Vision Inspired Simultaneous Localization and Mapping: The Algorithm

Shiva Karthik Reddy Koyya
Department of Electrical and Microelectronics Engineering
Rochester Institute of Technology
Rochester, New York 14623
Email: shiva.reddy37@gmail.com; kxr3779@g.rit.edu

***Abstract*—This paper presents a vision inspired Simultaneous Localization and Mapping algorithm which utilizes an improved Oriented Rotated Brief features for fast features matching. The project works in real time in a static indoor environment on an Adept Mobile robot (Amigobot) equipped with Xtion Pro Live as Sensor. The major goal of this project was to make an algorithm form the scrach capable of Localizing the Robot with high accuracy and in the process creating a sparse 3D map. The major contribution of this paper is in increasing the feature (Landmark in context of Simultaneous localization and mapping) matching speed by applying median filter on images before using Oriented Rotated Brief (ORB) which not only gives more distinctive feature point but improves the matching speed when using RANSAC for feature matching, which overall effects the real-time performance of the algorithm.**

I. Introduction

This paper aims to present and vision inspired particle filter based SLAM algorithm implemented on a mobile robot equipped with a 3D sensor.

Simultaneous Localization and Mapping (SLAM) is a challenging problem and has been studied extensively in the literature of Mobile robotics / Mobile Autonomous systems. The problem deals with building a map of an unknown environment and determining the robot pose in that environment simultaneously. The author in [4] presents a two part paper with an extensive study on the SLAM, approaches taken to solve the problem and issue that still needs to be addressed. Though the SLAM problem has been solved to a great extend issues like building a rich dense 3D map, Computational Complexity, speed, good data associations are still some areas to be worked upon. The paper proposes an improved Oriented Rotated Brief (ORB) feature based SLAM algorithm with Rao Blackwellised Particle Filter for state estimation to be implemented on an Adept Mobile Robot (Amigobot) equipped with a RGB and Depth sensor: Xtion Pro Live ;developed by Asus. The algorithm proposes use of Median filter on the RGB images before extraction of ORB features which results in a faster feature matching process with less false positive.. The main focus of the project was to come up with an algorithm which is able to localize the robot real time while creating a Sparse Feature map with just enough features/Landmarks so as to achieve precise localization. The paper also addresses the problem of Data Association via use of a Locality sensitive Hashing algorithm for initial correspondence and Random Sample Consensus for final correspondence of features/ Landmarks. Rest of the paper is arranged in the following sections: Section II briefs over the work done in the area of SLAM in the past, to be particular ORB Vision based SLAM[6]. The proposed algorithm is explained in detail discussing the image processing aspect, RBPF backend, sensor, robot system in section III. Section IV discusses the results, performance of the system and the paper concludes in section V suggesting future scope of improvements. In rest of the paper word feature and landmark are used interchangeably depending upon the context discussed but they mean the same.

II. Related Work

The introduction of 3D cameras like Microsoft Kinect, Xtion pro gave new dimensions to researchers in field of Mobile robot navigation and vision systems. These cameras being able to provide RGB image along with its associated depth proved beneficial in research related to SLAM, Mapping etc. The authors in [7], [3] used RGBD information from Kinect for generating a Dense 3D map of an indoor environment. In these studies the authors extracted visual features from a 3D point cloud, used visual odometry and algorithms like RANSAC and Iterative Closest Point (ICP) for alignments of consecutive frames to create a map of the environment. The process of creating a rich dense visual map is a computationally expensive process and needs good hardware. In this study to achieve good real-time performance sparse map representations is used. This paper focuses of implementation of SLAM in an indoor well structured environment which as abundant number of features like corners, edges etc. The use of features for map representation is a compact way and consumes less memory as compared to a grid based map. This study uses ORB features over SIFT and SURF features due to its inherent advantage in terms of speed of matching features as well as memory constraint. ORB descriptors as represented as 32 bit binary descriptors as compared to say SIFT with 128 bits descriptors. The author in [11] proposes a Monocular ORB SLAM with Bundle adjustment (BA) for camera localization. A very similar work to the one proposed in this paper is discussed in [6] where the data association is done using relative distance measure. The author in [14] provides a comparative study of stereo-camera and 3D camera for mobile robot localization. Particle filters based SLAM (Fast

SLAM 1 and 2) and localization gained its popularity due to its advantage over Kalman filters to scale due to its ability to scale large environments, represent non-linear behavior of the sensors and robot motion and computational effectiveness. The author in [10] discusses the limitation of Extended Kalman Filter over Particle filter and talks about FastSLAM an improved version of Particle Filters. The SLAM proposed in our approch utilizes FastSLAM 1 as its backend.[9] particle filters have been effectively used for people tracking , but in scope of this project we work with static environment.

