Proceedings of
IEEE International Conference on
Multisensor Fusion and Integration for Intelligent Systems
Seoul, Korea, August 20 - 22, 2008

Visual SLAM in Indoor Environments Using Autonomous Detection and Registration of Objects

Yong-Ju Lee and Jae-Bok Song

***Abstract*—For successful SLAM, landmarks for pose estimation should be continuously observed. This paper proposes autonomous detection of objects as visual landmarks for visual SLAM. Primitive features such as color and intensity, SIFT keypoints, and contour information are integrated to investigate environmental images and to distinguish objects from the background. Autonomous object detection can enable a robot to extract some objects without any prior information and it can help a vision system to cope with unknown environments. In addition, an adaptive weighting scheme and the use of a gradient of the gray scale are proposed to improve the performance of the proposed scheme. Using detected objects as landmarks, a robot can estimate its pose. A grid map of an unknown environment is built using an IR scanner and the detected objects are mapped in the grid map, which results in a hybrid grid/vision map. Visual SLAM using objects can have the less number of landmarks than other visual SLAM schemes using corners and lines. Various experiments show that the algorithm proposed in this paper can improve visual SLAM of a mobile robot.**

I. Introduction

When a robot navigates in an unknown environment, both accurate pose estimation of the robot and map building of the environment are important issues. Therefore, SLAM (Simultaneous Localization And Mapping) has been one of the most fundamental and challenging issues in the field of mobile robotics in recent years.

Range sensors (i.e., laser scanners, sonar sensors, and IR scanners) and vision sensors (i.e., monocular and stereo cameras) are usually employed for SLAM. Early researchers preferred range sensors because they could provide the range information directly, which made feature extraction easier than vision sensors. However, features that can be extracted from the range information are limited to lines and corners. On the other hand, a vision sensor offers much more information than a range sensor. Although a vision sensor requires complicated image processing to extract visual features, recent SLAM approaches tend to employ vision sensors as a main sensor.

As both range- and vision-based schemes use features to estimate the robot pose, it is clear that observation of features is the most important factor of successful SLAM. Among various types of features for visual SLAM, objects can serve as a good visual landmark because some object recognition methods are relatively robust and invariant to scale and rotation. Objects enable data association simpler than corners or lines and objects are also found easily in real environment [1].

Two approaches have been mainly used in object recognition; model-based scheme and appearance-based scheme. While a model-based (top-down) approach uses the model of an object, an appearance-based (bottom-up) approach does not use any prior knowledge of an object. Obviously, the latter is more suitable for SLAM which deals with unknown environments.

Researchers proposed several appearance-based approaches for object recognition. The saliency-based region selection strategy extracts multi-scale image features to find salient objects within a complex natural scene [2][3]. This scheme aims at searching objects as humans do and it can successfully extract objects from the background. However, it focused only on the image analysis and often extracts the objects that are too small or too easily movable (i.e., books and bags) to be used in navigation.

Another strategy for the appearance-based approach uses only SIFT keypoints or their clustering within an input image [4][5]. The main idea of these approaches is to extract the SIFT keypoints or to use clustering of SIFT keypoints as point landmarks. The landmarks are used only to estimate a robot pose and they do not offer any information on the environment, which means that they are just scale invariant points rather than meaningful objects for human (i.e., sinks and beds). These schemes have some drawbacks of using too many point features in a relatively small environment because too many features can cause inefficiency of SLAM or an increase in computational complexity.

The contribution of this paper is to propose a novel approach to object recognition that is applicable to SLAM. We propose an approach which finds useful objects without any prior information and exploits them as natural landmarks to estimate the robot pose and build an accurate environment map. The proposed scheme consists of the extraction method of various primitive features for reliable outputs and several steps for not selecting too small objects such as books and bags or parts of objects. If some objects are determined to be suitable for navigation, these detected objects are separated from the source image and registered in the database. These

This paper was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea.

