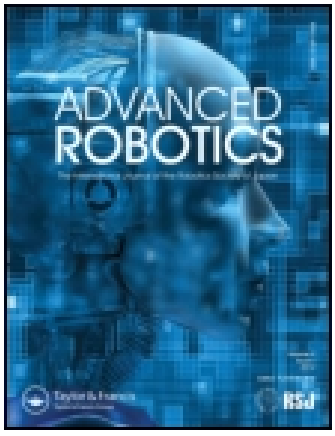

Advanced Robotics
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tadr20

Clustering and probabilistic matching of arbitrarily shaped ceiling features for monocular vision-based SLAM

Seo-Yeon Hwang [a] & Jae-Bok Song [a]
[a] School of Mechanical Engineering , Korea University , Seoul , Korea
Published online: 13 Apr 2013.

To cite this article: Seo-Yeon Hwang & Jae-Bok Song (2013) Clustering and probabilistic matching of arbitrarily shaped ceiling features for monocular vision-based SLAM, Advanced Robotics, 27:10, 739-747, DOI: 10.1080/01691864.2013.785377

To link to this article: http://dx.doi.org/10.1080/01691864.2013.785377

Advanced Robotics, 2013
Vol. 27, No. 10, 739–747, http://dx.doi.org/10.1080/01691864.2013.785377

Taylor & Francis
Taylor & Francis Group

FULL PAPER

Clustering and probabilistic matching of arbitrarily shaped ceiling features for monocular vision-based SLAM

Seo-Yeon Hwang and Jae-Bok Song*

School of Mechanical Engineering, Korea University, Seoul, Korea

(*Received 16 March 2012; accepted 4 July 2012*)

This paper presents improved extraction and matching methods for arbitrarily shaped (AS) ceiling features for monocular vision-based simultaneous localization and mapping. The feature descriptor, which is robust to illumination changes, comprises the vertex distribution, size, and orientation strength of the region of interest. However, to cope with the problem of vertices being detected at different positions in successive images, Bayes' rule is applied to preserve robust vertices and remove rarely observed vertices. Moreover, unstable features surrounded by similar features are clustered to create a robust feature by calculating their similarities to adjacent clusters. AS features from the proposed scheme are used as landmarks in the extended Kalman filter, and the effectiveness of the proposed scheme is verified through various experiments in real environments.

Keywords: mobile robot; ceiling; arbitrarily shaped feature; SLAM

1. Introduction

Visual features are widely used by mobile robots to build feature maps and localize themselves in unknown environments (i.e. simultaneous localization and mapping (SLAM)) [1] because they provide a variety of information from camera images. For example, corner and line features can be found in environments having sufficient textures, and they are used as main landmarks.[2] The keypoints from scale-invariant feature transform have also been widely adopted as robust features because they can be extracted from various scales and viewpoints.[3] In addition, door features can be employed to divide the environment into meaningful areas.[4]

SLAM using an upward-looking camera [5–7] is one of the most promising methods among the various monocular vision-based approaches. Ceiling image features cause small changes in feature scales and affine parameters because the distance between the camera and a ceiling feature is almost constant during navigation. Moreover, unlike forward- and side-looking [8] schemes, this approach is not affected by dynamic obstacles such as moving people. As a result of these properties, the algorithm does not require complex feature extraction and matching processes and thus can be successfully implemented in real-world applications such as cleaning robots and service robots.

Recent trends in upward-looking camera-based SLAM include integrating different types of features (e.g. corner, line, lamp, door),[9–11] since it enables stable navigation in larger environments. Using more than one type of feature increases the chance of observations, leading to reliable estimation of the robot pose. However, some real environments do not contain predefined features such as corner and line features, resulting in frequent localization failure. In addition, these conventional features, which have intensity-based descriptors, are not robust against changes in environmental illumination. To extract the features that are invariant to illumination and affine changes, the region-based features (e.g. maximally stable extremal region [12] and salient regions [13]) were proposed. They have been frequently adopted in various applications, [14,15] showing successful results, but it requires a substantial computational load, and is appropriate for many-to-many feature matching schemes such as object recognition. Therefore, it is not suitable for the SLAM scheme, which needs to use the minimum number of features to maintain real-time performance.

To deal with the above problems, the SLAM method using arbitrarily shaped (AS) features was proposed in our previous work.[16] To verify the performance of the proposed method, a number of experiments were performed in various environments, including an office, a corridor, and a house. Regardless of their shapes, AS features can be extracted at a place in which corner and line features do not exist, as various ceiling objects can be used. Furthermore, AS features provide better results in

*Corresponding author. Email: jbsong@korea.ac.kr

one-to-one matching than conventional region-based features because robust features with no neighboring similar features are used. The descriptor of an AS feature comprises the vertex distribution from the region contour, size, and orientation strength, and it is robust to illumination changes because no intensity information is involved. However, the vertices may be detected at different positions owing to noise variations in successive images, thus reducing the number of successful matches. In addition, a sufficient number of AS features cannot be extracted if similar object patterns are repeated on the ceiling.