III. PROPOSED ALGORITHM

The proposed algorithm is built in python using OpenCV for image processing and ROS for controlling the robot, interacting with the Xtion camera. Figure.1 below shows the general layout of the SLAM algorithm proposed. The rest of the section is divided as follows section A discusses about the RBPF backend in context of 3D landmark position. Image processing aspect which includes feature extraction, feature matching, improvement proposed in the ORB matching are discussed in section B. The frame transformations, Robot model, ROS is discussed in section C.

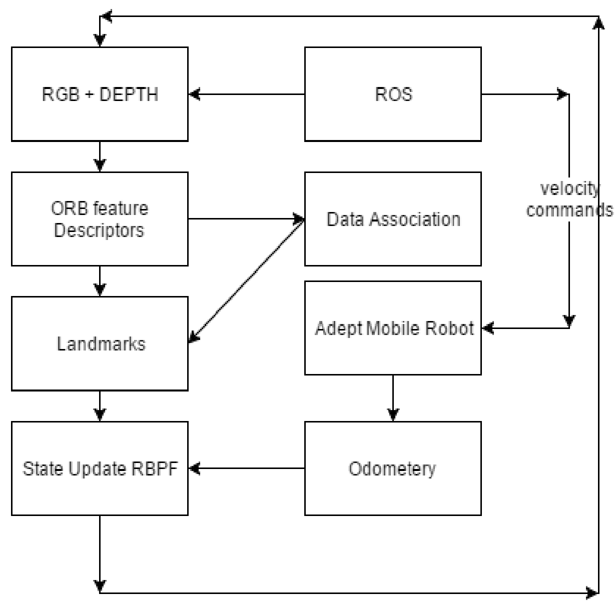

Fig. 1. Outline of the proposed algorithm. The detailed explanation of each module is presented in section III

A. FastSLAM Backend

Over the last decade, algorithms for the task of SLAM have grown tremendously. We have identified three paradigms to solve the problem of SLAM

1. Extended Kalman Filter (EKF) based
2. Particle Filter (PF) based
3. Graph based

The choice of a particular solution is based upon the requirements of the system, hardware availability, and external conditions. In our experiment we used FastSLAM, particle filter paradigm due to its easy implementation and successful application in past for achieving real-time SLAM [8], [13]. An effective implementation of FastSLAM results in a complexity of O(M log K) where M is the number of particles and K is the number of landmarks observed as compared to EFK which results in O(K2) complexity. Another advantage of FastSLAM over EKF SLAM is that in FastSLAM each particle has its own belief of landmarks positions which means data association is done on per particle basis which results is better data association.

The role of backend is to incorporate new observations into the map taking into account the current position of the robot. FastSLAM exploits the dependency between the robot pose and the landmark position i.e. if the pose of the robot is known with certain uncertainty the landmarks positions are independent from each other. Thus the FastSLAM decomposes the SLAM posterior distribution

$$p(s^t, \theta \left| u^t, z^t \right.) \tag{1}$$

(where s is the state of the robot at time t,theta is the map, u is the control command at time t and z is the observation at time t.)into seprate path and landmark distribution.

$$p(s^t \left| u^t, z^t \right.) \prod_{1}^{N} p(\theta | s^t, u^t, z^t) \tag{2}$$

In our experiment the robot state s is represented as 3-element vector

$$s = (x, y, \phi) \tag{3}$$

where x and y are robot 2d position and ϕ is the heading direction of the robot relative to the origin of the map.

the landmarks in the map are reperesented as

$$\Theta = (\theta_1, ...\theta_N) \tag{4}$$

Eash landmark is represnted by a triplet

$$\theta = (\mu, \sigma, d) \tag{5}$$

where μ is 3D landmark position, σ is the landmark error covarince and d is the landmark descriptor.

An observation i.e. a landmark as observed from the current robot position is described as a triplet

$$z = (p, R, d) \tag{6}$$

where p is the position, R is the error cavaince of the sensor, d is the landmark descriptor.