Yong-Ju Lee is with the Dept. of Mechanical Eng., Korea University, Seoul, Korea (Tel.: +82 2 929 8501; fax: +82 2 3290 3757; e-mail: yongju_lee@korea.ac.kr)

Jae-Bok Song is a Professor of the Dept. of Mechanical Eng., Korea University, Seoul, Korea (Tel.: +82 2 3290 3363; fax: +82 2 3290 3757; e-mail: jbsong@korea.ac.kr)

registered objects are subsequently used as landmarks to estimate the robot pose. Figure 1 shows some useful objects for navigation of a mobile robot in real indoor environments.

Fig. 1 Objects useful for navigation in real environments.

By both visual feature-based EKF SLAM and the proposed recognition algorithm in this paper, the robot autonomously models an unknown environment. In this research, both range and vision sensors are used for SLAM and the SLAM process can be implemented in real-time although it may take long to recognize objects as the number of objects in the database becomes larger.

The remainder of this paper is organized as follows. Section 2 presents an overall structure of the proposed scheme and section 3 deals with extraction of various features from the camera image. Section 4 represents feature combination and object selection and section 5 describes EKF-based SLAM using extracted objects. Section 6 describes experimental results. Finally, section 7 presents conclusions.

II. Overall Structure of Autonomous Registration of Objects

Figure 2 shows an overall structure of the proposed object recognition scheme. For successful performance, it is desirable to use various types of features that are not correlated with each other. The proposed scheme uses five types of features such as SIFT keypoints, object contours, hue, saturation, and intensity.

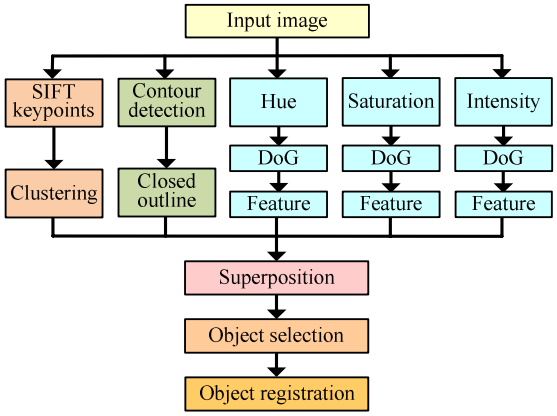

Fig. 2 Overall structure of the proposed scheme.

Among the five primitive features, the clustered region of SIFT keypoints and contours of objects are considered as object candidates. Inside the object candidates, color (hue, saturation, and intensity) information of the object candidates becomes a criterion to decide whether the object candidate can be selected as a useful object or not. After selecting an object, SIFT keypoints are exploited for object recognition. Object recognition offers the range and angle information of a recognized object because the stereo camera provides the position of the recognized object.

III. Feature Extraction from Camera Images

A. SIFT keypoint extraction and contour detection

SIFT (Scale Invariant Feature Transform) is one of the image recognition methods and extracts the feature points which are invariant to scale, rotation, and viewpoint [6]. The region where many SIFT keypoints exist is useful for navigation because the region is easily recognized by SIFT keypoints. The SIFT keypoints are clustered into several groups by the pixel distance in the image. Each group can be considered as a single object (although the keypoints of the group come from different physical objects) because SIFT keypoints tend to exist in the space whose pattern is obvious and remarkable (not flat and monotonous).

At the same time, the contour of an object is detected by the Canny edge algorithm [7]. The contour or outline of an object can help distinguish the object from the background or other objects. As objects in indoor environments are usually characterized by closed polygonal contours, it is useful to consider both the keypoints and contours together in selecting the objects. As an example, the detected contours and clustered regions of SIFT keypoints in the input image of Fig. 3(a) are marked as rectangles in Fig. 3(b) and (c), respectively. If the size of object candidate is too small (smaller than 40 pixels in both width and length), this candidate is discarded because it is not likely to be reliably recognized due to its small size or the distance from the robot.

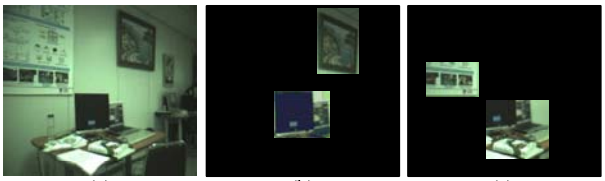

Fig. 3 (a) Input image, (b) contour detection, and (c) clustering of SIFT keypoints.

