Preprints of the 18th IFAC World Congress
Milano (Italy) August 28 - September 2, 2011

Robust Extraction of Arbitrary-Shaped Features in Ceiling for Upward-Looking Camera-Based SLAM

Seo-Yeon Hwang*, Jae-Bok Song*, Mun Sang Kim**

** School of Mechanical Eng., Korea University, Seoul, Korea*
(*Tel: 82-2-3290-3363; e-mail: etoile02, jbsong@korea.ac.kr*)
*** Center for Intelligent Robotics, Korea Inst. of Science and Technology, Seoul, Korea*
(*Tel: 82-2-958-5623; e-mail: munsang@kist.re.kr*)

Abstract: This paper proposes a novel scheme of extracting the features in the ceiling using an upward-looking camera for SLAM in indoor environments. The conventional approaches based on corner or line features have difficulties in associating adjacent indistinguishable features since their descriptors do not have enough information to distinguish them. Therefore, we used various properties of the objects in the ceiling, such as the distribution of nodes, and size and orientation strength of the region of interest. Also, the similarities among adjacent features are calculated to distinguish between unique and non-unique features. The non-unique features are discarded before they are registered to the feature map. The robustness of the proposed scheme is verified by several experiments using a mobile robot. The extended Kalman filter is used to estimate both robot pose and feature position and orientation, and the experimental results show that the proposed scheme successfully works in various indoor environments.

Keywords: Feature extraction, upward-looking camera, mobile robots, monocular vision-based SLAM.

1. INTRODUCTION

Simultaneous localization and mapping (SLAM) is one of the most difficult problems to solve in mobile robotics. In general, a robot should know its pose in order to build a map of the environment, and pose estimation can be successfully made by using the map. Therefore, the map building process and localization should be performed simultaneously in unknown environment. In previous researches, the range sensors such as laser scanners and IR scanners were adopted to achieve stable navigation with high pose estimation accuracy. However, due to their high costs and several difficulties in practical use, other approaches which use low-cost sensors such as monocular cameras and sonar sensors have drawn much attention in recent years. Especially, the use of the monocular cameras for SLAM has been a challenging issue, since the monocular cameras do not require any additional modules (e.g., a feature matching module using the images from two cameras) for calculating distances to the objects unlike the expensive stereo cameras.

Various monocular vision-based SLAM approaches have been proposed according to the camera directions (e.g., forward-looking (Se, 2002), side-looking (Lemaire, 2005), upward-looking (Hwang et al., 2008), and hand-held (Davison, 2007) cameras), and the upward-looking camera-based approach has been successfully applied to small-sized robots such as cleaning robots. Since the distance between the upward-looking camera and the ceiling is maintained constant during navigation, the images taken by the camera have only small changes in both scale and affine parameters. From this property, the feature extraction process does not require the complex schemes such as SIFT (scale-invariant feature transform) (Lowe, 2004) for robustness to the change of the distance and angle between the feature and the camera.

However, like the other camera-based approaches, corner and line features have been adopted as main features for the upward-looking camera-based SLAM methods (Jeong et al., 2006). The corner and line features suffer from the data association failure due to adjacent indistinguishable features and illumination changes, since their descriptors have insufficient information to handle such problems. In addition, the images taken from the ceiling usually have a relatively small number of corner and line features than those from the other directions of the camera, so the features cannot be observed for a long time in some cases, which tends to result in localization failure.

To overcome these difficulties, we propose a novel scheme of extracting robust features from the ceiling and a SLAM method based on the extended Kalman filter (EKF). A region of interest (ROI) is defined based on the edge detection and enhanced labeling process. Then, node points are detected on the contour of each ROI and their distribution is stored as a single property. Also, the size and orientation strength of the ROI are calculated and regarded as the important properties. When the sufficient properties are found, the ROI is considered as a visual feature. The stored properties are used as a descriptor of each feature, and they are associated with observed features by calculating the weighted sum of similarities. A feature, which is similar to the adjacent feature, is regarded as a non-unique feature and discarded in the extraction process to prevent localization failure. All properties are extracted based on the methods which are robust to illumination changes. The center point and orientation of the feature are used as observations in the EKF,

and the poses of the robot and features are estimated simultaneously. The main contribution of the proposed method is that the features can be extracted and tracked from arbitrarily shaped ceiling objects. This enables the SLAM algorithm to successfully perform in various indoor environments.

