only one scene or view, so that these approaches cannot robustly handle the changes in illumination, scale and view direction, while the proposed approach can do by applying the probabilistic pose to the consecutive scenes.

The main contribution of this paper is to develop a probabilistic method based on a sequence of images to recognize an object and to estimate its pose. The proposed method handles the object pose probabilistically. The probabilistic pose is drawn by particles and is updated by consecutive observations extracted from a sequence of images. The proposed method can recognize not only textured but also texture-less objects because the particle filtering framework of the proposed method can deal with various features such as photometric features (SIFT-Scale Invariant Feature Transform [12], color) and geometric features (line, square).

Fig. 1 illustrates a flow chart of the proposed method composed of six procedures. First of all, the valid features in an input image are selected by the cognitive perception engine (CPE) which perceives automatically an environment

Manuscript received September 15, 2006. This work is supported by the Intelligent Robotics Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea. This work is also supported in part by the Science and Technology Program of Gyeonggi Province as well as in part by the Sungkyunkwan University.

Sukhan Lee is with the school of information and communication engineering of Sungkyunkwan University, Suwon, Korea (corresponding author to provide phone: 82-31-299-7150; fax: 82-31-290-6479; e-mail: lsh@ece.skku.ac.kr).

Seongsoo Lee, Jeihun Lee, Dongju Moon, Eunyoung and Jeonghyun Seo are with the school of information and communication engineering of Sungkyunkwan University, Suwon, Korea. (e-mail: lss0703, jeihun81, dj_moon, samsam99, dimdim@ece.skku.ac.kr).

1-4244-0602-1/07/$20.00 ©2007 IEEE.

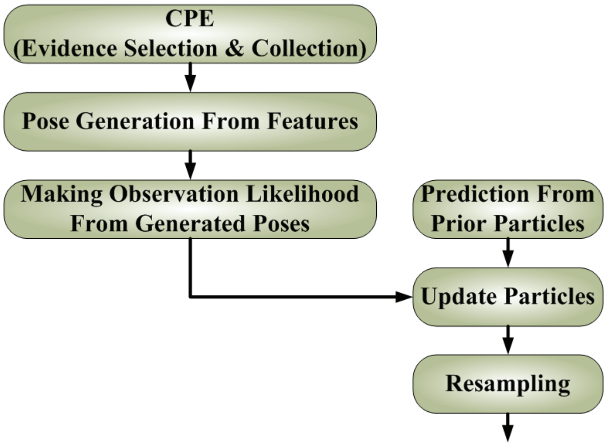

CPE
(Evidence Selection & Collection)
Pose Generation From Features
Making Observation Likelihood From Generated Poses
Prediction From Prior Particles
Update Particles
Resampling

Fig. 1 Flow chart of the proposed method

from a scene and keeps the evidences of all objects for their recognition. However, the CPE is omitted in this paper because we are currently developing the CPE. We assume that the valid features for each object in a current scene are already defined to the CPE. The multiple poses are then generated by features extracted from a scene. This poses are used for making observation likelihood. The particles representing the object pose are propagated from the previous state using the motion information. The weights are assigned to the predicted particles. Finally, we resample the particles according to their weights for obtaining important particles. These procedures are repeated until the particles are converged to a single pose.

II. Object Matching From Features

A. Pose generation from line & square feature

The 3D line feature is used for a texture-less object such as a refrigerator as shown in Fig. 2. All lines are firstly extracted from 2D images and these lines are converted to 3D lines through mapping 3D points corresponded to 2D lines. Since the Hough transform, the most famous algorithm for 2D line fitting, primarily finds strong lines in the scene, we made a simple algorithm based on the edge following approach in order to extract all lines needed for generating an object pose. [13] First of all, the edges are drawn by the canny edge algorithm. Then, we categorize the edges as horizontal, vertical and diagonal line segments based on the connection of edges. The 2D lines are found by connecting each line segments with adjoining line segments based on aliasing problem of lines in 2D. 3D lines can be obtained, if there are corresponding 3D points at the pixels of 2D lines.