The motion command is assocoated with the motion velocity, the direction of motion and time of motion. Taking into account all these modification of the stuture of represenation of 3d landmarks the exact upadate equation for the fast SLAM can be derived from [8].

B. Image Processing

Making sense of images to find suitable landmarks is the essence of the SLAM problem. Accuracy of the sensor in its measurements of the landmarks affects the confidence of the robot about certainty of its position. In context of Vision systems extracting unique and repeatable features which can be matched efficiently in overlapping images is essential. A lot of research has been done with regards to what kind of features can be used which gives unique features. SLAM literature has used SIFT, SURF, ORB and other similar feature detectors. In this experiment we use ORB features with slight modification to improve the feature matching speed. ORB features use FAST (Feature from Accelerated Segment Test) algorithm for feature point detection with modification so as to provide scale and viewpoint invariance. After obtaining feature points with assocaited direction descriptors are built using BRIEF (Binary Robust Independent Elementary Features) algorith. BRIEF generate descriptors around feature points by binary coding method. The BRIEF descriptor is simpler and storage space is smaller than its counterparts SIFT and SURF.The Figure 2 below describes the flow of the image processing part of the SLAM problem and Figure 3 shows ORB features extracted from a indoor view.The rest of the section discusses the ORB modification used and the Data assocation using FLANN based matcher and RANSAC algorithm.

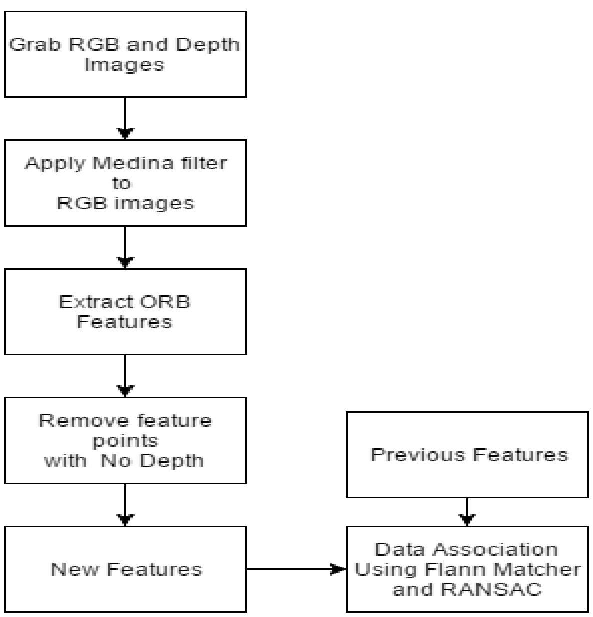

Fig. 2. The figure decipts the genral flow of the Image processing part of the SLAM

1) Median Filtering: In our algorithm we use a median filter on the images obtained in each frame before extracting the ORB features. The use of median filter removes the noise and gets higher precision matching points. The basic principle of Median filter is the value of point in the digital image is set to a mid-value, making the surrounding pixel values close to the real value and thus eliminating isolated noise. Figure

