Engineering and Technology Journal e-ISSN: 2456-3358
Volume 06 Issue 11 November-2021, Page No.-1066-1071
DOI: 10.47191/etj/v6i11.04, I.F. – 6.39
© 2021, ETJ

Study of Indoor Visual SLAM System for Semi-autonomous Robot Platform

Yeon Taek OH

School of Mechanical Engineering, Tongmyung University, 428 Sinseon-ro, Nam-gu, Busan, 48520, Korea

ABSTRACT: This study propose the use of heterogeneous visual landmarks, points and line segments, to achieve effective cooperation in indoor SLAM environments. In order to achieve un-delayed initialization required by the bearing-only observations, the well-known inverse-depth parameterization is adopted to estimate 3D points. Similarly, to estimate 3D line segments, we present a novel parameterization based on anchored Plücker coordinates, to which extensible endpoints are added.

KEYWORDS: Camera Calibration, Feature Extraction, Odometry, Indoor Visual SLAM, Semi-autonomous Robot

I. INTRODUCTION

Most sensors used for SLAM operation such as laser range finder, IR and Star Gazer are expensive or needed some artificial markers for the sensing. CCD camera can reduce cost of production of robot because it is relatively cheap and able to directly use natural markers which cover environment enables its broader application range. Point features have been most used as landmarks for vision sensors due to the fact that it facilitates its general use and application.

In comparison to the point features, straight line features can be easily detected, tracked and less affected by dynamic obstacles such as pedestrians. Furthermore, it enables the system to perform SLAM even with significantly fewer features than the case of using point features.

The simultaneous localization and mapping (SLAM) problem asks if it is possible for a mobile robot to be placed at an unknown location in an unknown environment and for the robot to incrementally build a consistent map of this environment while simultaneously determining its location within this map. A solution to the SLAM problem has been seen as a “holy grail” for the mobile robotics community as it would provide the means to make a robot truly autonomous [1]. This paper introduced a vision-based SLAM system that used both of point and line feature for mobile robots in indoor environment.

II. DEVELOPMENT OF INDOOR VISUAL SLAM SYSTEM

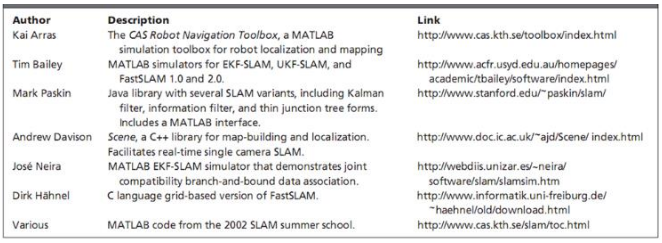
The solution of the SLAM problem has been one of the notable successes of the robotics community over the past decade. SLAM has been formulated and solved as a theoretical problem in a number of different forms. SLAM has also been implemented in a number of different domains from indoor robots to outdoor, underwater, and airborne systems. At atheoretical and conceptual level, SLAM can now be considered a solved problem. However, substantial issues remain in practically realizing more general SLAM solutions and notably in building and using perceptually rich maps as part of a SLAM algorithm.

SLAM is a process by which a mobile robot can build a map of an environment and at the same time use this map to deduce its location. In SLAM, both the trajectory of the platform and the location of all landmarks are estimated on line without the need for any a priori knowledge of location.

Solutions to the probabilistic SLAM problem involve finding an appropriate representation for both the observation model and motion model that allows efficient and consistent computation of the prior and posterior distributions in equations Time-update and Measurement Update. By far, the most common representation is in the form of a state-space model with additive Gaussian noise, leading to the use of the extended Kalman filter (EKF) to solve the SLAM problem. One important alternative representation is to describe the vehicle motion model in equation The motion model as a set of samples of a more general non-Gaussian probability distribution. This leads to the use of the Rao-Blackwellized particle filter, or Fast SLAM algorithm, to solve the SLAM problem. While EKF-SLAM and Fast SLAM are the two most important solution methods, newer alternatives, which offer much potential, have been proposed, including the use of the information-state form.