In this study, enhanced extraction and matching methods for AS features using a probabilistic approach are proposed. Probabilities are assigned to store landmark vertices according to observational frequency when the robot moves. High probabilities are assigned to frequently observed vertices, which play a relatively large role during the feature matching process. On the other hand, low probabilities are assigned to rarely observed vertices, which are removed from the descriptor. If the number of robust features in the current image is not sufficient, unstable features that are surrounded by similar features are clustered and used as a robust feature. The entire SLAM process was performed based on the extended Kalman filter (EKF) framework, and the proposed method was verified by various experiments in real environments. Consequently, the number of successful matches for each landmark and the available robust AS features have increased, resulting in significantly more stable SLAM results compared to our previous work.[16]

The rest of the paper is organized as follows: Section 2 reviews the AS feature extraction method and Section 3 discusses the problems with the previous approaches. Section 4 deals with the enhanced feature matching and clustering schemes and Section 5 presents the EKF-based SLAM method along with experimental results in real environments. Finally, conclusions are drawn and future work is outlined in Section 6.

2. AS ceiling features

In our previous work, AS features were extracted based on contour detection results from ceiling images. The Canny edge detector [17] was adopted to detect the contours from a grayscale image, and the image was divided into several regions based on the contours shown in Figure 1. The regions were then eroded to remove undesirable connections between each region. The resulting regions were separated by a labeling process, and these were defined as regions of interest (ROIs).

Each ROI descriptor was defined using the vertex distribution from the region contour, size, and orientation strength, which were used to characterize ROIs and enable the matching of two ROIs against illumination changes. The vertices were detected using features from the accelerated segment test algorithm.[18] The position of vertex i was represented as $n_i=(r_i,\ \theta_i)$ in the polar coordinate, and it can be also denoted as $n_i=(u_i,\ v_i)$ in Cartesian coordinates, as shown in Figure 2(a). The ROI size, A, indicates the number of pixels in the region, and the orientation strength, R_{ori}, is obtained by

$$R_{\text{ori}} = \sigma_{m1}/\sigma_{m2} \quad (\sigma_{m1} \geq \sigma_{m2}) \tag{1}$$

where σ_{m1} and σ_{m2} are the lengths of the major and minor axes, respectively, which can be obtained from the pixel distribution using the covariance matrix.[16]

To prevent matching failure between the features stored in the feature map and those extracted from the current image, only unique ROIs that have no similar ROIs around them were selected. The similarities between two ROIs were evaluated by comparing their descriptors (i.e. vertex distribution, ROI size, and orientation strength). The similarity between vertex distributions was calculated by rotating the vertices of one ROI about the mean point $P_m(u_m,\ v_m)$ by 0–360° at intervals of 1°, because the ROIs may not align exactly. Vertex pairs were defined at each rotation angle if the distance of two vertices from each distribution was smaller than a predefined range R_m. The discrepancy between the distributions was then calculated using the minimum mean squared error as

$$D = \min_{0\leq\theta\leq 2\pi} \frac{1}{N} \sum_{(i,\ j)\in P} \{d(n_i,\ n_j)\}^2 \tag{2}$$

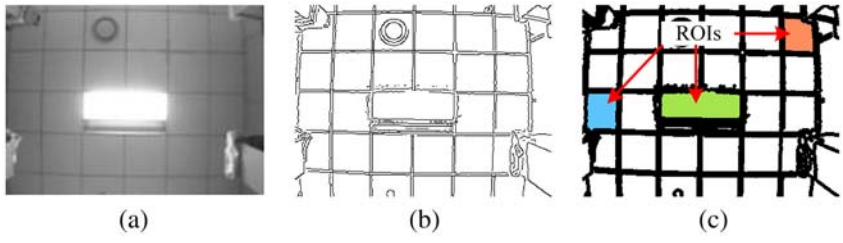

Figure 1. ROIs: (a) original image, (b) contour detection result from the Canny edge detector, and (c) example of ROI.

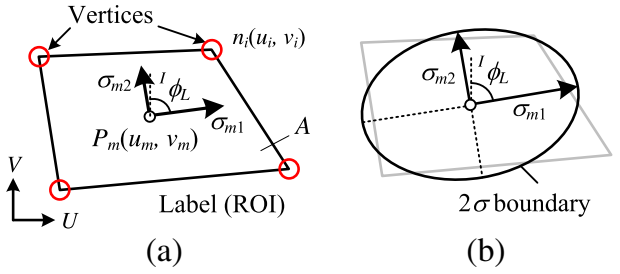

Figure 2. Descriptor for an AS region of interest: (a) vertices, size, and orientation of ROI, and (b) orientation determined from pixel distribution.

where D is the discrepancy, N is the total number of vertex pairs, P is the set of vertex pairs containing N pairs, $d(n_i, n_j)$ is the Euclidean distance between two vertices n_i and n_j. The similarity of vertex distributions, S_{ver}, is given by

$$S_{\text{ver}} = 1 - \frac{D}{R_m^2} \tag{3}$$

where $1/R_m^2$ is a normalizing factor, since the maximum value of D is R_m^2. Therefore, the range of S_{ver} becomes [0, 1] for every distribution.