An object to be recognized is represented by a set of 3D lines defined in the database. Multiple poses may be respectively spread to possible positions of the object in the environment. For efficiency, the salient lines on the object predefined in database are used to find the possible poses generated around 3D lines in the scene using two parameters in terms of the orientation and relationship. Firstly, in case of the refrigerator, lines vertical or horizontal to floor can be used because the refrigerator always stands up vertically to floor.

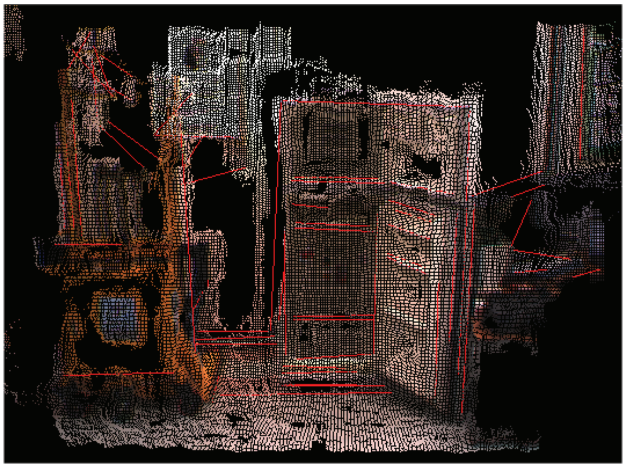
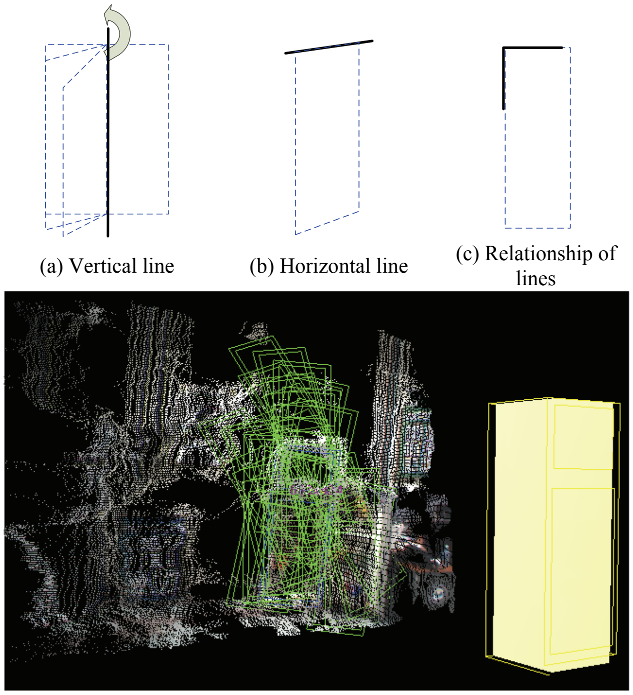


Fig. 2 3D lines extracted from depth image



(a) Vertical line

(b) Horizontal line

(c) Relationship of lines

(d) Generated poses using lines

Fig. 3 The generation of multiple refrigerator poses using lines in a scene

Fig. 3 (a) and (b) show how the multiple poses are generated by utilizing the vertical and horizontal lines. In case of (a), many hypotheses are generated by rotating vertical lines around the central vertical line. In case of horizontal line (b), the fewer hypotheses are formed because this line can give an approximate orientation of the object relatively. Also, if the relation between a certain line and surrounding lines in the scene is similar to that between lines in the model, the poses can be generated around the line as is shown in Fig. 3 (c). Fig. 3 (d) represents the multiple poses with green colored model generated by 3D line feature in a scene.

Fig. 4 shows the result which finds possible squares at 2D

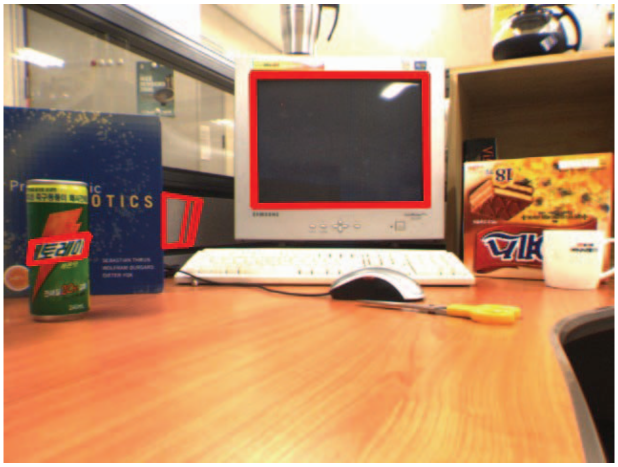


Fig. 4 The squares extracted from a scene for recognizing a monitor

image for recognizing the CRT monitor because a square composed of four lines is the salient lines for the CRT monitor.