B. Decomposition of input image

The input image in the form of RGB (i.e., red, green, and blue channels) is transformed into the form of HSI (i.e., three properties of color; hue, saturation, and intensity channels) through the procedure introduced in [8]. The HSI space is more intuitive and gives more information than the RGB space because the HSI space is similar to the human cognitive system and its three channels are not correlated. In the hue channel, all colors are represented as the values between 0 and 360. The saturation channel represents the degree of purity. For instance, dark blue and light blue are determined by adjusting the saturation channel. The intensity channel

represents light information obtained by the conversion of the color image to the gray image.

C. Extraction of features from each feature image

Primitive features are extracted from the hue, saturation, and intensity channels. Gaussian convolution is conducted twice on each channel with variances of σ and 2σ. Boundaries become smooth through Gaussian convolution. Then, difference between the Gaussian convolution images represents the boundaries. The differences of Gaussian convolution images represent the complexity of patterns at hue (color), saturation (purity), and intensity (light) channels. The magnitudes of the features are represented as a gray scale image as shown in Fig. 4 and the feature images are obtained by Eq. (1).

$$I = \left| L(\sigma) - L(2\sigma) \right| \tag{1}$$

where $L(\sigma)$ and $L(2\sigma)$ are the Gaussian convolution images with masks whose variances are σ and 2σ, respectively. I of Eq. (1) represents the primitive feature image. The magnitudes of the features are proportional to the gray scale of the I image.

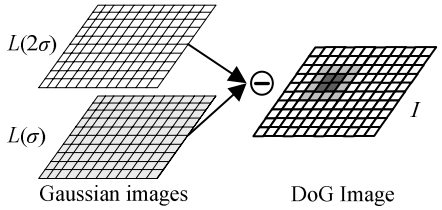

Fig. 4 DoG (Difference of Gaussian) image as a primitive feature image.

A result of feature extraction is shown in Fig. 5. A camera image and feature images of hue, saturation, and intensity are shown in Fig. 5(a), (b), (c), and (d), respectively. The final combined image is made by summing them. The three feature images are normalized before they are combined because they represent their features with different ranges.

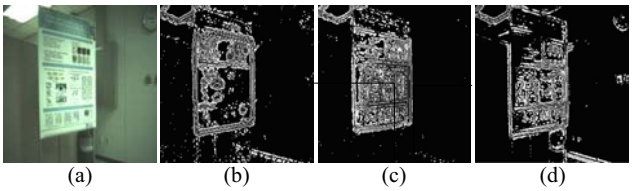

Fig. 5 Feature images; (a) Input camera image, (b) features extracted from hue image, (c) features extracted from saturation image, and (d) features extracted from intensity image.

IV. Combination of Primitive Features and Object Selection Mechanism

A. Combination of primitive features

The feature image extracted from hue, saturation, and intensity image inside an object candidate is the criterion for selecting suitable objects from the candidates. The gray scale of the feature images is related to salience. Salient objects or places look white in the feature images because the gray scale for the corresponding pixels is high.

Figure 6 shows examples of object candidates and their salience. Figure 6(a) is the input image. In Fig. 6(a), region A is the clustered region of SIFT keypoints and region B is the detected contour. The outer (yellow) rectangle in Fig. 6(a) represents a region of interest, which is used for prevention of the effect of insufficient information (i.e., the objects outside this region of interest are likely to be cut at the boundary of the camera image). Figure 6(b) represents the final combined image of Fig. 6(a). The three feature (hue, saturation, and intensity) images are combined with equal weights. Figure 6(c) shows the salience of regions A and B. In Fig. 6(c), all regions of Fig. 6(b) except A and B were discarded for better understanding. While region A is salient, region B is not salient, as shown in Fig. 6(c).