The remainder of this paper is organized as follows. Section 2 introduces the ceiling feature extraction method in detail and section 3 describes the feature matching method. The EKF-based SLAM method using the proposed visual features is discussed in Section 4. Experimental results are presented in section 5 and conclusions are drawn in section 6.

2. CEILING FEATURE EXTRACTION

2.1 ROI Selection

The contours of the object provide useful information to divide the objects into separate regions, and they are robustly extracted under illumination changes. In this paper, the Canny edge detector (Canny, 1981) is adopted to extract the contours of the objects in the ceiling as shown in Fig. 1. Compared to the other contour detection methods (e.g., Sobel operator), the Canny edge detector provides more reliable results for the proposed scheme since the detected contours can be clearly distinguished.

To divide the objects in the ceiling into separated regions, a labeling process was performed based on the result of contour detection. Since the white lines in Fig. 2(a) represent the contours, only the black regions were used in the labeling process. The Grassfire algorithm (Pitas, 1993) was adopted to perform the labeling process. However, when the labeling process was performed directly for the detected contours, the neighboring regions could be easily connected as shown in Fig. 2(b) because several contours were not completely closed. These barely connected regions were unstable and could not be continuously observed as the robot moved, so we added a pre-processing step to enhance the labeling performance. The basic concept is to place a small circle in an empty area in the contour image and to move the circle in the image not to overlap with the contours as shown in Fig. 3(a). Then the coverage of the center of each circular window results in separated labels as represented in Fig. 3(b) and Fig. 3(c). The labels that are adjacent to the image boundary were discarded since the remaining parts of the labels were unknown and the shape of the label could not change as the robot moved. Finally, these labels were selected as the ROIs.

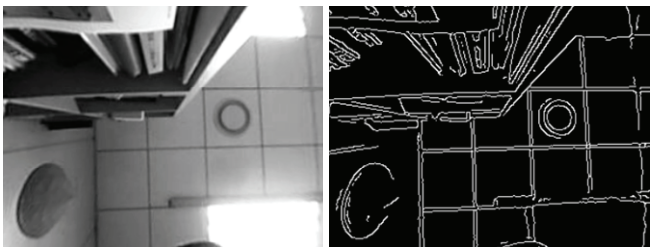

Fig. 1. Contour detection of ceiling image using Canny edge detector.

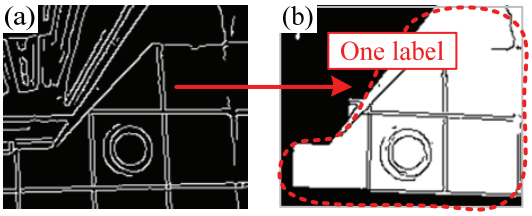

Fig. 2. Labeling result without pre-processing: (a) detected contours and (b) a labeled region (white).

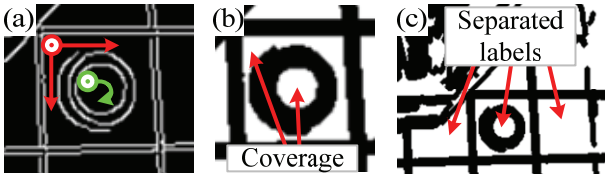

Fig. 3. Labeling result with pre-processing: (a) moving circular windows in empty areas, (b) coverage of circular windows, and (c) successfully separated labels.

2.2 Descriptor for ROI

The image patches are usually used as a descriptor of visual features such as corner and line features. To measure the similarity between two features, the sum of squared difference (SSD) or normalized cross correlation (NCC) is widely used. However, the results from these intensity-based methods are sensitive to illumination changes even though the resulting values are normalized. In this paper, therefore, various properties such as the distribution of nodes, the size of ROI, and the orientation strength of the ROI are used as descriptors.