Various researchers in the SLAM community have written software demonstrating SLAM, implemented in MATLAB, C++, and Java and available online. Collections of logged data are also listed. These datasets are from real sensors in real environments and are a valuable resource to assess and benchmark the various SLAM algorithms.

Table 1. Open source SLAM software



Author	Description	Link
Kai Arras	The *CAS Robot Navigation Toolbox*, a MATLAB simulation toolbox for robot localization and mapping	http://www.cas.kth.se/toolbox/index.html
Tim Bailey	MATLAB simulators for EKF-SLAM, UKF-SLAM, and FastSLAM 1.0 and 2.0.	http://www.acfr.usyd.edu.au/homepages/academic/tbailey/software/index.html
Mark Paskin	Java library with several SLAM variants, including Kalman filter, information filter, and thin junction tree forms. Includes a MATLAB interface.	http://www.stanford.edu/~paskin/slam/
Andrew Davison	*Scene*, a C++ library for map-building and localization. Facilitates real-time single camera SLAM.	http://www.doc.ic.ac.uk/~ajd/Scene/ index.html
José Neira	MATLAB EKF-SLAM simulator that demonstrates joint compatibility branch-and-bound data association.	http://webdiis.unizar.es/~neira/software/slam/slamsim.htm
Dirk Hähnel	C language grid-based version of FastSLAM.	http://www.informatik.uni-freiburg.de/~haehnel/old/download.html
Various	MATLAB code from the 2002 SLAM summer school.	http://www.cas.kth.se/slam/toc.html

Table 2. Online datasets

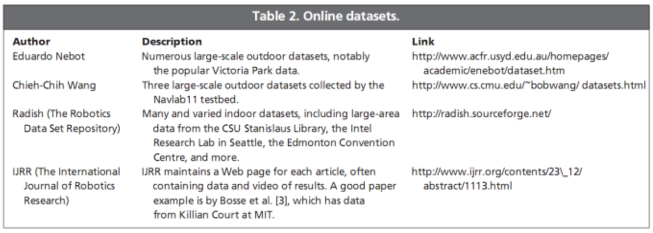


Table 2. Online datasets.

Author	Description	Link
Eduardo Nebot	Numerous large-scale outdoor datasets, notably the popular Victoria Park data.	http://www.acfr.usyd.edu.au/homepages/academic/enebot/dataset.htm
Chieh-Chih Wang	Three large-scale outdoor datasets collected by the Navlab11 testbed.	http://www.cs.cmu.edu/~bobwang/ datasets.html
Radish (The Robotics Data Set Repository)	Many and varied indoor datasets, including large-area data from the CSU Stanislaus Library, the Intel Research Lab in Seattle, the Edmonton Convention Centre, and more.	http://radish.sourceforge.net/
IJRR (The International Journal of Robotics Research)	IJRR maintains a Web page for each article, often containing data and video of results. A good paper example is by Bosse et al. [3], which has data from Killian Court at MIT.	http://www.ijrr.org/contents/23_12/abstract/1113.html

A Monocular camera is a projective sensor that measures the bearing of image features. Given an image sequence of a rigid 3-D scene taken from a moving camera, it is now well known that it is possible to compute both a scene structure and a camera motion up to a scale factor. To infer the 3-D position of each point feature, the moving camera must observe it repeatedly each time, capturing a ray of light from the point feature to its optic centre. The measured angle between the captured rays from different viewpoints is the point feature’s parallax—this is what allows its depth to be estimated [2].

Most of SLAM works have made use of point features because points are easy to extract, match, and represent. However, there is wide consensus on the fact that punctual world representations cannot provide satisfactory mapping results. Indeed, a map consisting of a sparse set of 3D points is far from describing the structure of the surrounding world. This is inherent to the dimensionless character of points, which contain no notion of neighbourhood by themselves. Instead, segment-based landmarks include the one-dimensional notion of connectivity, and the two- dimensional notion of boundary, providing the map with a much richer representativeness of reality. Connectivity and boundary information can be exploited to establish useful metrical and topological descriptions of the environment. This is effectively an important step forward for structured scenarios with plenty of straight lines [3].