In this paper, the object pose generated from features is represented by homogeneous transform matrix. The similarity weight should be calculated for identifying an object and is determined by the two factors:

a) The first factor N_{total_lines} is to check the number of lines which should be viewed in a generated hypothesis assuming that there is no sensing error.

b) The second factor $N_{matched\_lines}$ is the number of lines matched to the 3D object line model assumed from a generated pose at a scene.

The similarity weight for *j*th object location, w_j, is given by

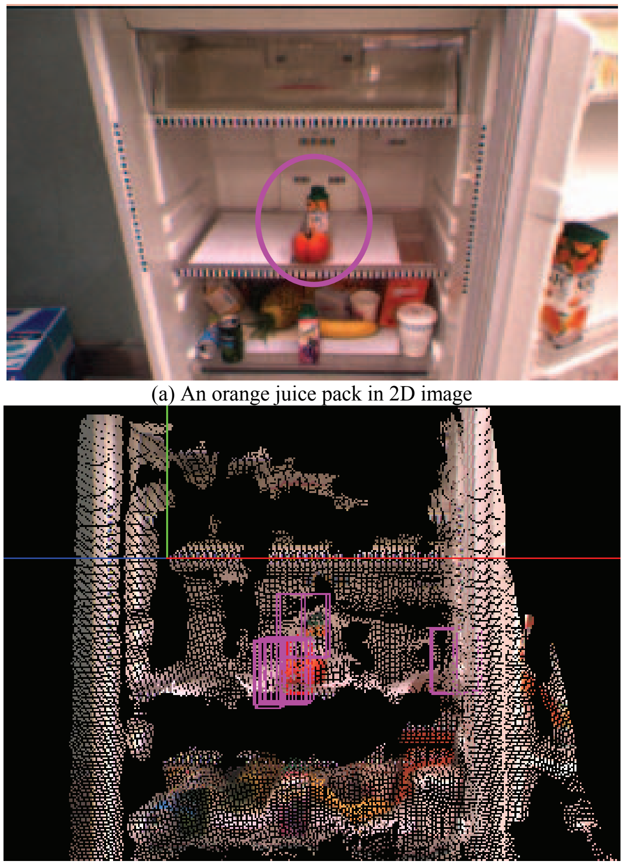
$$w_j = \frac{N_{matched\_lines}}{N_{total\_lines}} \qquad (1)$$

B. Pose generation from SIFT feature

The object pose can be generated by calculating a transformation between the SIFT features [12] measured at current frame and the corresponding ones in the database. The transformation is represented by a homogeneous transform matrix. The feature distances between the SIFT features from the scene and those from an object are first calculated, and the feature set that has the similar characteristics is deducted. The object pose can be generated using the 3D location from depth image if the matched features are 3 or more. The similarity weight for *j*th object location, w_j, is represented by

$$w_j = \frac{N_{matched\_SIFT}}{N_{total\_SIFT}} \qquad (2)$$

where N_{total_SIFT} is the number of SIFT features on the matched model composed of SIFT features among the database and $N_{matched\_SIFT}$ represents how many the SIFT features extracted from a scene are matched to those on the corresponding model among the database.



