CV-SLAM: A new Ceiling Vision-based SLAM technique

WooYeon Jeong
School of Radio Science and Communications
Hongik University
72-1, Sangsu-dong, Mapo-gu Seoul, 121-791, Korea
wota@cvlab.snu.ac.kr

Kyoung Mu Lee
School of Electrical Engineering and Computer Science
Seoul National University
San 56-1, Sillim-dong, Gwanak-gu, Seoul, 151-742, Korea
kyoungmu@snu.ac.kr

***Abstract* - We propose a fast and robust CV-SLAM (Ceiling Vision –based Simultaneous Localization and Mapping) technique using a single ceiling vision sensor. The proposed algorithm is suitable for system that demands very high localization accuracy such as an intelligent robot vacuum cleaner. A single camera looking upward direction (called ceiling vision system) is mounted on the robot, and salient image features are detected and tracked through the image sequence. Compared with the conventional frontal view systems, the ceiling vision has advantage in tracking, since it involves only rotation and affine transform without scale change. And, in this paper, we solve the rotation and affine transform problems using 3D gradient orientation estimation method and multi-view description of landmarks. By applying these methods to the solution for data association, we can reconstruct the 3D landmark map in real-time through the Extend Kalman filter based SLAM framework. Furthermore, relocation problem is solved efficiently by using a wide base line matching between the reconstructed 3D map and a 2D ceiling image. Experimental results demonstrate the accuracy and robustness of the proposed algorithm in real environments.**

Index Terms – SLAM, Ceiling Vision, data association

I. Introduction

Self-localization and map building are the most important problems in mobile robot system, and have been the central research topics in the robotic society for several decades. In most recent works, it is argued that the self localization and mapping problems could not be separated and have to be dealt simultaneously [1][2][3][8][9]. This is called SLAM (Simultaneous Localization and Mapping) problem. The purpose of SLAM is to minimize the localization and mapping error simultaneously, and it has been proved that the only constrain for the SLAM convergence is the perfect data association [2]. Conventionally, most SLAM algorithms have employed active range sensors such as laser scanner or sonar for data association. Especially laser range finder (Lidar) based SLAM has been applied for very wide range of robot applications in indoor [9] and outdoor [8] environment. However, due to the high cost, speed, accuracy and safety problem, these active sensors-based SLAM systems have limitations in practical applications. Moreover, since these sensors usually provide not enough unary information of landmarks, lots of multiple measurements should be combined to solve the relocation problem.

Recently, in order to overcome the drawbacks of using active range finders, some works have been proposed to use vision sensors for localization and mapping [1][3][10][11][12][14].

Jogan et al. [10] proposed an appearance based localization method using omni-directional camera. Appearance matching was carried out in the eigen-space of trained images, and to cope with occlusions robust- PCA technique was employed.

Lowe [3] introduced triclops-vision system that used their own wide baseline matching technique; SIFT (Scale Invariant Feature Transform) [4]. It maintains robot pose and landmark position separately.

Similarly, Kosecka et. al. [11] built a topological map by tracking SIFT features of frontal view images, and enhanced the localization performance using Hidden Markov Model.

Wolf et. al. [12] suggested image retrieval based localization technique by using Monte-Carlo localization method. This method could retrieve target image in spite of large camera motion, and minimize the location uncertainty using multi-hypothesis.

More recently, Davison [1] proposed a vision-based real-time SLAM, called Mono-SLAM, which employs only a single camera without odometry information. It increases localization accuracy by integrating camera velocity into optimization variables. However, it needs an initial manual calibration process to obtain the scale information.

In this paper, we propose a very fast and accurate SLAM system called CV-SLAM that uses a ceiling vision system that consists of a single camera pointing upward direction. We suggest an efficient data association method using view-invariant feature matching technique appropriate for ceiling vision, and we also develop a fast robot relocation technique.