The extended Kalman filter formulation of simultaneous localization and mapping (EKF-SLAM) algorithm produces very optimistic estimates once the “true” uncertainty in vehicle heading exceeds a limit. To overcome this, we decided to use the sensor-centred Extended Kalman Filter. Contrary to the standard EKF, where estimates are always referred to a world reference frame, the sensor-centred approach represents all feature locations and the camera motion in a reference frame local to the current camera. The typical correlated feature-sensor uncertainty arising from the uncertain camera location is transferred into uncertainty in the world reference frame, resulting in lower variances for the camera and map features, and thus in smaller linearization error. Another characteristic of our algorithm is the use of 1-point RANSAC for EKF estimation; which in the case of the EKF with tightly correlated priors is enough to discard spurious matches. A reduction in the number of random hypotheses that are needed as in our approach, comes as a result of incorporating into the motion model some restrictions on the allowed motion; specifically planar motion and a large radius of curvature typical of robot motion is assumed. The extra information available from a motion model to aid matching is dealt with in a much more general manner, and we are able to cope with smooth camera motion with the full robot motion [4].

III. SYSTEM DEVELOPMENT

This section the development of the Indoor Visual SLAM System. First, we will introduce how our system is configured. As a main computational unit, a laptop is placed on the track robot, and an USB camera is connected to the laptop. The track robot provides the odometer information through a serial port. From the calibrated monocular camera, the point feature and the line feature of the environment are extracted, and these visual features together with the odometer information are used as the sensory input to the EKF-SLAM system. The EKF-SLAM is implemented as camera-centered variant to prevent the inconsistency inherently included in the standard EKF algorithms.

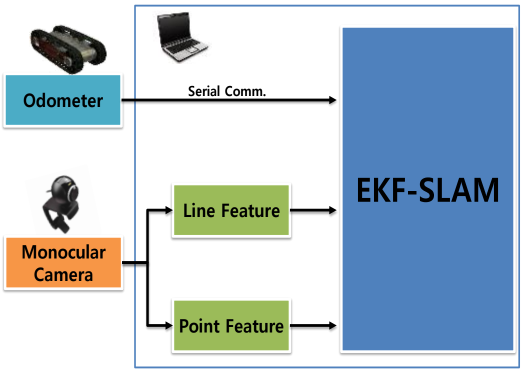


Fig. 1 System configuration

IV. EXPERIMENTS

We performed three sets of experiments in different environments; the first experiment was performed with a corridor dataset taken from indoor, the second experiment was performed with a laboratory dataset, and the third experiment was performed with an office dataset. The latter two dataset were all gathered at Robotics Corporation. Before proceed the experiments, the camera was calibrated firstly with the conventional check board method.

A. Camera Calibration

We used the open “Camera Calibration Toolbox for Matlab” for the calibration of the monocular camera. We used twenty images for the calibration and some of them are shown in the next figure.

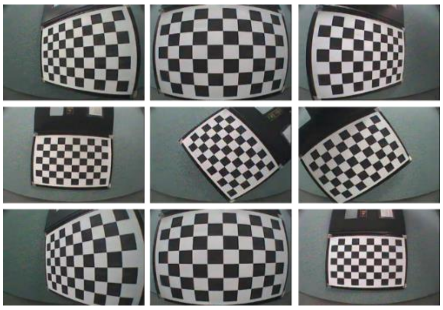


Fig. 2 Image used for the calibration of the camera

After reading in the pattern images, the four corners of each image should be manually provided using careful mouse clicks. Then the toolbox will ask you to type in some parameters about the check board pattern. After corner extraction, you can click on the Calibration button to run the main camera calibration procedure.