(a) An orange juice pack in 2D image

(b) Generated poses using SIFT

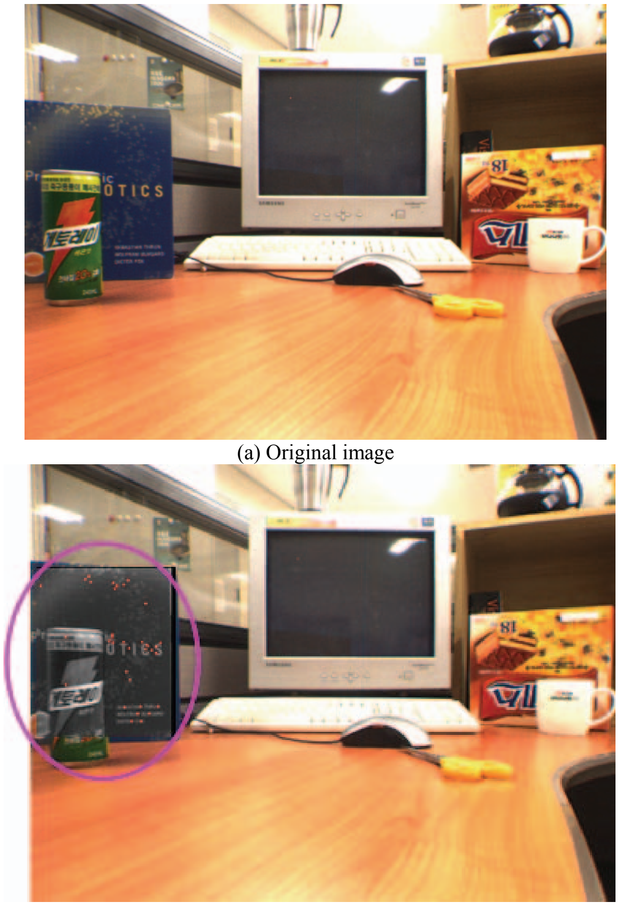
Fig. 5 The generation of multiple orange juice poses using SIFT features

Fig. 5 (a) and (b) show the orange juice pack in 2D image and the generated poses of an orange juice pack with a pink hexahedron in depth image.

C. Location generation from color information

The object with a particular color can be segmented by the color in the current scene. Although the segmented region can not provide an object's orientation, the object's location can be generated using the segmented region and depth image. In homogeneous transform matrix, the rotation part is defined by an identity matrix and the translation part represents an object's location. The similarity weight for *j*th object location, w_j, is denoted as a predefined constant with a comparatively small value in comparison with the similarity weight of the object pose generated by the other features. In particular, the color information can be combined with the other features.

Fig.6 shows the original image and the segmented region using blue color for recognizing a blue book. We use only SIFT features designated by red points within the segmented region in Fig. 6 (b) for matching them to database. The trial of this combination is closely connected with efficiency.



(a) Original image

(b) Segmented region

Fig. 6 Region of interest using blue color

D. Calculation of matching error covariance

We use the matching error covariance as a factor for correcting an object pose in particle filtering. Fig. 7 represents the matching error between three matched points. P1, P2 and P3 are the measured points at current frame while TP1, TP2 and TP3 mean the points transformed from ones in the database using a homogeneous transform matrix. The matching error of each point, e_i, is given by

$$\mathbf{e}_i = \begin{bmatrix} \mathrm{P}_i(x) \\ \mathrm{P}_i(y) \\ \mathrm{P}_i(z) \end{bmatrix} - \begin{bmatrix} \mathrm{TP}_i(x) \\ \mathrm{TP}_i(y) \\ \mathrm{TP}_i(z) \end{bmatrix} = \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \end{bmatrix} \tag{3}$$

where (*x*), (*y*) and (*z*) represent 3D location. The matching error covariance with zero mean for the *j*th object pose, $\mathbf{S}_j$, is calculated by

$$\mathbf{S}_j = \frac{1}{n}\sum_{i=1}^{n} \mathbf{e}_i \mathbf{e}_i^T \tag{4}$$

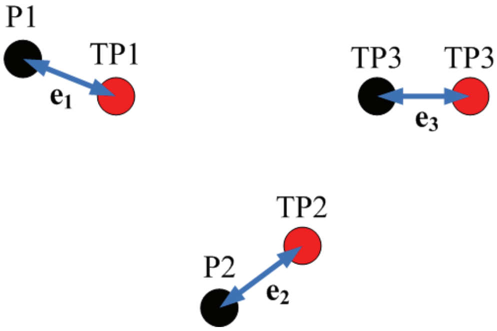


Fig. 7 The matching errors between the matched points

where *n* is the number of matched points. $\mathbf{S}_j$ is used for calculating an observation likelihood and is a predefined constant if the object pose generated by color information. The predefined constant will be comparatively a large value.