The similarities of the size, S_{size}, and the orientation strength, S_{ori}, are calculated as

$$S_{\text{size}} = 1 - \frac{|A_a - A_b|}{\max(A_a, A_b)} \tag{4}$$

$$S_{\text{ori}} = 1 - \frac{|R_{\text{ori},a} - R_{\text{ori},b}|}{\max(R_{\text{ori},a}, R_{\text{ori},b})} \tag{5}$$

where A_a and A_b are the sizes of each ROI and $R_{\text{ori},a}$, and $R_{\text{ori},b}$ represents the orientation strength of each ROI. In this case, the size and orientation strength discrepancies are normalized based on their maximum values. Finally, the similarity between ROIs, S, is obtained by using (3), (4), and (5)

$$S = \frac{w_{\text{ver}} S_{\text{ver}} + w_{\text{size}} S_{\text{size}} + w_{\text{ori}} S_{\text{ori}}}{w_{\text{ver}} + w_{\text{size}} + w_{\text{ori}}} \tag{6}$$

where w_{ver}, w_{size}, and w_{ori} are the weights for the vertex distribution, size, and orientation strength, respectively.

To determine whether an ROI is unique, the similarities between all ROIs are calculated with regard to their distance. The uniqueness of each ROI is then calculated as

$$U(F_{\text{int}}) = \min_i \{1 - g(F_{\text{int}}, F_1) \times S(F_{\text{int}}, F_1), \ldots, 1 - g(F_{\text{int}}, F_k) \times S(F_{\text{int}}, F_k)\} \tag{7}$$

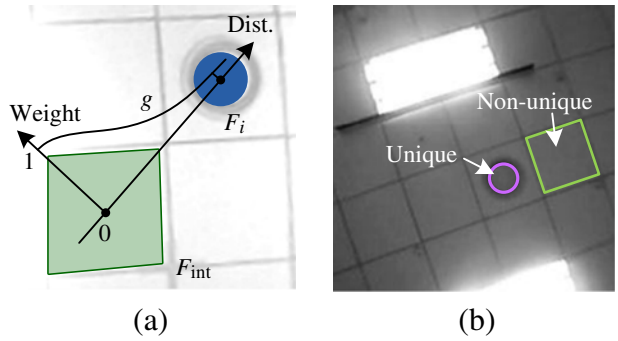

Figure 3. Calculation of uniqueness: (a) assignment of Gaussian weight to similarity and (b) unique and non-unique ROIs.

where F_{int} is the ROI of interest; F_i is the ith ROI, which is adjacent to F_{int}; g is the Gaussian weight function, which returns a large value when the distance between two ROIs is short; k is the number of ROIs around F_{int}; U is the uniqueness of the ROI; and S is the similarity of two ROIs defined by (6). According to (7), a ROI is highly unique when there are no similar ROIs nearby, and it is regarded as unique as shown in Figure 3(b). Finally, the unique ROIs are selected as AS features and used as a landmark in the EKF. Note that all AS features are assumed to be parallel to the floor, and features from slanting ceilings and walls are discarded during navigation since their shapes are deformed as the robot moves to different viewpoints.

3. Enhanced feature extraction and matching

3.1. *Clustering non-unique features*

The SLAM algorithm may suffer from a lack of features in some environments, even though unique ROIs are selected as features and provide reliable results. Those ROIs surrounded by similar ROIs are removed during the feature extraction process, so only a small number of features can be found in which non-unique ROIs are repeated. For example, if square tiles or circular lamps are placed within a short distance of each other, as shown in Figure 4, they are regarded as non-unique features and, thus, cannot be used as landmarks in the EKF.

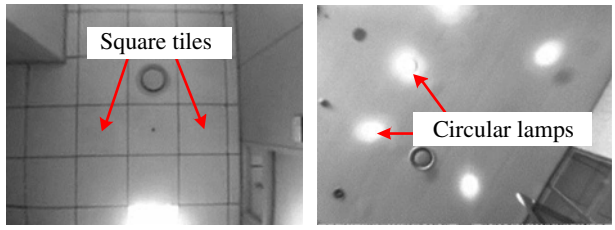

Figure 4. Repeated non-unique ROIs: examples in (a) an office environment and (b) a corridor.

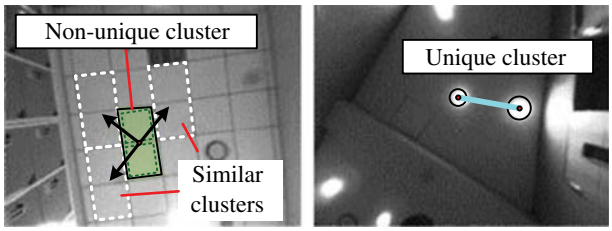

Figure 5. Clustering non-unique ROIs: examples of (a) non-unique and (b) unique clusters.

In this study, such non-unique ROIs were grouped and used as a unique cluster. A cluster is regarded as non-unique if there are similar ones nearby, as shown in Figure 5(a). Otherwise, a cluster with no similar ones nearby is regarded as unique, as shown in Figure 5(b). The similarity of adjacent ROIs is represented in the image coordinates using the 2D Gaussian distribution, as shown in Figure 6 and is given by

$$P_j(u,v) = \sum_{i=1}^{k} S_i \exp\left\{-\frac{1}{2}\left(\frac{(u-u_{m,i})^2}{\sigma_u^2} + \frac{(v-v_{m,i})^2}{\sigma_v^2}\right)\right\} \tag{8}$$

where P_j is the similarity between the current ROI, j, and its adjacent ROIs at an arbitrary point (u, v) in the image coordinates; S_i is the similarity of the adjacent ROI, i, obtained by (6); $(u_{m,i}, v_{m,i})$ is the mean point of the adjacent ROI; k is the number of adjacent ROIs; and σ_u and σ_v are the observation errors in the U and V axes, respectively. As a result, the similarity distribution represented in the image coordinates shows the location of similar ROIs near the current ROI.