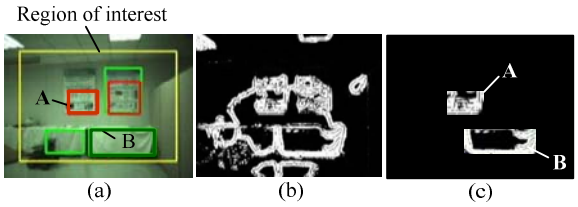

Fig. 6 (a) Object candidates represented as rectangles, (b) the final combined image where hue, saturation, and intensity feature images are combined with equal weights, and (c) HSI information of regions A and B in the final combined image.

For successful results, we use an adaptive weighting strategy. The weights of feature images are determined according to the distribution of the gray scale values. That is, the weight is increased (or decreased) for the weight of dense (or sparse) features. It is described mathematically by

$$I_F = \omega_H I_H + \omega_S I_S + \omega_I I_I \tag{2}$$

$$\omega_H = \frac{\sigma_S^2 \sigma_I^2}{\sigma_H^2 \sigma_S^2 + \sigma_S^2 \sigma_I^2 + \sigma_I^2 \sigma_H^2} \tag{3}$$

$$\omega_S = \frac{\sigma_H^2 \sigma_I^2}{\sigma_H^2 \sigma_S^2 + \sigma_S^2 \sigma_I^2 + \sigma_I^2 \sigma_H^2} \tag{4}$$

$$\omega_I = \frac{\sigma_H^2 \sigma_S^2}{\sigma_H^2 \sigma_S^2 + \sigma_S^2 \sigma_I^2 + \sigma_I^2 \sigma_H^2} \tag{5}$$

where σ represents the distribution of the gray scale in the feature images, ω the weight of each image and I the feature image. The subscripts H, S, I, and F mean hue, saturation, intensity, and final combined feature. Whenever various scenes are captured, the weights change.

An example of adaptive weighting is illustrated in Fig. 7. A rectangle is extracted by the contour detection algorithm and the region inside the rectangle is assumed to be a candidate for an object. The variance of hue, saturation, and intensity inside

the candidate is 2944, 1561, and 1584, respectively. From (3), (4), and (5), the weight becomes 0.21 for hue, 0.40 for saturation, and 0.39 for intensity. Features commonly extracted from all feature images become salient and they are represented in white, whereas non-salient areas in black.

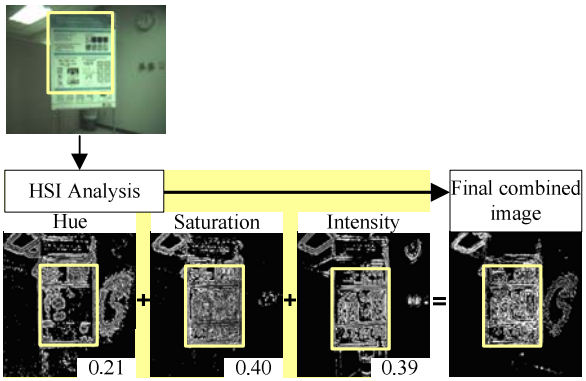

Fig. 7 Adaptive weighting; final combined image where hue, saturation, and intensity feature images are combined with different weights.

B. *Object selection mechanism*

We also propose a filtering step using gray scale values in the final combined image for robust performance. The main idea is to investigate the average gray scale values along the boundary (10 pixels from the boundary) in four directions; left, right, top, and bottom. The scheme of investigating the average gray scale values to the left of the object candidate is shown in Fig. 8. As the object is distinguishable from the background in the final combined image, the area outside of the object candidate is not salient.

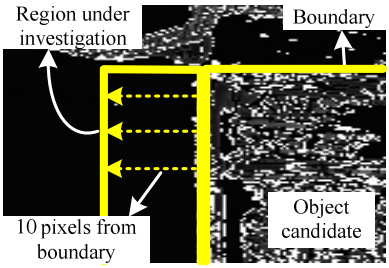

Fig. 8 Determination of the gradient from the boundary.

A more detailed explanation is described in Fig. 9. In Fig. 9(a), all the averages of the gray scale values outside the object candidate in four directions are low. On the other hand, in Fig. 9(b), the gray values outside of the object candidate are relatively high in all directions. Therefore, the object candidate of Fig. 9(a) is selected as an object, but that of Fig. 9(b) is discarded.