III. Particle Filtering Framework

The recognized object pose is estimated by particle filtering in a sequence of images over time in order that we represent the object pose with an arbitrary distribution. Let $\mathbf{O} = [x, y, z, \phi, \theta, \psi]^T$ represent the 3D object pose with respect to the camera frame, where $[x, y, z]^T$ describes translations along respective axes, and $[\phi, \theta, \psi]^T$ describes *roll*, *pitch* and *yaw Euler angles*. The probability distribution of the object pose at time t, $\mathbf{O}_t$, is represented by *k* particles.

$$\mathbf{O}_t \sim \{\mathbf{O}_t^{[1]}, ..., \mathbf{O}_t^{[k]}\} \tag{5}$$

A. Motion model

The particles of the object pose at time t-1 $\{\mathbf{O}_{t-1}^{[1]}, ..., \mathbf{O}_{t-1}^{[k]}\}$

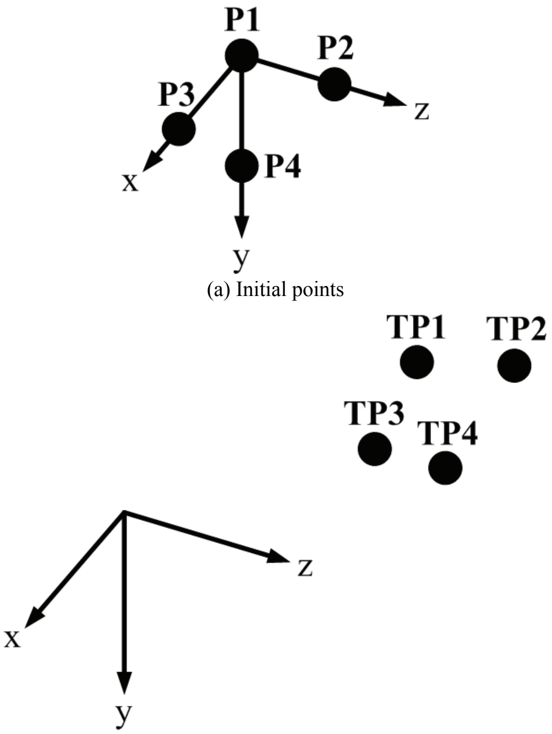
The particles of the object pose at time t-1 $\{\mathbf{O}_{t-1}^{[1]}, ..., \mathbf{O}_{t-1}^{[k]}\}$ are used to generate a probabilistic prediction of the object pose at time t with the following probabilistic motion model:

$$\mathbf{O}_t^{[i]} \sim p(\mathbf{O}_t \mid \mathbf{O}_{t-1}^{[i]}, \mathbf{u}_t), \quad (i = 1, ..., k) \tag{6}$$

where $\mathbf{u}_t$ is a camera motion control between time *t*-1 and time *t*.

B. Observation model

The multiple object poses $\{\mathbf{O}^{[1]}, ..., \mathbf{O}^{[m]}\}$ generated from features at current frame without prior particles are used for making observation model, where *m* is the number of generated objects at current frame. Here, we designate four points (P1, P2, P3, P4) at camera frame as Fig. 1 (a). The four points are transformed by the homogeneous transform matrix parameterized by the six spatial degrees of freedom. Fig. 1 (b)



(a) Initial points

(b) Transformed points with homogeneous transform matrix

Fig. 8 The designated four points for making the observation likelihood

shows the transformed points (TP1, TP2, TP3, TP4) with an arbitrary homogeneous transform matrix. Using homogenous transform matrices composed of multiple object poses at current frame $\{\mathbf{O}^{[1]},...,\mathbf{O}^{[m]}\}$ and the prior particles $\{\mathbf{O}_t^{[1]},...,\mathbf{O}_t^{[k]}\}$, we obtain the set of the four points (TP1, TP2, TP3, TP4) transformed from (P1, P2, P3, P4). Let (Ob_TP1[i], Ob_TP2[i], Ob_TP2[i], Ob_TP2[i]) represent the transformed points with $\mathbf{O}^{[i]}$ while (St_TP1[i], St_TP2[i], St_TP2[i], St_TP2[i]) mean those with $\mathbf{O}_t^{[i]}$. Using the Mahalanobis distance metric, we define the observation likelihood $p(\mathbf{Z}_t \mid \mathbf{O}_t^{[i]})$:

$$p(\mathbf{Z}_t \mid \mathbf{O}_t^{[i]}) = \sum_{j=1}^{m} w_j \exp\left[-0.5 \times \sum_{l=1}^{4} \left\{ \begin{aligned} &(\text{Ob\_TP}j - \text{St\_TP}j)^T \\ &\times \mathbf{S}_j^{-1}(\text{Ob\_TP}j - \text{St\_TP}j) \end{aligned} \right\} \right] \tag{7}$$

where w_j and $\mathbf{S}_j$ are the similarity weight and the 3x3 matching error covariance matrix related to transformed points with $\mathbf{O}^{[j]}$, respectively. Note that w_j is used for identifying an object and $\mathbf{S}_j$ is a parameter for correcting a pose. The designated four points for making the observation likelihood is used for estimating the object's orientation as well as its location. If the observation measures only the object's location, a single point transformed from the origin at camera frame P1 will be available for our framework because the other points are assigned for both object's location and orientation. It is also important to note that this approach assures that the observation likelihood can be calculated easily by the Mahalanobis distance between points even though both the particles of the state and the measurement extracted from features are represented by homogeneous transform matrices.

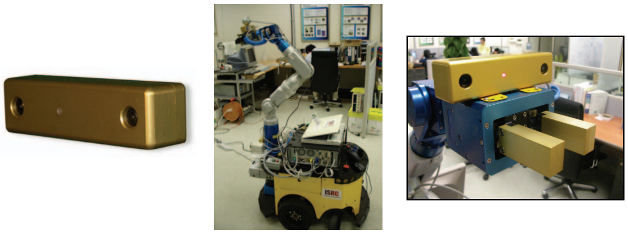
C. Re-sampling & additional sampling from the observation

The importance weight is assigned to each particle, which represents an object pose, using (7). According to the particle's weights, we resample k particles with their weight to $k-k_a$ particles, where k_a is the number which additionally sampled from the current observation. To add k_a particles to the particles of the state, we resample m object poses $\{\mathbf{O}^{[1]},...,\mathbf{O}^{[m]}\}$, measured at current frame without a help of the prior particles, with w_j of (7) to k_a particles. It should be noted that the additionally sampling from the observation prevents the particles from the degeneracy phenomenon or impoverish problem because there can be a lot of particles incorrectly predicted from the previous state in particle filtering. The priority of additional particles generated by the observation depends on the trade-off between k and k_a.

IV. Experimental Results

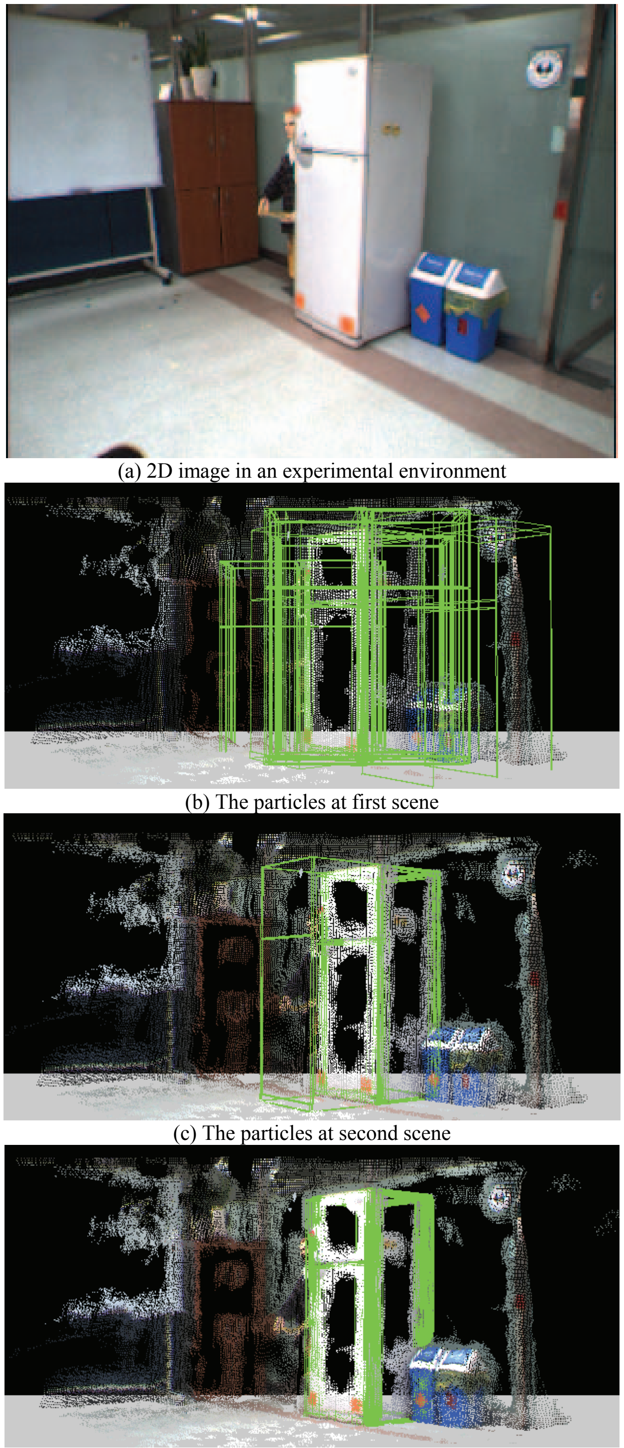
This paper focuses on recognizing an object while estimating its pose concurrently in a sequence of images. The proposed method is tested in textured and texture-less objects. The robot used in the experiment is a PowerBot–AGV with a Bumblebee stereo camera mounted on the end effecter of an arm with an eye-on-hand configuration as is seen in Fig. 9. The camera motion information in (6) is calculated by the internal encoder.