To calculate the similarity between a cluster and its adjacent ROIs, the similarity distributions of each ROI from (8) are integrated as shown in Figure 7. If ROIs 1 and 2 have similar ROIs 1′ and 2′ in different directions and distances with respect to each ROI, as shown in Figure 7(a), the integrated similarity becomes low,

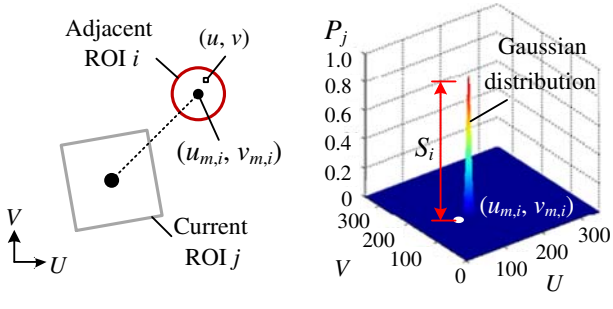

Figure 6. Similarity of adjacent ROIs represented by 2D Gaussian distribution.

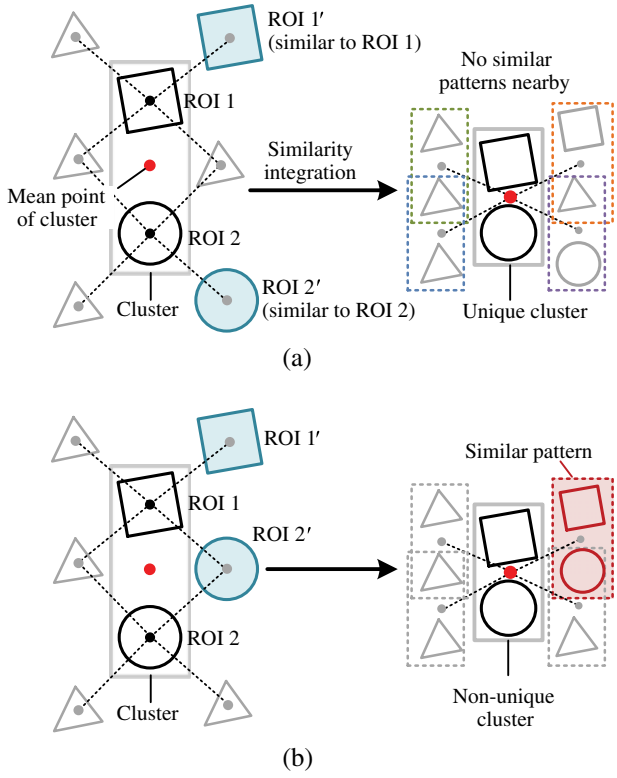

Figure 7. Calculation of similarity between clusters and adjacent ROIs: examples of (a) unique and (b) non-unique clusters.

indicating that there are no similar patterns nearby. On the other hand, if ROIs 1′ and 2′ are located in the same direction and distance with respect to ROI 1 and 2, respectively, the integrated similarity becomes high at the corresponding position, implying the existence of a similar pattern, as shown in Figure 7(b). The integrated similarity, P, at the point (u, v) is given by

$$P(u,v) = P_1(u,v)P_2(u,v)\ldots P_n(u,v) \tag{9}$$

where n is the number of ROIs in a cluster. Low integrated similarities at all points indicate that the cluster does not have similar patterns nearby, thus making it unique. The uniqueness of a cluster, U_C, is expressed by

$$U_C = 1 - \max_{(u,v)\in I} P(u,v) \tag{10}$$

where I is a set of pixel points for the entire image.

Figure 8 shows an example of calculating uniqueness using a 2D Gaussian representation. ROIs 1 and 2 have several similar ROIs nearby, but their similarities are significantly reduced through the integration process. The uniqueness, U_C, is then obtained from the point having a maximum similarity from (10). If U_C is larger than a predefined threshold, the cluster is considered unique and is regarded as an AS feature and used as a landmark

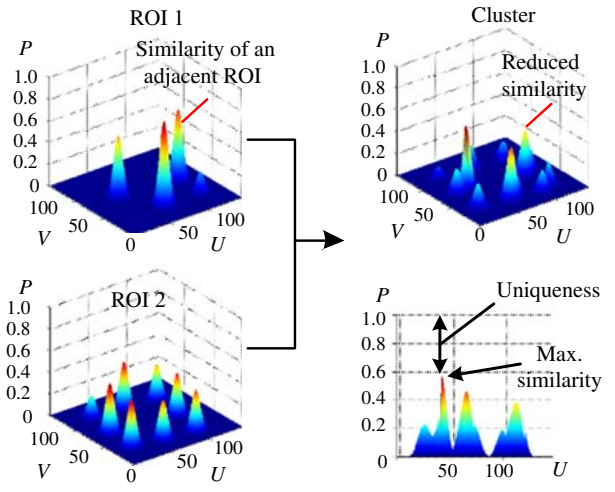

Figure 8. Integrating 2D Gaussian distributions to obtain the uniqueness of a cluster.

in the EKF. If a cluster is proven non-unique, a new ROI can be added to the cluster and the uniqueness is recalculated in the same way. Note that there is no limit to the number of ROIs in a cluster because the similarity distributions of two or more ROIs can be integrated using the proposed method.