II. EKF based SLAM

The solution to SLAM problem always converges if successful data association is guaranteed [2], and a lot of research over the last decade has shown that SLAM is indeed possible without any priori knowledge of a map. The basic EK-based SLAM can be formulated by following equations.

$$\mathbf{x}(k) = F(\mathbf{x}(k-1), \mathbf{u}(k)) + \mathbf{v}(k) \qquad (1)$$

$$\mathbf{z}_i(k) = h_i(\mathbf{x}(k)) + \mathbf{w}_i(k) \qquad (2)$$

Equation (1) is the state transition model, where $F(\cdot)$ models robot kinematics, $\mathbf{u}(\cdot)$ is the control input, and $\mathbf{v}(\cdot)$ is the motion noise. Equation (2) is the observation model, where $h_i(\cdot)$ is an observation function that projects the i-th landmark to the observed measurement $\mathbf{z}_\mathrm{i}(\mathrm{k})$, and $\mathbf{w}_i(\cdot)$ is the measurement noise. $\mathbf{x}(k)$ is the state vector to be optimized. By placing both robot and landmark position on the state vector at the same time as follows, we can minimize localization and mapping error simultaneously [2].

$$\mathbf{x}(k) = [\mathbf{R}(k) \quad \mathbf{L}_0 \quad \mathbf{L}_1 \quad \cdots \quad \mathbf{L}_N]^T,$$

where the robot pose $\mathbf{R}(k) = [R_x(k), R_y(k), R_\theta(k)]^{\mathrm{T}}$, the landmark position $\mathbf{L}_{\mathrm{i}} = [L^i_x, L^i_y, L^i_z]^{\mathrm{T}}$,and N is the number of landmarks.

A. Kinematics model

In this paper, we consider a two wheel based robot system on which a ceiling vision camera is mounted. So, its kinematics model can be described as follows.

$$\begin{bmatrix} R_x(k+1) \\ R_y(k+1) \\ R_\theta(k+1) \end{bmatrix} = \begin{bmatrix} R_x(k) + u_r(k)\cos u_\theta(k) \\ R_y(k) + u_r(k)\sin u_\theta(k) \\ R_\theta(k) + u_\theta(k) \end{bmatrix} = F(\mathbf{x}(k), \mathbf{u}(k)), \quad (3)$$

where the control input $\mathbf{u}(k) = [u_r(k)\ u_\theta(k)]^{\mathrm{T}}$, and where $u_r(k)$ is the radial distance and $u_\theta(k)$ is the angle.

B. Observation Model

Observation model is the projection function that projects a 3D landmark to the sensor observation. Our ceiling vision system has a camera positioned at the center of the robot, aligned with the robot orientation. Thus, as shown in Fig. 1 3D landmarks are projected onto the 2D image plane by the following observation model.

$$\mathbf{z}_i(k) = \mathbf{h}_i(\mathbf{x}(k)) = \begin{bmatrix} z^i_r \\ z^i_\theta \end{bmatrix} \quad (4)$$

$$= \begin{bmatrix} \sqrt{\left(L^i_x - R_x(k)\right)^2 + \left(L^i_y - R_y(k)\right)^2} \times \dfrac{f}{L_z} \\ \tan^{-1}\dfrac{L^i_y - R_y(k)}{L^i_x - R_x(k)} - R_\theta(k) \end{bmatrix},$$

where f denotes the focal length of the camera.

III. Visual Data Association using Ceiling Vision

As mentioned in previous section, the only constrain of SLAM solution is the validity of data association. From the vision sensor's point of view, successful data association means successful correspondence establishment across multi view images, which is called view-invariant or wide base line matching.

A. Ceiling Vision

When comparing two images acquired from quite different view points, the perspective distortion usually makes the correspondence problem extremely difficult. There have been a lot of research on view-invariant matching under the rotation, scale and affine transformations [3][5][6]. Note that one of the special characteristics of ceiling vision compared to the general camera setting is that no scale change occurs between ceiling images, but only rotation and sheer deformation exist for planar landmark patches. Fig. 2 shows an example of this scale invariant property of ceiling vision. Patch **A** is a ceiling region that has only rotation change between views, while **B** that includes a wall area exhibits both rotation and sheer transformations. By this scale invariant property, we can not only save a lot of searching time for finding the scale of each landmark, but also achieve matching with high localization accuracy.