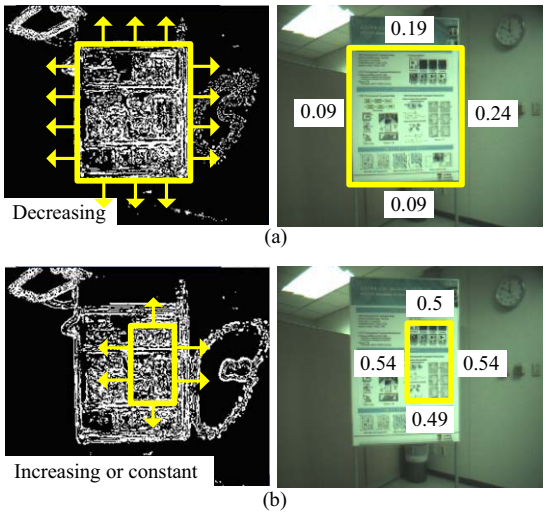

Fig. 9 Investigation of gradient of object candidates.

Figure 10 illustrates the recognized objects during navigation. By the proposed recognition method, a poster and part of a bookshelf were recognized, as shown in Fig. 10(a) and (b), respectively. In Fig. 10(c), another poster and a picture are matched from a quite long distance. In object recognition, the center of an object is selected as a point representing the object because the object has its own size at the input image. The red cross in the figures represents the center point of the recognized object. The affine transform, which calculates the geometrical relationship between the object recognized in the scene and that stored in the database, is used to extract the accurate center point with various viewpoints.

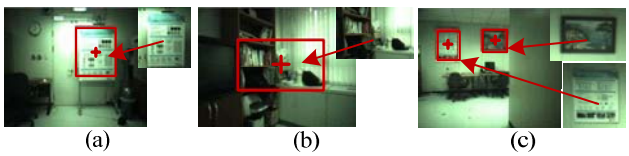

Fig. 10 Recognition of autonomously registered objects during navigation by the proposed method.

V. EKF-BASED SLAM

The EKF (Extended Kalman Filter) algorithm has proven to be the most appropriate framework in visual SLAM by much literature [9]. It compensates for the error accumulated due to both systematic and non-systematic errors during navigation. In EKF, the robot pose and landmark positions are stored in a state vector represented as **X**, and the position uncertainties of components of the state vector are stored in a covariance matrix denoted as **P**. The state vector and covariance matrix are updated recursively through sensor measurements.

A. Prediction

At the prediction stage, the state vector and its covariance matrix at time t are obtained as follows:

$$\hat{\mathbf{X}}_t^- = \mathbf{f}(\hat{\mathbf{X}}_{t-1}, \mathbf{u}_t, t) + \mathbf{w}_t \quad (6)$$

$$\mathbf{P}_t^- = \mathbf{F}_\mathbf{x}\mathbf{P}_{t-1}\mathbf{F}_\mathbf{x}^T + \mathbf{F}_\mathbf{u}\mathbf{Q}_t\mathbf{F}_\mathbf{u}^T \quad (7)$$

where $\hat{\mathbf{X}}_t^-$ and $\mathbf{P}_t^-$ are the predictions of the state vector and its covariance matrix at time t, respectively, and $\mathbf{u}_t$ is the displacement of the robot between time t-1 and time t. The vector $\mathbf{w}_t$ represents the process noise with zero mean and $\mathbf{Q}$ is the covariance matrix of the process noise. The matrices $\mathbf{F}_\mathbf{x}$ and $\mathbf{F}_\mathbf{u}$ are the Jacobian matrices of the nonlinear motion model $\mathbf{f}(\cdot)$ with respect to the state vector and the displacement $\mathbf{u}_t$, respectively.

If the robot observes a feature, it compares this feature with the features in the state vector $\mathbf{X}$. If it turns out to be a new feature, this feature is initialized and included in the state vector and its covariance matrix. If it is found to be one of the existing features, the EKF algorithm conducts the update stage.