The overall clustering scheme is represented in Figure 9. First, the unique features are registered, and then, the registration-restricted areas are set to maintain a minimum number of features in an image. After all unique features are registered, unique clusters can be found elsewhere in the image other than in the restricted areas by grouping the non-unique features. Consequently, the proposed clustering method increases the number of available AS features in an environment that lacks unique features.

3.2. *Vertex matching*

When an AS feature is registered to the EKF, its descriptor (i.e. vertex distribution, ROI size, and orientation strength) is also stored in the database for a posterior matching process. However, the vertices of the registered features are not always detected in the ceiling image owing to noise variations, whereas region size and orientation strength are reliably calculated. As shown in Figure 10, the shapes are partly deformed because noise affects the contour detection results. The vertices are then detected from different positions, leading to different matching results.

To deal with the problem of the vertices disappearing because of noise variations, Bayes' rule was adopted to keep only frequently observed vertices and remove rarely observed ones. A probability is assigned to each vertex in the stored descriptor and is updated according to Bayes' rule:

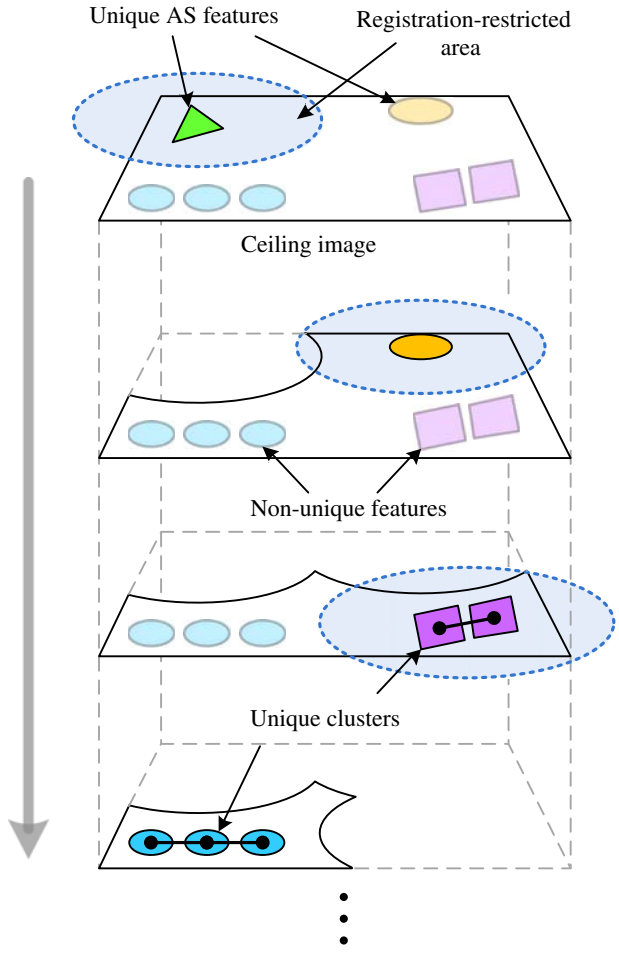

Figure 9. Overall clustering scheme.

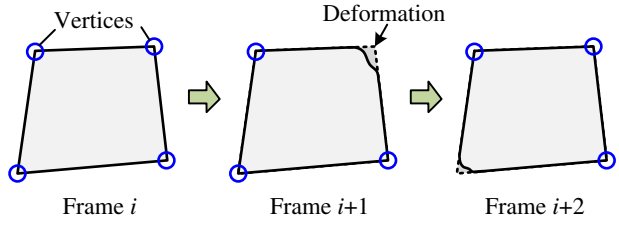

Figure 10. Vertex detection results according to shape deformation.

$$P(E|O_{t+1}) = \frac{P(O_{t+1}|E)P(E|O_t)}{P(O_{t+1}|E)P(E|O_t) + \{1 - P(O_{t+1}|E)\}\{1 - P(E|O_t)\}} \quad (11)$$

where E represents the existence of a vertex and O_t and O_{t+1} denote the observation of a vertex at times t and $t+1$, respectively. The initial probability for each vertex, $P(E|O)$, is set to 0.5, and the likelihood, $P(O|E)$, is experimentally obtained; for example, $P(O|E) = 0.7$ when the vertex is observed, and $P(O|E) = 0.2$ when it is not observed. Using (11), the probabilities of frequently

observed vertices gradually increase, and they remain in the database throughout navigation. On the other hand, the probabilities of rarely observed vertices decrease through iterations, and they are removed from the database if the probability becomes low, as shown in Figure 11(b). All probabilities are updated only when the robot moves in order to prevent artificial convergence or overconfidence reported in [19].

4. EKF-based slam

4.1. EKF

The distance to a landmark cannot be obtained in a single observation using only one monocular camera, so several observations at different viewpoints are required to estimate the position of the landmark. The initial uncertainty of a landmark pose (i.e. position and orientation) follows the Gaussian distribution, and its size is determined from the observation error of the camera and the maximum ceiling height. Sufficient observations reduce uncertainty, as shown in Figure 12, and landmarks with small uncertainties have a large effect on the estimated robot pose.