The results of the calibration process conducted on the previously shown images are given in the next table. The table includes the camera intrinsic parameters and the radial distortion parameters.

Table 3. Calibration Results

Intrinsic parameters	
f_x	208.20556
f_y	208.44251
x_0	159.3736
y_0	119.71155
Radial distortion parameters	
κ_1	-0.31880
κ_2	0.07455
κ_3	0.00025
κ_4	0.00022

After the camera calibration, the extracted intrinsic parameters were used in visual data processing for accurate environment mapping. In the following experiments, all images were collected at a resolution of 320×240 pixels.

B. Feature Extraction

The point features are extracted using the FAST Corner detector; the line features are extracted by checking pixel continuity test with Canny edge images.

The FAST Corner detector [5] is used to find the point feature in the input image. The segment test criterion operates by considering a circle of sixteen pixels around the corner candidate p. The detector classifies p as a corner if there exist a set of 9 contiguous pixels in the circle which are all brighter than the intensity of the candidate pixel Ip plus a threshold t, or all darker than Ip-t, as illustrated in next figure.

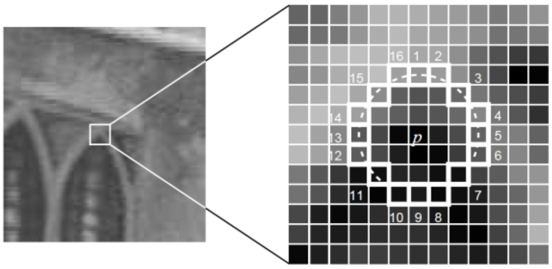


Fig. 3 Fast Corner detection

For the line-based SLAM feature, the line segments are extracted as follow: First Canny edge detector is used to extract edges, and then edges are split into points at high curvatures. Finally, segments whose length is less than 20 pixels are discarded and lines are fitted to the split edges using the least squares method. Next figure shows an example of the line processing result for the input image.

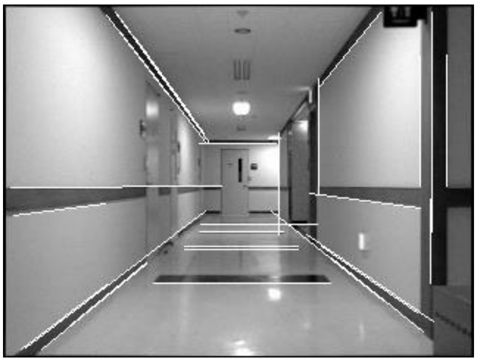


Fig. 4 An example of line extraction from a sample image

C. Corridor Environment

The first experiment was conducted in an indoor corridor environment and the typical input images are shown in next figure. The corridor environment is featured by the narrow navigation path, small number of point features, and rich line features. The floor is flat and decorated with rectangular black blocks. Each block has a width of 0.45m and they are spaced about 2.3m apart. The robot started from the lower part of the left corridor, and travelled in a loop through the interior of the building. The entire rectangular path has dimensions of approximately 11.5×24m, and the robot was manually driven during the experiment.

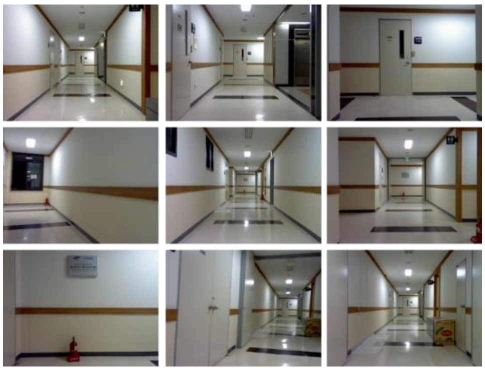


Fig. 5 Image samples from the corridor environment

The odometer readings collected during our experiment are plotted in the next figure for reference. The robot was calibrated before the experiment, and accordingly it showed a straight trajectory when it was traveling forward; however, it showed a strong tendency of turning right when changing its direction. As a result, the whole path shows a severe trend of spinning inward, and forms a twisted rectangle.