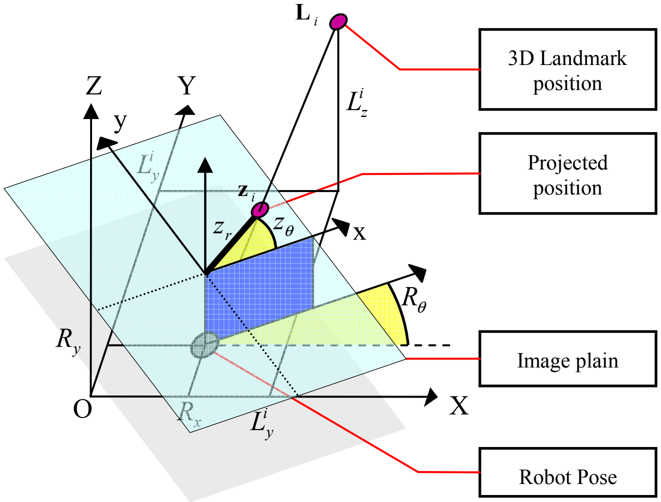


Fig. 1 Observation Model: 3D to 2D projection

B. POI (Point of Interest) Selection

Most of view-invariant feature matching techniques try to find POI first, and then compare its local regions through image correlation or their own invariant descriptor-based matching methods. The purpose of this POI detector is to reduce the searching space by comparing only selected candidate region instead of full image search. SIFT (Scale Invariant Feature Transform) [4] is known to be the state-of-the-art for the wide-base line matching. However, although it establishes very robust correspondences under scale and affine variations, the localization accuracy of each POI is relatively low. And, especially for the case of no scale change between views as in the ceiling images, it is not appropriate to use SIFT. Thus, in this paper, in order to achieve very accurate landmark association between ceiling views, we employ Harris corner detector for POI detection. [7]

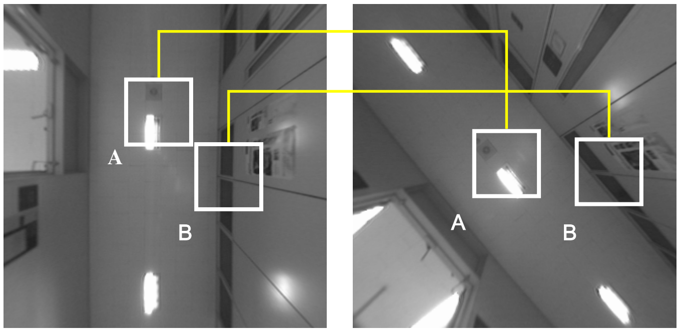


Fig. 2 Scale invariant property of ceiling image; A has rotation variation ; B has rotation and sheer deformation

C. Estimation of Landmark Orientation

1) 2D Gradient Orientation Estimation: If there exists only 2D rotation variation between corresponding landmarks in different views, matching can be done easily after finding and aligning their own orientations. The orientation of a landmark can be determined effectively by the gradient estimation technique similar to SIFT [4], in which the local gradient orientation histogram is used for finding the orientation of a patch. The gradient magnitude m and the orientation θ of each pixel in the landmark region (local neighborhood of the POI) is calculated by

$$\begin{array}{ll} m = \sqrt{dI_x^2 + dI_y^2} & dI_x = I_{x+1,y} - I_{x-1,y} \\ \theta = \tan^{-1}\left(dI_y / dI_x\right) & dI_y = I_{x,y+1} - I_{x,y-1} \end{array} , \qquad (5)$$

where, I_{xy} is the intensity value at (x, y) position. Then the orientation histogram weighted by both its magnitude and a weighting mask can be constructed. In SIFT, the Gaussian mask is used to make the orientations around the center to be highly weighted. However, since we are using the corner points as POIs, around which the gradient orientations usually become unstable, applying the Gaussian mask is not appropriate in our case. Therefore, instead we attenuate the unreliable central part by using a donut like Gaussian mask as follows.

$$G(r) = \begin{cases} \dfrac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}} & (r > threshold) \\ 0 & (r \le threshold) \end{cases} \qquad (6)$$

Now, the unique orientation of a POI can be determined by selecting multiple peaks in the histogram that satisfy some constraints [4]. Fig. 3 shows the detected corner points and their estimated orientations, and this orientation information can be used for 2-D rotation-invariant feature matching.