In this study, the EKF was adopted to deal with the relationship between the robot pose and the landmark poses and handle the nonlinearities involved in robot motion and sensor measurements. The state vector is defined as

$$X = [X_R, X_{L1}, \ldots, X_{Ln}, X_{LO1}, \ldots, X_{LOm}]^T \tag{12}$$

$$X_R = [{}^G x_R, {}^G y_R, {}^G \theta_R] \tag{13}$$

$$X_L = [{}^G x_L, {}^G y_L, {}^G z_L] \tag{14}$$

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

where X_R represents the robot pose, X_L denotes the landmark position, and X_{LO} is the landmark position with orientation. First, the state vector and its covariance matrix are predicted as follows:

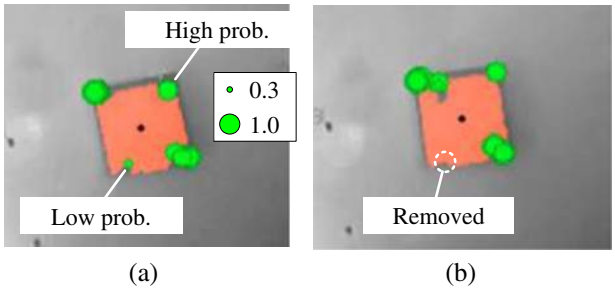

Figure 11. Assigning probabilities to vertices: (a) vertices from square lamp, and (b) removal of rarely observed vertices.

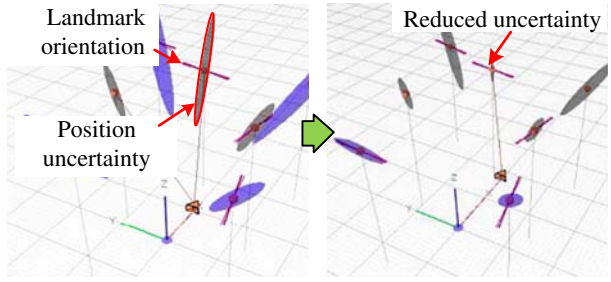

Figure 12. Basic concept of monocular vision-based SLAM.

$$\hat{X}_t^- = f(\hat{X}_{t-1}, u_t) \tag{16}$$

$$P_t^- = \nabla F_x P_{t-1} \nabla F_x^T + \nabla F_u Q \nabla F_u^T \tag{17}$$

where f is a function of the odometric position update for a differential drive robot, the input u_t denotes the distances traveled by the right and left wheels between time $t-1$ and t, P is the covariance matrix of the state vector, Q is the process noise, and $\nabla F_x = \partial f / \partial X$ and $\nabla F_u = \partial f / \partial u$ are the Jacobian matrices. Note that the superscript '$-$' shows the predicted state before the update step.

The predicted state vector and covariance matrix are then updated from successive observations. The relationship between the sensor and global frames shown in Figure 13 is defined as follows:

$$\hat{Z}_t = h(\hat{X}_t^-) \tag{18}$$

where $\hat{Z}_t$ represents the predicted poses of landmarks in the sensor frame at time t based on the predicted state vector. The observation model, h, is given by

$$H = \begin{bmatrix} {}^I r_L \\ {}^I \theta_L \\ {}^I \phi_L \end{bmatrix} = \begin{bmatrix} \sqrt{({}^G x_L - {}^G x_R)^2 + ({}^G y_L - {}^G y_R)^2} \times \frac{f_{\text{cam}}}{{}^G z_L} \\ \tan^{-1}\left(\frac{{}^G y_L - {}^G y_R}{{}^G x_L - {}^G x_R}\right) - {}^G \theta_R \\ {}^G \phi_L - {}^G \theta_R \end{bmatrix} \tag{19}$$

where f_{cam} is the focal length of the camera from the calibration process and $[{}^I r_L, {}^I \theta_L, {}^I \phi_L]^T$ is the position and orientation of a landmark in the sensor frame. If a landmark does not have an orientation, $h = [{}^I r_L, {}^I \theta_L]^T$ is used instead. Finally, the state vector and its covariance matrix are updated using the Cholesky decomposition to reduce the computational load.[20]

$$\Psi_t = H_t P_t^- H_t^T + R_t = U^T U \tag{20}$$

$$K_t = P_t^- H_t^T (U^T U)^{-1} \tag{21}$$

$$\hat{X}_t = \hat{X}_t^- + K_t (Z_t - \hat{Z}_t) \tag{22}$$

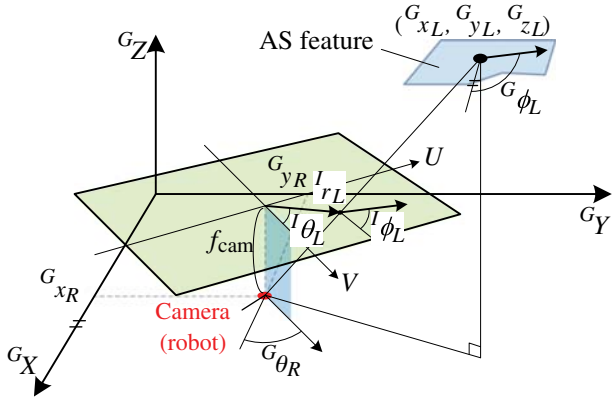

Figure 13. Relationship between the sensor and global frames.

$$P_t = (I - K_t H_t) P_t^- \tag{23}$$

where U is the upper triangular matrix obtained from the decomposition, K is the Kalman gain, $H = \partial h / \partial X$ is the Jacobian matrix, and R is the sensor noise.