Nodes are detected by the FAST (features from accelerated segment test) algorithm (Rosten et al., 2006) from the label shown in Fig. 3(c). Consider a labeled region shown in Fig. 4, where $P_{\text{mean}} = (m_u, m_v)$ corresponds to the mean of all pixels in this label and regarded as the origin. The position of a node is represented in the polar coordinates, where r is the distance from the mean point to the node, and θ is the angle measured from the V-axis. The size of the ROI, which is determined from the number of pixels in the region, can be used as an important descriptor since it is kept almost the same even when the noise exists. The orientation strength of the ROI is calculated by measuring the major and minor axes of the pixel distribution in the region. The distribution can be represented as a covariance matrix as follows:

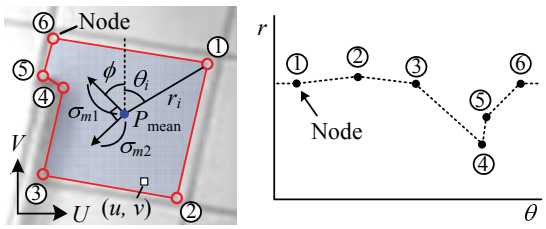

Fig. 4. Nodes and orientation of ROI.

$$\mathrm{cov}(u,v) = \begin{bmatrix} \mathrm{E}[(u-m_u)^2] & \mathrm{E}[(u-m_u)(v-m_v)] \\ \mathrm{E}[(v-m_v)(u-m_u)] & \mathrm{E}[(v-m_v)^2] \end{bmatrix} = \begin{bmatrix} a & b \\ b & c \end{bmatrix} \quad (1)$$

where (u, v) is the coordinate of the pixel in the ROI. Equation (1) can be also represented by

$$\mathrm{cov}(u,v) = M^T \begin{bmatrix} \sigma_{m1}^2 & 0 \\ 0 & \sigma_{m2}^2 \end{bmatrix} M \quad (2)$$

where σ_{m1} and σ_{m2} are the magnitudes of the major and minor axes, and M is the rotation matrix, respectively. Then, the major and minor axes of the distribution, the orientation, and the orientation strength are derived by using the elements a, b, and c from (1):

$$\phi = \frac{\pi}{2} - \frac{1}{2}\tan^{-1}\left(\frac{2b}{a-c}\right) \quad (3)$$

$$\sigma_{m1} = \sqrt{a + b\tan\left(\frac{\pi}{2}-\phi\right)},\ \sigma_{m2} = \sqrt{c - b\tan\left(\frac{\pi}{2}-\phi\right)} \quad (4)$$

$$R_{\mathrm{ori}} = \sigma_{m1}/\sigma_{m2} \quad (\sigma_{m1} > \sigma_{m2}) \quad (5)$$

where ϕ is the orientation of the ROI and R_{ori} is the orientation strength.

3. FEATURE MATCHING

A feature matching process is required to provide reliable observations from the successive images. The matching process is performed by comparing the descriptors between two features. Successful matching is very important since the matching failure can easily lead to localization failure. Reliable matching is obtained by calculating the similarities in terms of the node distribution, size, and orientation strength of the ROI at once. Also, the uniqueness of each feature is determined to prevent the matching failure in advance.

3.1 Similarity between two features

The difference between two node distributions is calculated by shifting the θ values of the nodes in the θ - r coordinates as shown in Fig. 5 since the node distribution from the observed feature can be rotated due to the robot motion. The shift angle which minimizes the difference is selected and the similarity between the node distributions is calculated. A node pair is determined when the observed node comes into the pre-defined matching range of the stored node. The distance between the nodes from the stored and observed features is defined by

$$d(n_i, n_j) = \sqrt{(r_i - r_j)^2 + \{(\theta_i - \theta_j - \theta_{\mathrm{shift}})\cdot k\}^2} \quad (6)$$

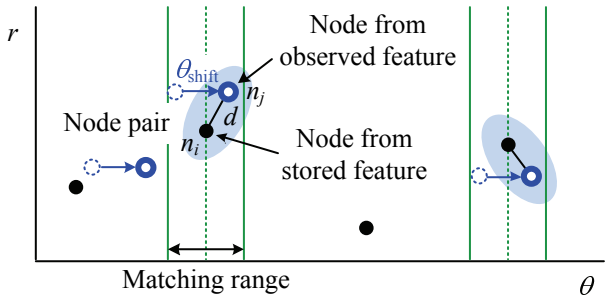

Fig. 5. Calculating difference between stored and observed node distributions.