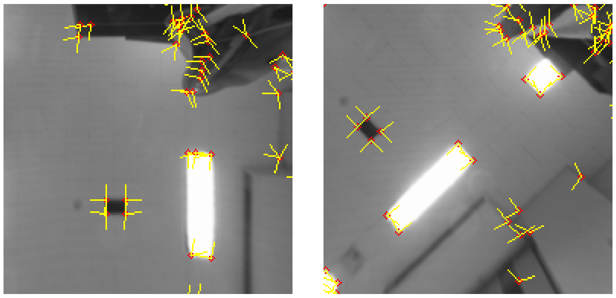
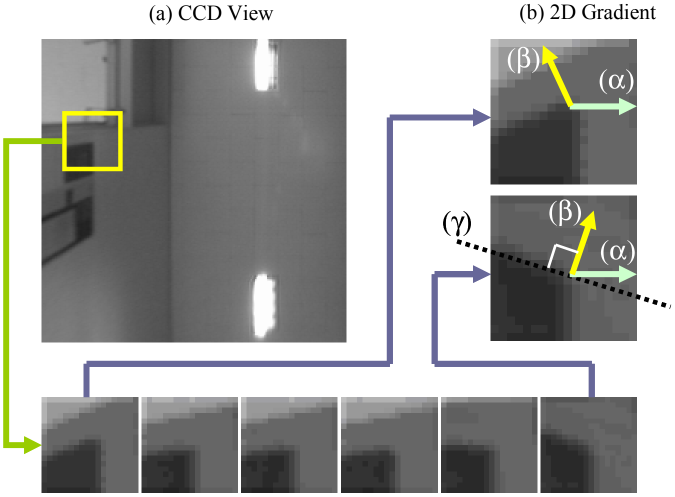


Fig. 3 Corner points and their estimated orientations in views with rotation only


(a) CCD View
(b) 2D Gradient

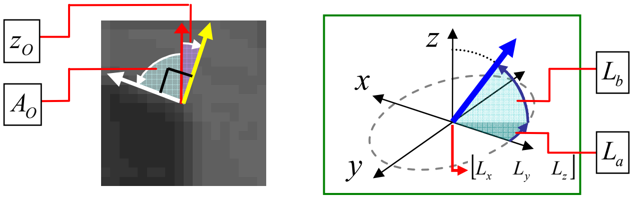
(c) Deformation of a landmark by the robot motion

Fig. 4 Gradient orientation variation by the robot movement

2) 3D Gradient Orientation Estimation: Note that if the robot moves, the landmarks on the region parallel to the image plane undergo Euclidian transform, while those on the other planes are deformed by affine (shear) transform. Thus the 2D gradient orientation estimation technique can be applied only to the landmarks on the ceiling, but not on walls. Fig. 4 shows the view variations of a landmark on a wall while the robot was in translational motion. Fig. 4(b) shows the estimated gradient orientations of the first and the last landmarks in Fig. 4(c). We can observe that the gradient orientation vector (β) becomes different even though there is no rotational movement. Note that we can not recover the corresponding 3D gradient vector using the projected 2D gradient orientation vectors uniquely. However, it is possible to reconstruct its normal vector (the tangential orientation vector) in 3D space by back projecting the 2D measurements as in Fig. 4(γ). By simply adding right angle to the 2D gradient orientation we can obtain 2D tangential vector. Now the remained problem is how to reconstruct 3D tangential vector from multiple measurement of 2D tangential vectors. As shown in Fig. 5, the 3D to 2D vector projection function is non-linear, so we use EKF again to solve this problem using (1) and (2). In this case, the state transition function $F(\cdot)$ becomes an identity matrix, and the state vector is $\mathbf{x}(k) = [L_a, L_b]^{\mathrm{T}}$, where L_a and L_b are the azimuth and elevation of the 3D orientation vector. And, the observation function $h_i(\cdot)$ is given by

$$A_O = z_O + 90 = \tan^{-1}\left(\frac{L_z \sin L_a \cos L_b + \left(L_y - R_y\right)\sin L_b}{L_z \cos L_a \cos L_b + \left(L_x - R_x\right)\sin L_b}\right) - R_\theta \qquad (7)$$

where z_O is the 2D gradient orientation angle and A_O is the angle of 2D tangential vector. Overall process of 3D tangential vector reconstruction is shown in Fig. 6. When landmark is initially registered, its 3D tangential vector is roughly estimated first with appropriate covariance, and then gradually updated. As the sequential update proceeds, its covariance