4.2. Experimental results

Several experiments were performed using a Pioneer 3-DX (MobileRobots) mobile robot equipped with an upward-looking camera having a 120° field of view, as shown in Figure 14. A notebook computer with a Core i7 1.6 GHz CPU was used to run the proposed algorithm.

The experimental results are represented in Figure 15. Two AS features were extracted from an air vent and a rectangular tile at the initial navigation state shown in Figure 15(a). The uncertainties of the landmark positions are depicted as a 3D Gaussian distribution. A unique cluster was found near the rectangular lamp, as shown in Figure 15(b). Frequently observed vertices are represented as relatively large circles in Figure 15(c), and rarely observed vertices were removed from the descriptor. The robot pose was accurately estimated based on landmarks with small uncertainties, as shown in Figure 15(d). The SLAM was successfully performed,

Figure 14. Mobile platform and experimental environment.

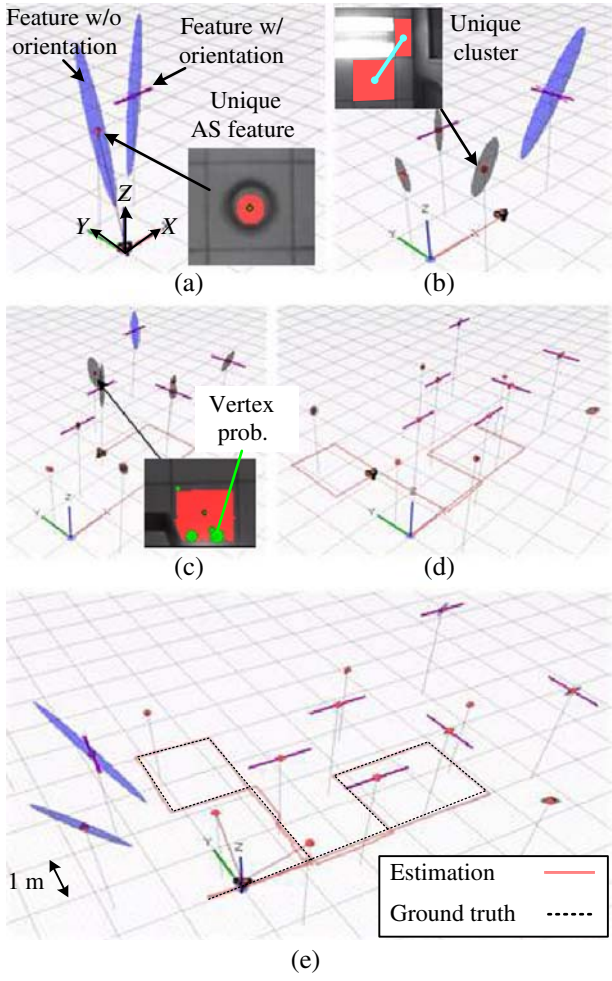

Figure 15. Experimental results in a room of an office: (a) initial state, (b) ROI cluster, (c) vertex probabilities, (d) converged uncertainties, and (e) final results.

and the estimation results are compared with the ground truth in Figure 15(e).

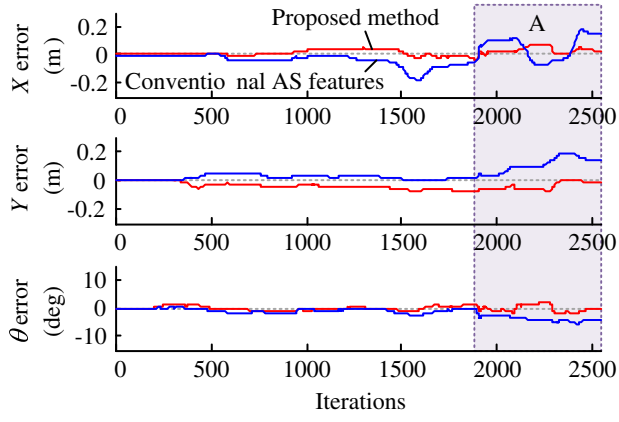

Figure 16. Robot pose errors over 2500 iterations.

Table 1. Computation time for each step.

Step	Elapsed time (ms)
Preprocessing (image acquisition)	10
AS feature extraction	16
AS feature clustering	4
EKF (30 landmarks)	56
Total	86

The robot pose error variations over 2500 iterations are represented in Figure 16. The pose error from the SLAM method using the proposed method shows smaller variations than that using the conventional AS features. The number of successful matches increases on assigning probabilities to the vertices, resulting in a reduction in pose errors. Region A in Figure 16 represents the pose error from the environment in which non-unique ROIs are repeated. The robot pose was accurately estimated because the unique clusters were successfully extracted from the non-unique ROIs. The computation time of each step in Table 1 is so small that the entire SLAM process occurs in real-time.