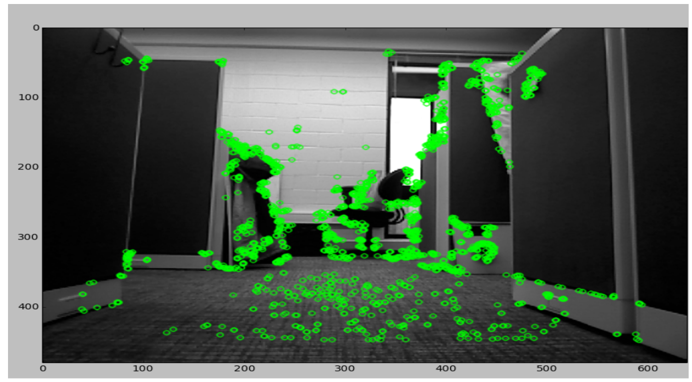

Fig. 3. 1000 ORB features extracted using OpenCV in an indoor office Environment

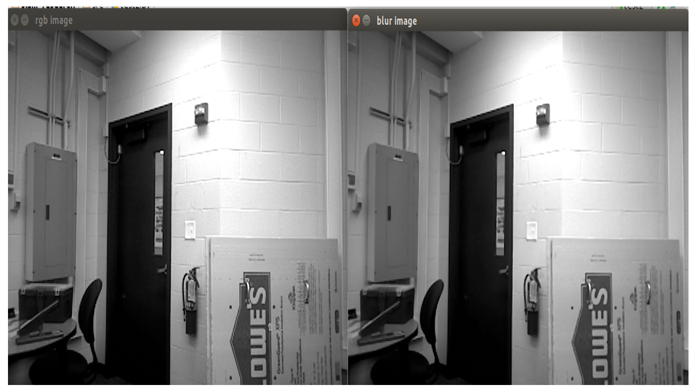

Fig. 4. The image on the left is the normal image and one on the right is the same image after median blur. The effect may not be so obvious but if we look closely along the chair we can see that the left image has some noise along the edges which is smoothened after Bluring.

4 shows an image before and after Median filtering. Median filtering is particularly effective in removing salt and pepper noise and at the same time could keep the edge sharpness; it will remove texture in the same area, such as the trees in background. The use of median filters affects the performance of RANSAC algorithm and provides more correct matching points. The improvement in performance with use median filters in increasing the matching accuracy as well as the speed of matching is discussed in Results section.

2) Data Association: Two Factors contribute to uncertainty in the SLAM posterior : Measurment noise and motion noise. Each leads to diffrent type of data ambiguity. In other words uncertainty of landmark position during measurement leads to confusion between nearby landmarks.Figure 5 below shows one such condition where due to pose uncernaity there is confusion in terms of the measurment obtained corresponds to which landmark. In our experiment the data association between landmarks is done in two steps. First an initial correspondence is made using nearest neighbor estimate between the current observations and previous Landmarks Descriptors. In case of a large map the observations has to be compared to all the landmarks which can be time consuming , therefore

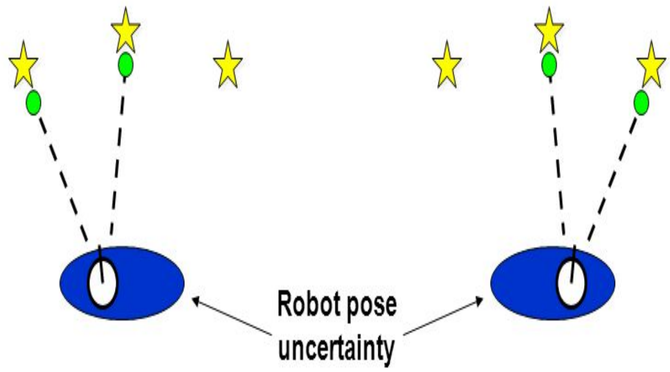

Fig. 5. The figure shows the data association problem due to uncertainty in the pose of the robot, i.e. if the robot is the as shown in left its assumed that the measurements are from two left most landmarks and when the robot pose is assumed as shown in the right the measurements are assumed from the rightmost landmarks, which leads to uncertainty of measurement.

the Locality Sensitive Hashing (LSH) algorithm[5] was used for nearest neighbor search.The LSH algorithm shows good performance with high dimensional ORB descriptors hence suitable for several thousands of landmarks. Once the initial correspondence was established the correspondence was further refined using RANSAC algorithm [2]. RANSAC algorithm in an iterative manner tries to find the best association taking into account inliers and outliers. Both RANSAC and LSH algorits were implemented using OpenCV.

C. ROS and Robot Model

ROS is an open source framework which provides libraries and tools for creating software applications for robot and provides hardware abstraction.ROS provides its SLAM package [1],ROS has been sucesfully been used with kinect for indoor mapping [12] and localization. In our experiment we use ROS to grab images frames and control the robot. ROS Aria package is used to control and give commands to the Amigobot. All the frame transformations for the entire SLAM system was taken care by creating a Universal Robot Descriptive Format(URDF)file for the robot mounted with the Xtion camera and then used ROS TF package to publish all trasformations.Figure 6 below show robot model with all associated frames.