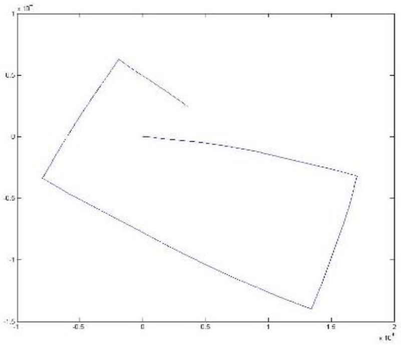


Fig. 6 Plotted odometer data from the corridor environment

Next figure shows the result of the corridor SLAM. Unlike the figure with the odometer readings, in the resulting SLAM map, the tendency of turning right was significantly decreased, and much of the twisted path is correctly revised.

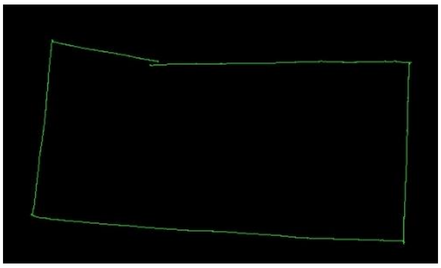


Fig. 7 Robot trajectory of the corridor SLAM

Including line features in the 3D SLAM map has an apparent advantage than using only point features. As can be seen from next figure, the line features convey important information of the surroundings such as navigable areas, contours of doors, entrances, corners, and so on. The rich map does not only assist the SLAM system in achieving successful localization, but the map itself can accommodate as much of the structural properties of the surroundings as possible. The rich map plays an important role in connecting the metric map with the semantic map to provide more useful services to human.

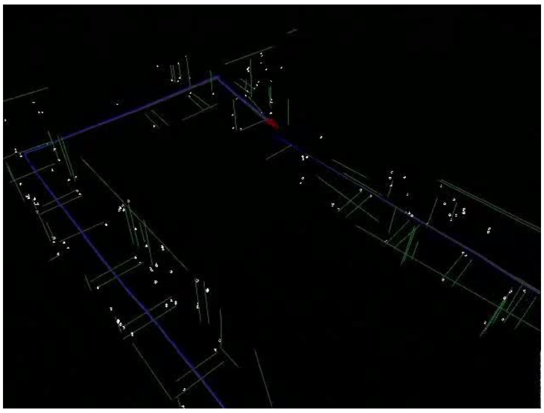


Fig. 8 The 3D map of the corridor SLAM

D. Laboratory Environment

The laboratory dataset was taken in a laboratory of the Robotics Corporation. Since the camera was placed at a low height, the lower part of the input image is occupied by no meaning region of floor plane, as can be seen from the next figure. This means placing the camera at a low height is challenging for SLAM in an indoor environment, since fewer features could be employed in the SLAM process. The laboratory dataset is also featured by the cluttered environment, with small number of line features.

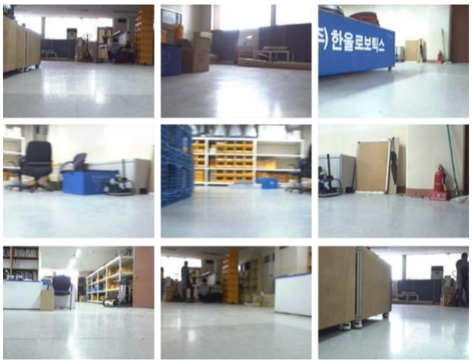
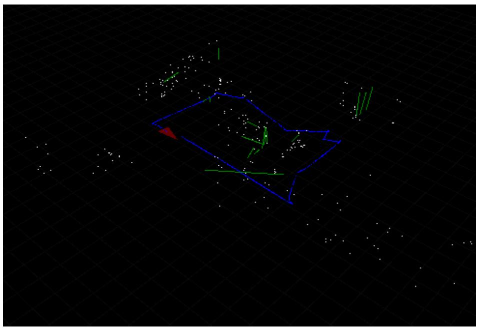