B. Update

The state variables, the robot pose and landmark positions and the covariance matrix of the state vector are updated by the measurement of the sensor at the update stage. In this paper, the measurement is obtained from object recognition in the form of a relative range and orientation of the object from the robot. The state vector and its covariance matrix $\mathbf{P}$ at time t are updated as follows:

$$\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}_t^T(\mathbf{H}_t\mathbf{P}_t^-\mathbf{H}_t^T + \mathbf{R}_t)^{-1} \quad (8)$$

$$\hat{\mathbf{X}}_t = \hat{\mathbf{X}}_t^- + \mathbf{K}_t(\mathbf{Z}_t - \hat{\mathbf{Z}}_t) \quad (9)$$

$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\mathbf{P}_t^- \quad (10)$$

where $\mathbf{K}_t$ represents the Kalman gain, and $\mathbf{H}_t$ is the Jacobian matrix of the sensor model with respect to the state vector. The error on the pose of the robot due to disturbances is compensated by the Kalman gain which is proportional to the difference between predictions and measurements. If none of landmarks are matched, the uncertainties of landmarks are kept unchanged.

$$\hat{\mathbf{X}}_t = \hat{\mathbf{X}}_t^- \quad (11)$$

$$\mathbf{P}_t = \mathbf{P}_t^- \quad (12)$$

In this case, only the robot pose is calculated by the motion model and the uncertainty of the robot pose increases.

VI. Experimental Results

Various experiments were performed using a robot equipped with an IR scanner (Hokuyo PBS-03JN) and a stereo camera (Videre STH-MDI-C). The camera is used for object recognition and the IR scanner is used to build a grid map of the environment. The experimental environment consists of three rooms, as shown in the Fig. 11(a). The total area of the experimental environment is 10m x 10m. Figure 11(b) shows the CAD data of the environment which will be compared with the map built by the proposed algorithm. The grid size of both the CAD data and the grid map built by SLAM is 10cm.

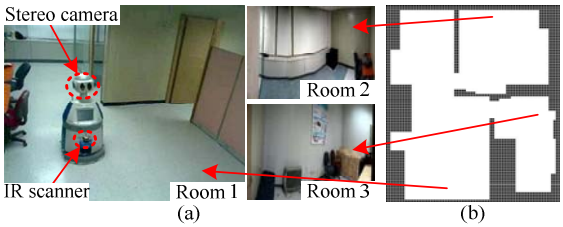

Fig. 11 Experimental environment; (a) mobile robot platform and experimental environment, and (b) CAD data.

Figure 12 illustrates the mapping process of the experimental environment using the proposed algorithm. In the experiment, pictures and bookshelves are selected as objects for estimating the robot pose. Since the cluttered environment such as chairs, table legs and small objects such as books are useless for localization, these objects are not selected. Fig. 12(a) represents the initial state of the robot. In Fig. 12(b), the robot moves in the environment, builds the grid map and marks the objects in their own position. In room 2, it was difficult to detect some objects because of non-systematic errors generated by a carpet and slip of the wheels of the mobile robot. The map was distorted after navigating room 2, as shown in Fig. 12(c). However, the map was recovered from distortion by observing the registered object again, in Fig. 12(d). The recovery from the distortion is a result of data association, and object recognition can eliminate accumulated errors. It follows that object recognition makes data association easily compared to other features such as corners or lines.

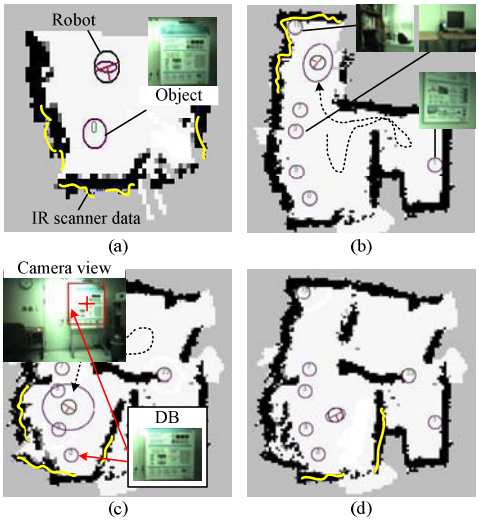

Fig. 12. Indoor SLAM with autonomous object registration.