2D orientation on POI

3D tangential orientation on landmark

Fig. 5 Observation Model of projecting 3D orientation to 2D

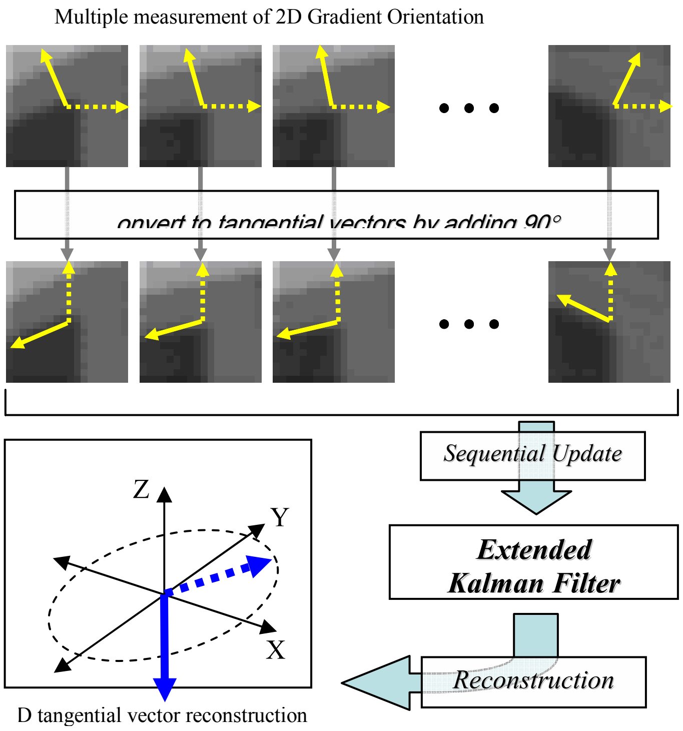

Multiple measurement of 2D Gradient Orientation
onvert to tangential vectors by adding 90°
Sequential Update
Extended Kalman Filter
Reconstruction
Z
Y
X
D tangential vector reconstruction

Fig. 6 3D tangential vector reconstruction using multiple 2D gradient orientation measurement

decreases and the state vector converges. The estimated 3D orientation information of each landmark can be used for the landmark matching between views and more effectively for the robot relocation problem as described in section in IV.

E. Multi-view description of Landmarks

As the landmark tracking progress, its image pattern becomes more and more differ from the initial one that was trained at the registration time. Molton [13] solved this problem by estimating the surface normal using the inverse compositional image alignment technique, but it was not robust in noisy environment. We solve this problem by training all image patches acquired from all possible view positions into a finite number of classes. The flow chart of the landmark multi-view description scheme is shown in Fig. 7. Simultaneous matching and training process is done by a double thresholding technique. The first threshold is used to determine whether the current landmark is similar to the previous one or not, and the second threshold is used for making decision whether the current landmark have to be trained as a new pattern or not. This multi-view description scheme can cope with any kind of deformation even if landmark is not locally planner, and works very fast and robustly in real environment. This method is similar to [14], but our method is different in that it trains not only the image patterns but also their corresponding robot positions.

IV. Robot Relocation

Relocation can be performed by matching the current image features to the map of the reconstructed 3D landmarks.

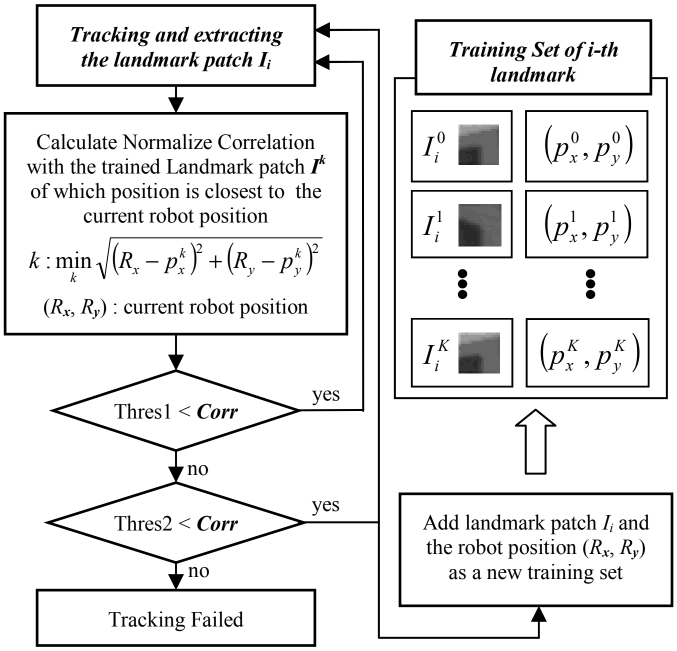
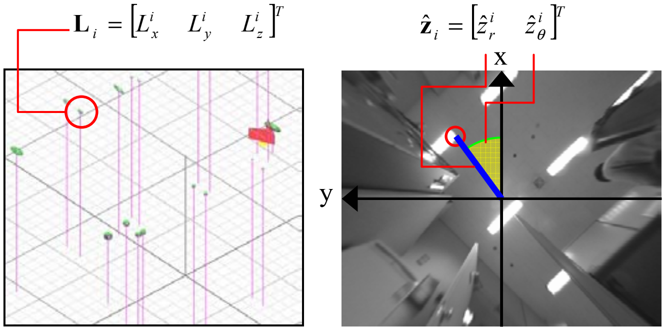