IV. Results

The results of the experiment are discussed in two parts first we discuss how does use of median filter effect the performance of ORB descriptors in terms of its matching speed. Then we look at the SLAM results. The entire experiment was performed using Python, OpenCV and ROS on a hp notebook with Intel core i3 processor. The table 1 shows the results of ORB and Median blur + ORB methods with different number of features extracted from two partially overlapping scene (Figure 7). It can been seen from the table that Median + ORB is not only faster than ORB but given better correspondences (more matches). In a particular case where we extracted 500 features from the scenes ORB couldn't find any

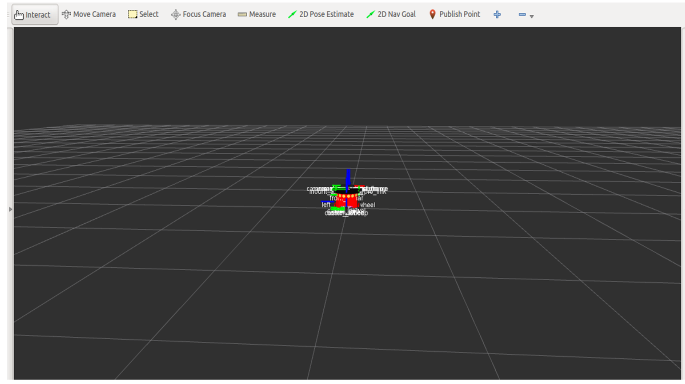

Fig. 6. Robot model with all frames; Rviz visualization.

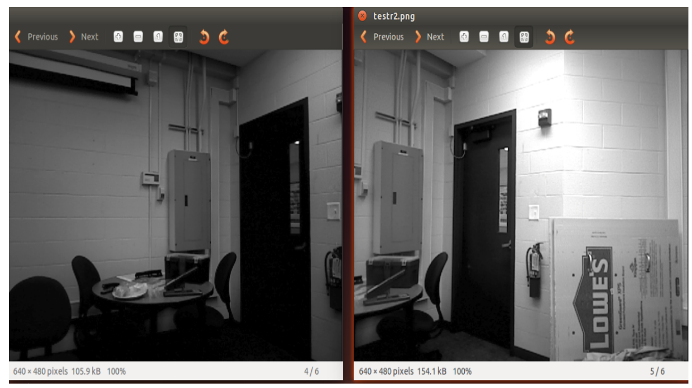

Fig. 7. The scene on which ORB and Median+ ORB methods were tested for speed of matching and number of good correspondence.

matches but ORB + median found 12 correct matches. Better correspondence between scenes in SLAM context is very important because greater the landmarks reobserved better is the localization accuracy which in turn effects the mapping accuracy. As mentioned earlier we are more concerned about the localization accuracy of the system while creating just a sparse map of the environment.

Method	Img1 and Img2	Correspondence	Time(sec)
ORB	1000,1000	25	0.09978
Median+ORB	1000,1000	32	0.08804
ORB	5000,1000	26	0.11261
Median+ORB	5000,1000	34	0.09400
ORB	500,500	0	0.13461
Median+ORB	500,500	34	0.09400

TABLE I
The Results for ORB and Median + ORB are listed along with the number of features extracted from each image, number of good correspondence made after RANSAC and time taken for making those correspondence

In early stages of our experiment, results were computed with diffrent number of paricle set size and diffrent number of Landmarks been extracted from each view. In the final run of the experiment we extracted 300 landmarks per view and used 20 particles for estimation of the pose of the robot. These final values were considered with keeping in

Fig. 8. The scene on which ORB and Median+ ORB methods were tested for their localization accuracies.

mind the realtime performance and the localization accuacy desired. The localization results for a particular office setup is below discussed which shows ORB + median filters features outperform the ORB features when it comes to convergence rate of the predicted value to the real world environment. Table 2 below shows the localization accuracies in both for both methods when tested in the same environment. In the particle filter algorithms,there are different ways to initiate the particles when the systems starts, in our experiment we initiate all the particles with their x coordinate, y coordinate and orientation as zero (0,0,0). This reasonable assumption because whenever our system initiates we assume the robot is a (0, 0, 0) and start mapping from there. In literature when the exact size of the mapping area is known particles are be distributed uniformly in the area but since in our case the map grows dynamically we use the (0, 0, 0) initialization. When computing the localization accuracy of the system at each time stamp we average over the pose of all the particles and compare it against the current pose of the robot. The table 2 discusses the case where the robot is moving in a straight line path and trying to map the environment as shown in figure 8. The mapping results are not discussed in this research.