Figure 13 shows the constructed map of the environment

shown in Fig. 11 and the comparison of the trajectory estimated by the odometry (dotted line) with that by the proposed EKF-based SLAM (solid line) approach. The constructed map is referred to as a hybrid grid/vision map because it contains visual features as well as occupancy grids. Black objects or legs of tables cannot be represented in the map because an IR scanner cannot detect a light absorbing object and the object whose width is narrower than its angular resolution. The positional error of the resulting map is about ±20cm and orientation error is 5°.

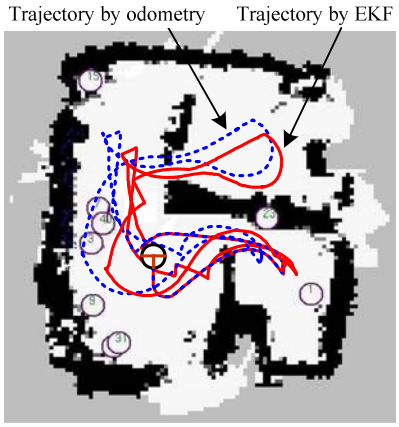

Fig. 13 Comparison of robot trajectory by odometry (dotted) with that by EKF-based SLAM (solid).

Several objects of different sizes can be registered from an identical object due to some factors such as lighting condition which affects the RGB image and thus HSI information. However, the hybrid grid/vision map is not much affected by the light condition because matching of the SIFT keypoints is relatively robust. For example, object B is hardly matched to object A in Fig. 14. Research on the consistent selection of an object under various lighting conditions is under way.

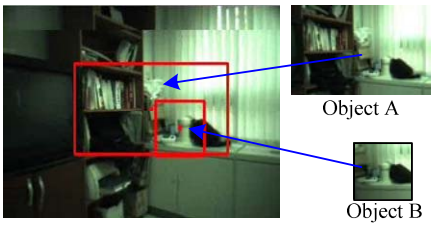

Fig. 14 Objects of different sizes from the same object.

VII. Conclusions

Object recognition is useful for navigation of a mobile robot because various objects exist in indoor environments. Current object recognition schemes require object information in the database, so objects which do not exist in the database cannot be recognized. However, the proposed scheme can recognize objects without any information. The experimental results of the proposed scheme and its application to SLAM are shown in the previous chapter. From the experiments, the following conclusions were drawn.

1. It is possible to autonomously select an object or a group of objects and register them as visual landmarks for SLAM without human interference.
2. The proposed scheme can solve data association or loop-closing problems relatively easily compared to that based on the range sensors alone because object recognition offers quite accurate feature matching results.

References

[1] P. Newman, D. Cole, and K. Ho, "Outdoor SLAM using Visual Appearance and Laser Ranging," *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 1180-1187, May, 2006.

[2] L. Itti, C. Koch, and E. Niebur, "A Model of Saliency-based Visual Attention for Rapid Scene Analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254-1259, November, 1998.

[3] D. Walther, U. Rutishauser, C. Koch, and P. Perona, "On the usefulness of Attention for Object Recognition," 2nd Workshop on Attention and Performance in Computational Vision, pp. 96-103, May, 2004.

[4] R. Sim and J.J. Little, "Autonomous vision-based exploration and mapping using hybrid maps and Rao-Blackwellised particle filters," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2082-2089, October, 2006.

[5] R.H. Luke, J.M. Keller, M. Skubic and S. Senger, "Acquiring and Maintaining Abstract Landmark Chunks for Cognitive Robot Navigation," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2566- 2571, August, 2005.

[6] D.G. Lowe, "Distinctive image features from scale invariant keypoints," *Int'l Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, Novemver, 2004.

[7] P. Bao, L. Zhang, and Xiaolin Wu, "Canny Edge Detection Enhancement by Scale Multiplication," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.27, No.9, pp.1485-1490, September, 2005.

[8] R.C. Gonzalez, and R.E. Woods, *Digital Image Processing*. Addison-Wesley Publishing Company. 1992.

[9] G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csobra, "A solution to the Simultaneous Localization and Map Building Problem," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, pp. 229-241, June, 2001.