Tracking and extracting the landmark patch I_i
Calculate Normalize Correlation with the trained Landmark patch I^k of which position is closest to the current robot position
$k:\min_k \sqrt{(R_x-p_x^k)^2+(R_y-p_y^k)^2}$
(R_x, R_y) : current robot position
Thres1 < Corr
yes
no
Thres2 < Corr
yes
no
Tracking Failed
Training Set of i-th landmark
I_i^0
(p_x^0, p_y^0)
I_i^1
(p_x^1, p_y^1)
I_i^K
(p_x^K, p_y^K)
Add landmark patch I_i and the robot position (R_x, R_y) as a new training set

Fig. 7 Flow chart of the generation of multi-view descriptor for a landmark

Note that given one correspondence between an image feature $\hat{\mathbf{z}}_i$ and a landmark $\mathbf{L}_i$, we can deduce the possible current robot positions by the observation model in (4) as follows.


$\mathbf{L}_i=[L_x^i \quad L_y^i \quad L_z^i]^T$
$\hat{\mathbf{z}}_i=[\hat{z}_r^i \quad \hat{z}_\theta^i]^T$
x
y

Landmark in 3D Map

Feature measurement in 2D Image

$$\begin{bmatrix}R_x\\R_y\end{bmatrix}=\begin{bmatrix}G_r\cos G_\theta+L_x^i\\G_r\sin G_\theta+L_y^i\end{bmatrix}\qquad G_r=\hat{z}_r^i\times L_z^i/f \qquad (8)$$

Fig. 8 Correspondence matching between 3D Map and 2D Image

where G_r and G_θ are the distance and angle between the robot center and the projected landmark position onto the ground plane. Since we can not determine G_θ from this correspondence, the solution will be the locus of circle with radius G_r. However, with multiple correspondence pairs, we can estimate the true robot position using the Hough clustering technique. Since each correspondence provides a circle in the Hough domain, we can determine the robot position by finding the majority vote. Fig. 9 shows an example of Hough clustering, in which the solid circles represent the correct correspondences and the dashed ones show wrong matches. However, due to the speed and memory problem, the naïve Hough clustering technique can not be used for real-time

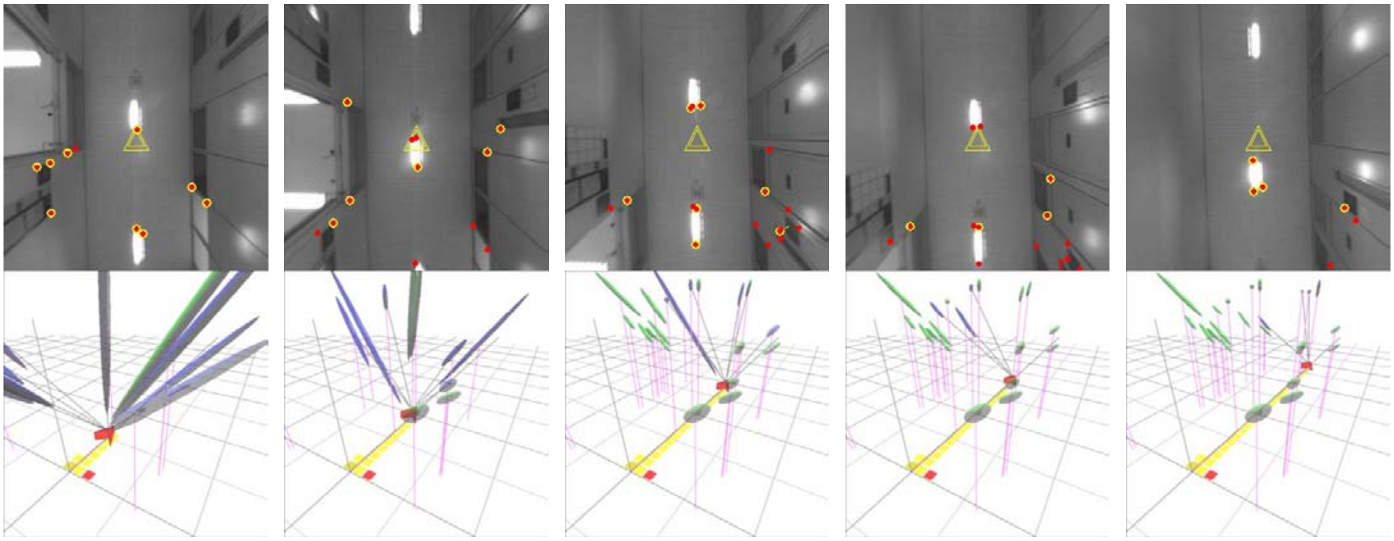


Fig. 10 Real time sequential map building experiment: corridor
Video is available from http://cv.snu.ac.kr/cvslam/