Fig. 9 Image samples from the laboratory environment

For a 3D point feature, providing parallax means that the camera should deviate from the line connecting the point to the position from which the point feature was initially observed. On the other hand, for a 3D line feature, providing parallax means that the camera must deviate from the plane made by joining the line feature to the position of the initial observation. This implies that using lines as SLAM features has more constraints compared to using point features, and some lines are prone to remaining as immature features since the uncertainty of the line feature does not converge to a reliable threshold. This result is clearly reflected in the next figure of the 3D SLAM resulting from the laboratory dataset. Compared to the corridor environment, the number of the converged lines is apparently diminished; however, the point number is much higher than the corridor case. This is caused by the open and cluttered laboratory environment.



E. Office Environment

The laboratory dataset was also taken from an office of the Robotics Corporation. Like in the case of the laboratory dataset, the lower part of the input image is also occupied by texture-less region. Compared to the laboratory environment, the office is featured by the fewer point features and richer line features.

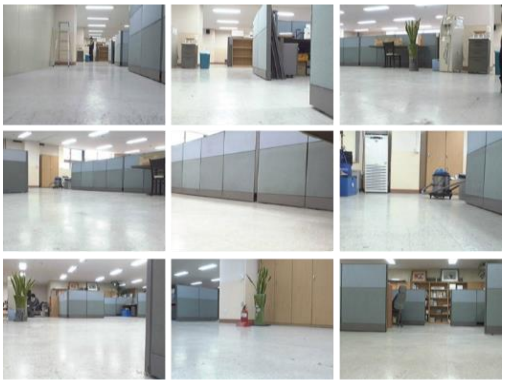


Fig. 10 Image samples from the office environment

As shown in the next figure, the resulting 3D SLAM map of the office dataset clearly conveys what we have explained in the previous paragraph about the characteristic of the office environment. Compared to the 3D map of the laboratory environment, there are fewer point features in the map; even though the number of line features are more than the laboratory dataset, however, it is still less than the case of the corridor environment. This is caused by the fact: the office environment is usually more open than the corridor environment. This implies that the line features observed by the robot experience less parallax than it is located in a narrow corridor.

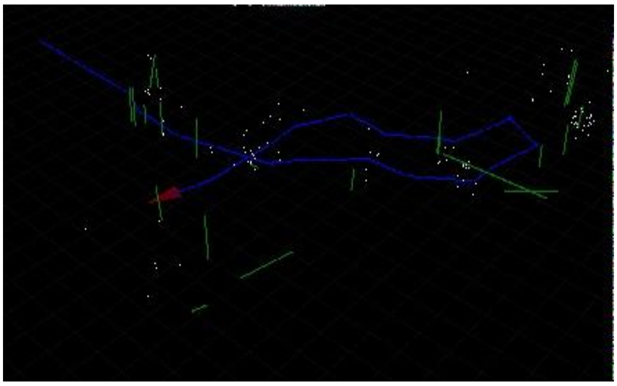


Fig. 11 The 3D map of the office SLAM

The final demo was conducted in the laboratory environment. The next figure shows the resulting 3D map.

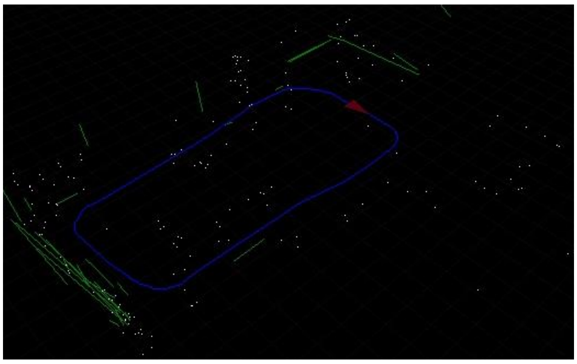


Fig. 12 The 3d map of the final demo

The following figure shows the top view of the resulting map.