where θ_{shift} represents the shift angle of the observed node along the θ-axis, $n_i = (\theta_i, r_i)$ and $n_j = (\theta_j+\theta_{\mathrm{shift}}, r_j)$ are the stored and observed node points in a node pair, and k denotes the scale factor for the θ coordinate of the node. The entire nodes from the observed feature are shifted by θ_{shift}. The similarity between two node distributions, S_{node}, is represented by calculating the minimum value of the mean squared error (MSE) by changing $\theta_{\mathrm{shift}} = 0$ to 2π as follows:

$$S_{\mathrm{node}} = 1 - \min_{0 \le \theta_{\mathrm{shift}} < 2\pi} \frac{1}{N}\sum \{d(n_i, n_j)\}^2 \quad (7)$$

where N is the number of node pairs. This approach can be applied even when the nodes are partially observed since only the node pairs have an effect on the similarity.

The similarity for the size, S_{size}, and the similarity for the orientation strength, S_{ori}, are defined by

$$S_{\mathrm{size}} = 1 - \frac{|A_{\mathrm{stored}} - A_{\mathrm{observed}}|}{A_{\mathrm{stored}}} \quad (8)$$

$$S_{\mathrm{ori}} = 1 - \frac{|R_{\mathrm{stored}} - R_{\mathrm{observed}}|}{R_{\mathrm{stored}}} \quad (9)$$

where A is the size of the ROI, and R is the orientation strength from (5), respectively.

Using S_{node}, S_{size}, and S_{ori}, the overall similarity between the stored and observed feature is defined as:

$$S = \frac{w_{\mathrm{node}}S_{\mathrm{node}} + w_{\mathrm{size}}S_{\mathrm{size}} + w_{\mathrm{ori}}S_{\mathrm{ori}}}{w_{\mathrm{total}}}$$
$$w_{\mathrm{total}} = w_{\mathrm{node}} + w_{\mathrm{size}} + w_{\mathrm{ori}} \quad (10)$$

where w_{node}, w_{size}, and w_{ori} are the weights for the node, size, and orientation strength, respectively. If the feature has no nodes, then w_{node}=0. The matching is regarded successful if S is larger than a threshold. In this paper, the threshold is set to 0.85.

3.2 Uniqueness

The matching failure occurs between the observed features when the indistinguishable features are adjacent. Therefore, the stable localization can be achieved when unique features are used for SLAM. The uniqueness of each feature is

determined by calculating the similarities from the adjacent features as follows:

$$\text{Uniqueness} = \min\{1 - S(F_{\text{int}}, F_{\text{adj},k}) \cdot g(D(F_{\text{int}}, F_{\text{adj},k}))\} \quad (11)$$

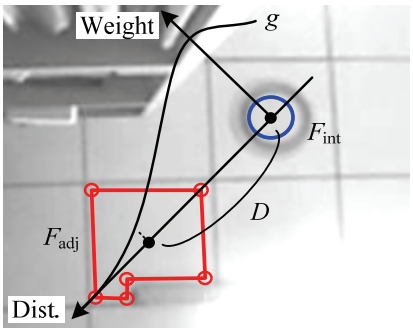

Fig. 6. Calculation of feature uniqueness.

where F_{int} represents the feature of interest, $F_{\text{adj},k}$ is the feature k adjacent to F_{int}, S is the similarity function defined by Eq. (10), D is the function which calculates the Euclidian distance between the center points of two features, and g is the Gaussian weighting function. According to Eq. (11), the uniqueness becomes large when there are no similar features around F_{int}. If the uniqueness is larger than the pre-defined threshold, the feature is regarded as unique and used as a landmark in the extended Kalman filter (EKF).

4. EKF-BASED SLAM

In contrast to stereo cameras, monocular cameras cannot measure the distance to the object directly. Therefore, the distance need to be probabilistically estimated by analyzing the bearing information on the object on each image taken by the monocular camera. The camera itself has an observation error, thus the initial uncertainty of a landmark pose (i.e., position and orientation) can be represented as an ellipsoid which follows the Gaussian distribution. If the landmark is observed from multiple robot poses, then the uncertainty will converge as shown in Fig. 7. The landmarks which have small uncertainties have a large effect on the estimated robot pose when they are observed.