processing. Thus, we modified it by dividing 3D landmarks into two groups according to their orientations. The first group is for the landmarks with horizontal orientation direction, i.e., parallel to the image plane, and the other one is for ones with non-horizontal orientation directions. If we know a certain correspondence of a landmark that has horizontal orientation vector, we can obtain G_θ and the robot orientation angle R_θ by

$$G_\theta = \hat{z}^i_\theta - \left(L^i_a - \left(\hat{z}^i_0 + 90^o\right)\right) \tag{9}$$

and by placing $L^i{}_b$ to zero in (7), we can also determine the robot orientation angle by

$$R_\theta = L^i_b - \left(\hat{z}^i_0 + 90^o\right). \tag{10}$$

From this equation, one correspondence can be mapped to one unique robot pose. Thus, a finite number of initial pose candidates can be obtained using the first group. And then, a verification procedure is followed by the other group. The proposed relocation process is summarized in the following:

- Find all possible correspondences with high correlation
- Draw all possible robot pose using the correspondences of horizontal direction landmarks, and find some pose candidates by the majority vote rule.
- As the final robot pose, select the pose among the candidates that are maximally supported by the other correspondences with non-horizontal directions as the final robot pose.

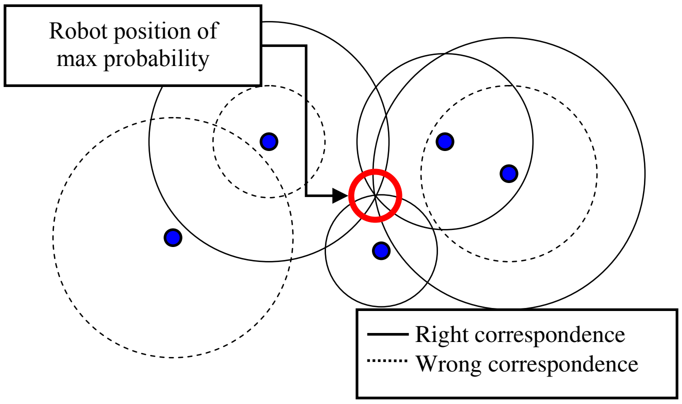
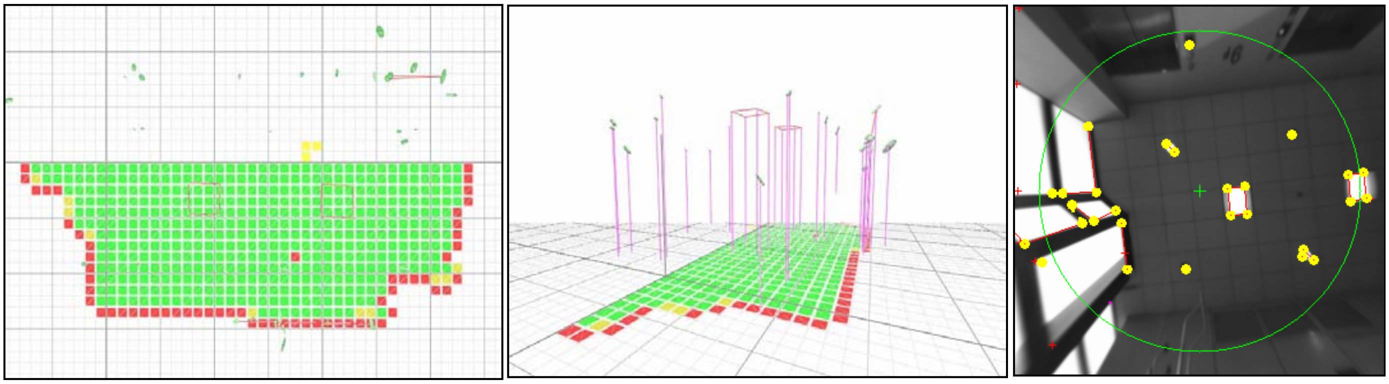


Fig. 9 Example of Hough clustering

The speed of the proposed relocation method depends on the map size (the number of landmarks). For about 100 landmarks, the processing time is less than 300 msec.