5. Conclusion and future work

This paper presented enhanced extraction and matching methods for (AS) features. Non-unique ROIs were clustered and their uniqueness was measured using pre-calculated similarities between the ROIs. Unique clusters were also used with the unique ROIs as landmarks in the EKF. In addition, a probability was assigned to each stored vertex, and only frequently observed vertices were selected. The proposed schemes were validated by several experiments, and the following conclusions were drawn:

(1) The robot pose was accurately estimated using both conventional AS features and the unique ROI clusters, although the non-unique AS features were repeated on the ceiling.
(2) The unique ROI clusters enabled stable SLAM in an environment containing many non-unique ROIs and there was no limit on the number of ROIs in a cluster.
(3) The similarities between ROIs were reused to calculate the uniqueness of each cluster. Hence, the computational burden of the proposed scheme is low.
(4) AS features can be matched in successive images by assigning a probability to each vertex, even if noise exists in the image.

However, the proposed scheme works well with ceiling features that are parallel to the floor. Future work will include dealing with AS features from any region, including walls, by taking geometric relationships into consideration.

Acknowledgment

This research was supported by the Intelligent Robotics Development Program (Ministry of Knowledge Economy) and by Human Resources Development Program for Convergence Robot Specialists (Ministry of Knowledge Economy) (NIPA-2012-H1502-12-1002).

Notes on contributors

Seo-Yeon Hwang received the BS degree in mechanical engineering from Korea University, Seoul, Korea, in 2006. He was awarded his PhD degree from Korea University in 2013. Dr Hwang is currently a post-doctoral fellow in Mechanical Engineering at Korea University. His research interests include mobile robot navigation and vision-based techniques.

Jae-Bok Song received his BS and MS degree in mechanical engineering from Seoul National University in 1983 and 1985, respectively. He was awarded his PhD degree from M.I.T. in 1992. He is currently a professor at the School of Mechanical Engineering at Korea University. He has served as a director of Intelligent Robotics Laboratory from 1993. His research interests include safe manipulators, design and control of robotic systems, and indoor/outdoor navigation.

References

[1] Pathirana PN, Savkinb AV, Ekanayake SW, Bauer NJ. A robust solution to the stereo-vision-based simultaneous localization and mapping problem with steady and moving landmarks. Adv. Rob. 2011;25:765–788.
[2] Davison A, Reid I, Molton N, Stasse O. Mono SLAM: real-time single camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. 2007;29:1052–1067.
[3] Se S, Lowe D, Little J. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. Int. J. Rob. Res. 2002;8:735–758.
[4] Hwang SY, Song JB. Upward monocular camera based SLAM using corner and door features. In: Proceedings of the 17th world congress of IFAC; 2008. p. 1663–1668.
[5] Jeong WY, Lee KM. Visual SLAM with line and corner features. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems; 2006. p. 2570–2575.
[6] Choi H, Kim DY, Hwang JP, Park CW, Kim E. Efficient simultaneous localization and mapping based on ceiling-view: ceiling boundary feature map approach. Adv. Rob. 2012;26:653–671.
[7] Lee S, Lee S, Baek SM. Vision-based kidnap recovery with SLAM for home cleaning robots. J. Intell. Rob. Syst. 2012;67:7–24.

[8] Lemaire T, Lacroix S, Sola J. A practical 3D bearing-only SLAM algorithm. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems; 2005. p. 2449–2454.

[9] Smith P, Reid I, Davison A. Real-time monocular SLAM with straight lines. In: Proceedings of the 17th British Machine Vision Conference; 2006.

[10] Lemaire T, Lacroix S. Monocular-vision based SLAM using line segments. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2007. p. 2791–2796.

[11] Folkesson J, Jensfelt P, Christensen HI. Visual SLAM in the measurement subspace. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2005. p. 30–35.

[12] Matas J, Chum O, Urban M, Pajdla T. Robust wide baseline stereo from maximally stable extremal regions. In: Proceedings of the 13th British Machine Vision Conference; 2002. p. 384–393.

[13] Kadir T, Zisserman A, Brady M. An affine invariant salient region detector. In: Proceedings of the European Conference on Computer Vision; 2004. p. 345–457.

[14] Kunze L, Lingemann K, Nuchter A, Hertzberg J. Salient visual features to help close the loop in 6D SLAM. In: Proceedings of the Conference on Computer Vision Systems; 2007. p. 1–10.

[15] Ramisa A, Tapus A, Aldavert D, Toledo R, Lopez R. Robust vision-based robot localization using combinations of local feature region detectors. Auton. Rob. 2009;27:373–385.

[16] Hwang SY, Song JB, Kim MS. Robust extraction of arbitrary-shaped features in ceiling for upward-looking camera-based SLAM. In: Proceedings of the 18th world congress of IFAC; 2011. p. 8165–8170.

[17] Canny J. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. 1986;8:679–714.

[18] Rosten E, Drummond T. Machine learning for high-speed corner detection. In: Proceedings of European Conference on Computer Vision; 2006.

[19] Eade E, Drummond T. Scalable monocular SLAM. In Proceedings of the Conference on Computer Vision and, Pattern Recognition; 2006. p. 469–476.

[20] Eubank RL. A Kalman filter primer. Chapman and Hall/CRC; 2005. p. 169–175.