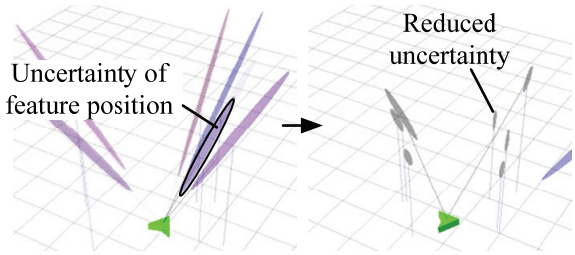

Fig. 7. Basic concept of monocular vision-based SLAM.

In this paper, we adopted the EKF (Thrun et al., 2005) to estimate both the feature pose and the robot pose. The EKF is widely used to deal with the nonlinearities involved in robot motion and sensor measurements. The visual features extracted from Section 2 are used as the landmarks in the EKF, and the state vector is defined as follows:

$$X = [X_R, X_{L1}, \cdots, X_{Ln}, X_{LO1}, \cdots, X_{LOm}]^T \quad (12)$$

$$X_R = [{}^G x_R, {}^G y_R, {}^G \theta_R] \quad (13)$$

$$X_L = [{}^G x_L, {}^G y_L, {}^G z_L] \quad (14)$$

$$X_{LO} = [{}^G x_L, {}^G y_L, {}^G z_L, {}^G \phi_L] \quad (15)$$

where X_R represents the robot pose, X_L is the landmark position, and X_{LO} is the landmark position having the orientation. Note that the superscript "G" indicates the global coordinate frame. The covariance matrix corresponding to the state vector is given by

$$P = \begin{bmatrix} P_R & P_{R,L} & P_{R,LO} \\ P_{L,R} & P_L & P_{L,LO} \\ P_{LO,L} & P_{LO,L} & P_{LO} \end{bmatrix} \quad (16)$$

where P_R, P_L, and P_{LO} are the covariance matrices for the robot pose, the landmark position, and the landmark position having the orientation, respectively. The off-diagonal elements are the cross-correlation matrices of P_R, P_L, and P_{LO}. The state vector and covariance matrix are estimated by the following prediction and update steps.

4.1 Prediction

The state vector and its covariance matrix at time t are predicted from the state at time t-1 by applying the odometry information between time t-1 and t as follows:

$$\hat{X}_t^- = f(\hat{X}_{t-1}, u_t) \quad (17)$$

$$P_t^- = \nabla F_x P_{t-1} \nabla F_x^T + \nabla F_u Q \nabla F_u^T \quad (18)$$

where f is a function of the system dynamics, the input u_t=($\Delta s_{r,t}$, $\Delta s_{l,t}$) is the distances traveled by the right and left wheels between time t-1 and t, Q is the covariance matrix of the process noise, and $\nabla F_x = \partial f / \partial X$ and $\nabla F_u = \partial f / \partial u$ are the Jacobian matrices of the nonlinear function f with respect to the state and input, respectively. The predicted state before the update step is represented by the superscript "-" in the notations.

4.2 Update

In the update step, the predicted states are corrected from the landmark observations. The relationships between the sensor and the global coordinates are defined by an observation model h as follows:

$$\hat{Z}_t = h(\hat{X}_t^-) \quad (19)$$

where $\hat{Z}_t$ represents the predicted positions and orientations of the landmarks in the image coordinate at time t from the predicted state $\hat{X}_t^-$ in the global coordinates. The observation model h is defined as