V. Experimental Results

The proposed algorithm was applied to a robot vacuum cleaner. Maximum speed of the robot was 15cm/sec. We used an NTSC 1/3 inch CCD camera, wide angle lens of 150 degree, and a Pentium III 1 GHz CPU. 320×320 resolution for each view was used after lens calibration, and the size of each landmark patch was set to be 20×20. We have tested our system in many different real indoor environments including laboratory, corridor, lobby, and apartment living rooms. The test was carried out in two steps. The first test was for the real time 3D landmark map-building by the proposed CV-SLAM, and the second test was the zigzag motion test and floor-map building for vacuum cleaning using the ready made 3D landmark map. All experiment was accomplished in real time and all procedures were performed automatically except the motion command for the 3D map-building step. Fig. 10 shows an example of real time map convergence. As the robot moved on, the uncertainties of landmarks monotonically decreased. Note that due to the multi-view description, landmarks on the walls also have converged quickly. Fig. 11 shows the constructed 3D landmark map and the partial floor sweeping result by the robot's zigzag motion. The green blocks represent the space that robot has passed and the red block denote the obstacles, and each block size is 20×20 cm^2. Table 1 shows the elapsed time of each stage of the proposed algorithm for 201 landmarks. Note that all stage run in constant time except the EKF observation stage. Since it uses



(a) Orthographic view

(a) Perspective view

(a) Landmark tracking state

Fig.11 Map building result: Lobby

full-covariance EKF, observation time increases in proportion to the order of N^2.

Stage		Elapsed Time (ms)
Preprocessing		72.0332
Data Association		89.8094
EKF	Prediction	0.0042
	Update	5.0009
	Observe	305.2380
Total		472.0857

Table 1. Elapsed time table: (201 landmarks)

VI. Conclusion & Future works

We have proposed a new CV-SLAM that uses a single upward camera for visual correspondence of natural landmarks. The scale-invariant property of the ceiling vision made the visual data association problem to be rather simple so that only rotation and shear deformation need to be considered. 2D/3D gradient orientations estimation and multi-view description technique are used for efficient view-invariant landmark matching. Experimental results in various real environments showed that the proposed CV-SLAM technique was very fast, stable and accurate. Our further works will include the solutions to the very large closing loop and repeated landmark problems.

Acknowledgements

This work has been supported in part by the Ministry of information & Communications, Korea, under the information Technology Research Center (ITRC) Support Program.

References

[1] A.J. Davison. Real-time simultaneous localization and mapping with a single camera. In Proc. ICCV, 2003

[2] G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building(SLAM) problem. IEEE Trans. Robotics and Automation, 2001.

[3] Se, S., Lowe, D., and Little, K. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks IJRR 2002

[4] David G. Lowe. Distinctive image features from scale-invariant keypoints. IJCV 2004

[5] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In ECCV, pp. 128–142, 2002.

[6] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In ICCV 2001

[7] C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” Alvey Vision Conf., 1988, pp. 147–151.

[8] C.-C. Wang. And C. Thorpe. Simultaneous Localization Mapping with Detection and Tracking of Moving Objects. In Proc. ICRA 2002.

[9] M. Montemerlo and S. Thrun. Simultaneous Localization and Mapping with Unknown Data Association Using FastSLAM. In Proc. ICRA 2003

[10]M. Jorgan and A Leonardis, “Robust localization using an omni directional appearance based subspace model of environment”, Robotics and Autonomous Systems, Volume 45, Issue 1, pp. 51~72, Elsevier Science, 2003

[11]J. Koˇseck´a and F. Li, “Vision Based Topological Markov Localization” In Proc. ICRA 2004

[12]J. Wolf, W. Burgard, H. Burkhardt. “Robust Vision-based Localization for Mobile Robots Using an Image Retrieval System Based on Invariant Features,” In Proc. ICRA 2002.

[13]N. D. Molton and A. J. Davison and Ian D. Locally Planar Patch Features for Real-Time Structure from Motion. In Proc. BMVC 2004

[14]J. Meltzer, R. Gupta, M-H. Yang, S. Soatto. "Simultaneous Localization and Mapping using Multiple View Feature Descriptors." In Proc. IROS 2004