Fig. 10 illustrates how multiple particles representing the refrigerator pose are updated in a sequence of scenes and are converged to a single pose. In the first frame, the particles are initialized to possible poses using the salient line features. In the second scene, the multiple poses can be generated by the line features and the proposed particle filtering fuses these poses and the prior particles propagated from previous state.



(a)Stereo camera (b) Robot (c) Eye on hand

Fig. 9 Equipments for Experiment



(a) 2D image in an experimental environment

(b) The particles at first scene

(c) The particles at second scene

(d) The particles at third scene

Fig 10. The distribution of particles in a sequence of images. (The particles are designated by green boxes.)

In this case, the refrigerator is converged in the third frame.

Fig. 11 shows the recognized result in a sequence of images such as Fig. 10. There are four objects including textured and texture-less ones in this experimental environment. The monitor is recognized by squares and lines, while the others are recognized by SIFT features. In particular, the blue book is recognized by only SIFT features near the blue book using color information. In this case, the convergence of all object poses is achieved in averagely five frames when the robot position is initially a few meters from the objects. In this experiment, the proposed method requires averagely the computing time less than 400ms per object at each frame, as is seen in Table I.

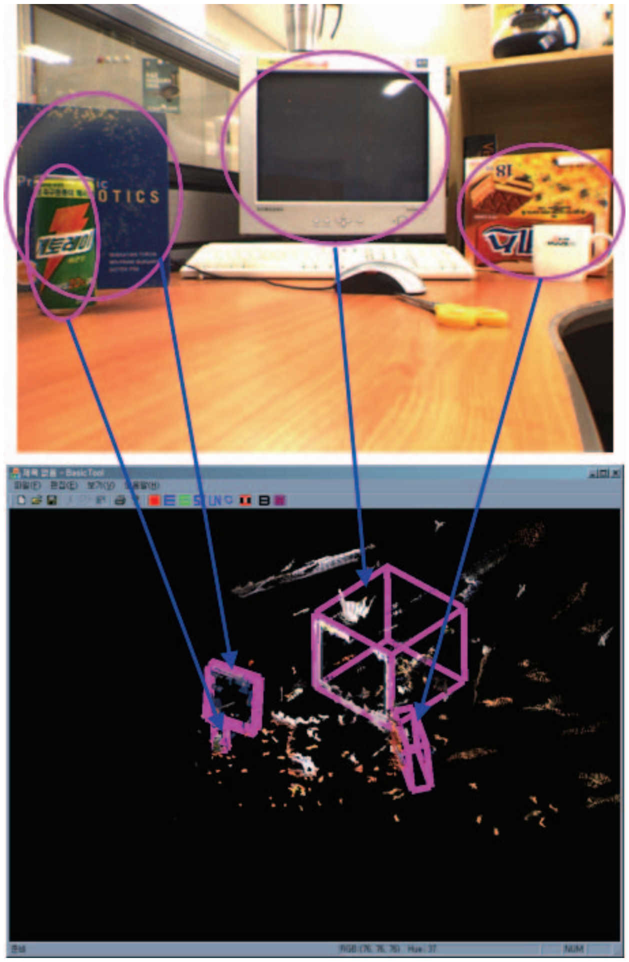


Fig. 11 The result of the converged object poses with their individual characteristics

TABLE I
THE AVERAGE COMPUTATION TIME AT EACH FRAME

Recognized Object	Total Time (ms)
Drinking Water (SIFT)	393
Ohyes Box (SIFT)	361
Blue Book (Color & SIFT)	211
Monitor (Square & Line)	298

V. Conclusion

We have concentrated on developing a probabilistic method using multiple evidences based on sequence of images to recognize an object and to estimate its pose. The proposed method represents probabilistically the recognized object pose with particles to draw an arbitrary distribution. The particles are updated by consecutive observations in a sequence of images and are converged to a single pose. The proposed method can recognize various objects with individual characteristics because it can incorporates easily multiple features such as photometric features (SIFT, color) and geometric features (line, square) into the proposed filtering framework. We test the proposed method with a stereo camera in an experimental environment including textured and texture-less objects. The experiment result demonstrates that the proposed method recognizes robustly various objects with individual characteristics such as textured and texture-less objects.