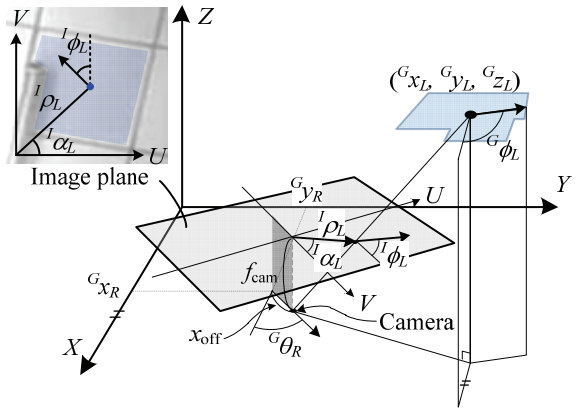

Fig. 8. Observation model of proposed ceiling feature.

$$h=\begin{bmatrix} {}^{I}\rho_L \\ {}^{I}\alpha_L \\ {}^{I}\phi_L \end{bmatrix}=\begin{bmatrix} \sqrt{(x_{RL})^2+(y_{RL})^2}\times\dfrac{f_{\text{cam}}}{{}^{G}z_L} \\ \tan^{-1}\left(\dfrac{y_{RL}}{x_{RL}}\right)-{}^{G}\theta_R \\ {}^{G}\phi_L-{}^{G}\theta_R \end{bmatrix} \tag{20}$$

where

$$x_{RL}={}^{G}x_L-{}^{G}x_R-x_{\text{off}}\cos{}^{G}\theta_R\,, \tag{21}$$

$$y_{RL}={}^{G}y_L-{}^{G}y_R-x_{\text{off}}\sin{}^{G}\theta_R\,, \tag{22}$$

f_{cam} is the focal length of the camera which can be obtained from the calibration process, x_{off} is the camera offset from the rotation center of the robot to the front direction, and $[{}^{I}\rho_L, {}^{I}\alpha_L, {}^{I}\phi_L]^T$ is the position and orientation of the landmarks in the image coordinate system as shown in Fig. 8. For the landmarks which do not have the orientation, ${}^{I}\phi_L$ is not considered thus $h=[{}^{I}\rho_L, {}^{I}\alpha_L]^T$ is used as the observation model. Finally, the state vector and its covariance matrix at time t are updated as follows:

$$\hat{X}_t=\hat{X}_t^-+K_t(Z_t-\hat{Z}_t) \tag{23}$$

$$P_t=(I-K_tH_t)P_t^- \tag{24}$$

$$K_t=P_t^-H_t^T(H_tP_t^-H_t^T+R_t)^{-1} \tag{25}$$

where K is the Kalman gain, $H=\partial h/\partial X$ is the Jacobian matrix of the observation model with respect to the state vector, and R is the covariance of the sensor noise.

5. EXPERIMENTAL RESULTS

The experiments were conducted in the real environment using the MobileRobots Pioneer 3-DX mobile robot equipped with an upward-looking camera with a field of view of 120° as shown in Fig. 9. The camera image was undistorted using the distortion parameters obtained from the calibration process before the experiment. The features in the ceiling were extracted by the proposed scheme and the initial uncertainty of each feature was set to 0 to 5 m since the experiments were conducted in indoor environments. The unstable features, which were rarely observed, were removed from the EKF process during navigation. The average speed of the robot was 40 cm/s, and the entire SLAM process worked in real-time.

Fig. 9. Experimental environment and mobile robot platform.