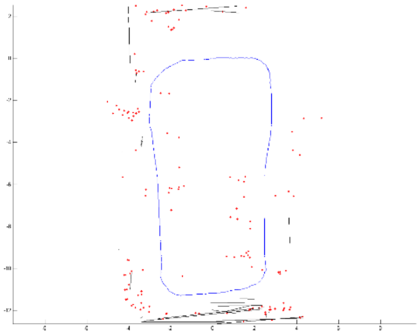


Fig. 13 The top view of the resulting map

V. CONCLUSION

In this work, we propose the use of heterogeneous visual landmarks, points and line segments, to achieve effective cooperation in indoor SLAM environments. In order to achieve un-delayed initialization required by the bearing-only observations, the well-known inverse-depth parameterization is adopted to estimate 3D points. Similarly, to estimate 3D line segments, we present a novel parameterization based on anchored Plücker coordinates, to which extensible endpoints are added.

To overcome the inconsistency problem inherently hide in the standard EKF-SLAM system, we have decided to adopt the sensor-centered Extended Kalman Filter. Contrary to the standard EKF, where estimates are always referred to a world reference frame, the sensor-centered approach represents all feature locations and the camera motion in a reference frame local to the current camera. The typical correlated feature-sensor uncertainty arising from the uncertain camera location is transferred into uncertainty in the world reference frame, resulting in lower variances for the camera and map features, and thus in smaller linearization errors. Another characteristic of our algorithm is the use of 1-point RANSAC for EKF estimation; which in the case of the EKF with tightly correlated priors is enough to discard spurious matches. A reduction in the number of random hypotheses that are needed as in our approach, comes as a result of incorporating into the motion model some restrictions on the allowed motion; specifically planar motion and a large radius of curvature typical of robot motion is assumed. The extra information available from a motion model to aid matching is dealt with in a much more general manner, and we are able to cope with smooth camera motion with the full robot motion.

To validate the efficiency of our proposed Indoor Visual SLAM system, we conducted three sets of experiments with real sensory data collected from differently characterized environments. With the same parameter setting, we found that the line features are more easily initialized in the corridor environment, and the point feature is more suitable for the cluttered laboratory environment. All the experimental results show that our developed Indoor Visual SLAM system is sufficient for mapping a moderately sized, structured indoor building environment.

In future work, we will consider improving the line feature matching by using a globally distinguishable descriptor, and will implement the loop closure algorithm for both the point and the line features. To solve the scalability problem and to improve the accuracy of the resulting SLAM map, we also plan transferring the core SLAM algorithm from the extended Kalman filtering to the bundle adjustment techniques.

REFERENCES

1) H. Durrant-Whyte, T. Bailey, "Simultaneous localization and mapping: Part I", IEEE Robotics & Automation Magazine, vol. 13, no. 2, pp. 99-110, Jun. 2006.
2) J. Civera and A. Davison and J. Montiel, "Inverse depth parametrization for monocular SLAM", IEEE Transactions on Robotics, vol. 24, no. 5, 2008.
3) J. Solà and T. Vidal-Calleja and M. Devy, "Un-delayed initialization of line segments in monocular SLAM", in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, St. Louis, USA, 2009.
4) J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel, "1-pointRANSAC for EKF-based structure from motion," In Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS), 2009.
5) E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," In Proceedings of the International Conference on Computer Vision (ICCV), 2005.
6) G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," Interface, vol. 2, no. 2, 1998.
7) J. Allen and R. Xu, "Object tracking using camshift algorithm and multiple quantized feature spaces," Proceedings of the Pan-Sydney area, vol. 36, pp. 3-7, 2004.
8) P. Fieguth and D. Terzopoulos, "Color-based tracking of heads and other mobile objects at video frame rates," Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 21-27, 1997.
9) O.-dris Nouar, G. Ali, and C. Raphaël, "Improved object tracking with camshift algorithm," pp. 657-660, 2006.