REFERENCES

[1] M. F. S. Farias and J. M. de Carvalho, “Multi-view Technique For 3D Polyhedral Object Rocognition Using Surface Representation,” *Revista Controle & Automacao.*, pp. 107-117, 1999.

[2] Y. Shirai, *Three-Dimensional Computer Vision*. New York: Springer Verlag.

[3] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Comm. Assoc. Comp. Mach*, vol. 24, no. 6, pp. 381-395, 1981.

[4] C.F. Olson. “Efficient pose clustering using a randomized algorithm,” *IJCV*, vol. 23, no. 2, pp.131-147, June 1997.

[5] M. P. David, D. F. DelMenthon, R. Duraiswami, and H. Samet, “Softposit: Simultaneous pose and correspondence determination,” *In 7th ECCV*, vol. 3, pp. 698-703, Copenhagen, Denmark, May 2002.

[6] A. E. Johnson and M. Hebert, “Using spin images for efficient object recognition in cluttered 3D scenes,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, May 1999.

[7] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik, “Recognizing Objects in Range Data Using Regional Point Descriptors,” *To appear in European Conf. Computer Vision*, Prague, Czech Republic, 2004.

[8] Fred Rothganger, Svetlana Lazebnik, Cordelia Schmid and Jean Ponce, “3D Object Modeling and Recognition Using Affine-Invariant Patches and Multi-View Spatial Constraints,” *CVPR*, vol. 2, pp. 272-280, 2003.

[9] I. Han, I. D. Yun, and S. U. Lee, “Model-based Object Recognition Using the Hausdorff Distance with Explicit Pairing,*” International Conf. Image Processing*, pp. 83-87, 1999.

[10] I. K. Park, K. M. Lee, and S. U. Lee, “Recognition and Reconstruction of 3D Objects Using Model Based Perceptual Grouping,” *In proceeding 15th International Conf. Pattern Recognition*, pp. 720-724, 2000.

[11] I. Weiss and M. Ray, “Model-based recognition of 3D Objects from single images,” *IEEE Trans. Pattern Analysis and Machine Intell.*, pp. 116-128, 2001.

[12] D. Lowe. “Object recognition from local scale invariant features,” *In Proc. 7th International Conf. Computer Vision (ICCV’99)*, pp. 1150–1157, Kerkyra, Greece, September 1999.

[13] Sukhan Lee, Eunyoung Kim and Yeonchool Park, "3D Object Recognition using Multiple Features for Robotic Manipulation," *IEEE International Conf. Robotics and Automation*, pp. 3768-3774, May 2006.