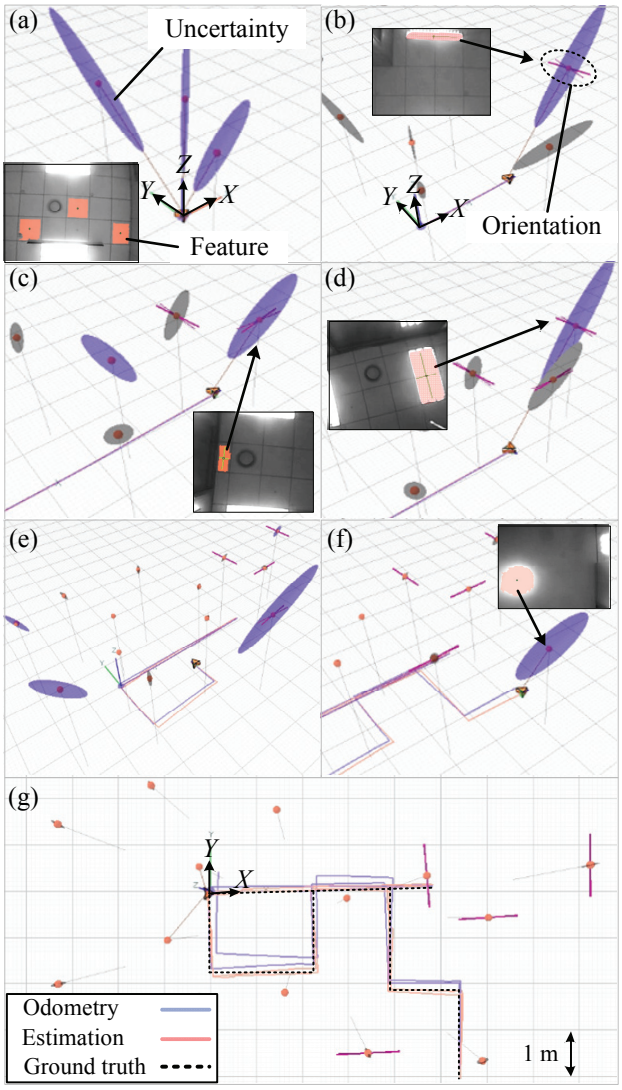

Fig. 10. Experimental results.

The experimental results are shown in Fig. 10. At the initial state of navigation, the uncertainties of each landmark were

set to large ellipsoids in Fig. 10(a). Three square-shaped features were extracted and no orientations were found from each feature. A rectangular feature was extracted from the ceiling lamp in Fig. 10(b), and its orientation was found. The uncertainty boundary of the landmark orientation is represented as two lines for each landmark. The second, the third, and the fourth features which have the orientations were extracted in Fig. 10(c), Fig. 10(d), and Fig. 10(e), respectively. A circular feature was extracted from a lamp in Fig. 10(f), and no orientation was found. The final result of SLAM is represented in Fig. 10(g). The robot returned back to the starting point, and the paths based on the odometry, estimation, and ground truth are illustrated in the figure. The robot pose was corrected so well that the position error was less than 10 cm.

6. CONCLUSIONS

In this paper, we proposed the feature extraction method for arbitrary-shaped objects for upward-looking camera based SLAM. The distribution of nodes, size, and orientation strength of a feature were used as the significant descriptors for the feature. The uniqueness of each feature is determined by comparing the descriptors of adjacent features, and the unique features were used as the landmarks in the EKF process. The successful results were acquired from various experiments, and the following conclusions were drawn.

1. Since the proposed scheme can successfully extract the features having arbitrary shapes, SLAM can be performed in a stable manner even in the environments where corner and line features do not exist.

2. The features extracted by the proposed scheme are robust to illumination changes since they are extracted from the contours and their descriptors are not based on the intensity information.

3. The proposed SLAM method can be applied to low-cost robots (e.g., cleaning robots, security robots, and service robots) since only the monocular camera and the mobile platform are required for its implementation.

ACKNOWLEDGEMENT

This research was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Knowledge Economy of Korea.

REFERENCES

Canny, J. (1981). A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8, 679-714.

Se, S., Lowe, D., and Little, J. (2002). Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *Int. Journal of Robotics Research*, 8(21), 735-758.

Lemaire, T., Lacroix, S., and Sola, J. (2005). A practical 3D bearing-only SLAM algorithm. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2449-2454.

Davison, A., Reid, I., Molton, N., and Stasse, O. (2007). MonoSLAM: real-time single camera SLAM. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(6), 1052-1067.

Jeong, W. Y., and Lee, K. M. (2006). Visual SLAM with line and corner features. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2570-2575.

Hwang, S. Y., and Song, J.-B. (2008). Upward monocular camera based SLAM using corner and door features. *Proc. of the 17th world congress of IFAC*, 1663-1668.

Lowe, D. (2004). Distinctive image feature from scale-invariant keypoints. Int. Journal of Computer Vision, 2(60), 91-110.

Pitas, I. (1993). *Digital image processing algorithms*, Prentice-Hall.

Rosten, E., and Drummond, T. (2006). Machine learning for high-speed corner detection. *Proc. of European Conf. on Computer Vision*.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*, MIT Press, Massachusetts.