TimeStamp	ORB	Median+ORB	Corr ORB	Corr Median+ORB
1	0.61	0.71	0	0
2	0.67	0.70	200/32	212/35
3	0.54	0.505	193/179	213/199
4	0.47	0.30	235/22	200/199
5	0.38	0.17	240/225	204/202
6	0.25	0.10	250/28	239/12
7	0.14	0.06	273/40	246/32

TABLE II

LOCALIZATION ACCURACY OF ORB AND ORB + MEDIAN ARE COMPARED, WHEN THE ROBOT IS MOVING IN STRAIGHT LINE. THE TABLE SHOWS THE CORRESPONDENCE MADE IN EACH STEP IN TERMS OF THE NUMBER OF MATCHES FOUND / NUMBER OF GOOD CORRESPONDENCE MADE.

the table 2 shows the accuracies of both the methods at each time stamp when tested in the same environment. It can be seen that the Median + ORB converges to real world with an error of 0.06m in 7 time stamps which corresponds to approximately 7 meters of robot movement. We can see that in time stamp 3 Median + ORB makes better correspondences as compared to ORB and converges to ground truth better.

Table 3 shows the timing performance of both the methods , i.e the time taken for the particle filter backend to run at each time stamp. We can observe from table 3 that the average time taken by the particle filter back end to run for the ORB + median filter is less than that or normal ORB which also emphasizes on the fact that updating observed landmarks is less time consuming that initializing new landmarks.Since ORB + median gives better good correspondances therefore less landmarks have to be initilized.

TimeStamp	ORB pf timing(sec)	Median+ORB pf timing(sec)
1	1.12	1.25
2	0.35	0.31
3	0.49	0.56
4	0.24	0.04
5	0.65	0.05
6	0.10	0.20
7	0.71	0.54
	max = 0.71	max = 0.56
	avg = 0.52	avg = 0.42

TABLE III

TIMING CHART FOR THE TIME TAKEN BY ORB AND ORB + MEDIAN PARTICLE FILTERS(PF) TO RUN.

V. CONCLUSION

In this project we introduced a Vision Inspired Simultaneous localization and mapping algorith with improved ORB features for better correspondance and better maching speed. The project worked in realtime on a Amigobot equiped with Xtion Pro live sensor and utilising Particle Filters as backend for the stateEstimation.The project was sucessful as the localization aspect of SLAM problem was sucessfully addressed. Use of Median filter on the images before extraction of ORB features helped in improving correspondance beteen two consicutive view which improved the overall performance of the system in terms of timing as well as localization.The reslts showed a localization accuracy with an error less than 0.06 meters over a run of 10 meters in an indoor staic office like envionemnt with pronounced corners and edges.

The experimental results showed that the performance of the syetems increases with incresing the number of paricles in the system estimating the robot pose and maintaing their own maps , but the increase in the number of particles also effected the realtime performance of the system as the particle filter run time increased with the number of particles increased.

VI. FUTURE WORK

In this particular implementation of SLAM algorithm a lot of aspects of the problems were relaxed, first creating a rich visual map of the environment, secondly the environment in which the system was tested was static and well-structured environment. Aspects like loop closure, kidnapped robot problem, environments with not or similar features were not addressed in the scope of the work. As an extension of the work, GPU pipeline system can be employed for creating a rich visual maps of the environment with high detail. A SLAM system working in dynamic environment where the landmarks can change with time is still an active topic of research and coming up with algorithms to tackle this particular aspect has to be worked upon. Tackling featureless environment and loop closure will require a sophisticated algorithm for feature database which can make decisions about which feature to track and the once to be remove over a long period of time. One observation with particle filters is that for unknown environment choosing a large number of particles effect the real-time performance of the system and choosing less particles effects the accuracy of the system as there are sufficient samples of the pose of the robot ,which enforces us to look at other alternatives like graph SLAM which have showed better results than particle Filter is some cases.

References

[1] B. L. E. A. Balasuriya, B. A. H. Chathuranga, B. H. M. D. Jayasundara, N. R. A. C. Napagoda, S. P. Kumarawadu, D. P. Chandima, and A. G. B. P. Jayasekara. Outdoor robot navigation using gmapping based slam algorithm. In *2016 Moratuwa Engineering Research Conference (MERCon)*, pages 403–408, April 2016.

[2] Robert Cupec, E Karlo Nyarko, and Ratko Grbic. Detection of dominant planar surfaces in disparity images based on random sampling. *Tehnicki Vjesnik*, 18(3):301–308, 2011.

[3] H. Dong, N. Figueroa, and A. El Saddik. Towards consistent reconstructions of indoor spaces based on 6d rgb-d odometry and kinectfusion. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1796–1803, Sept 2014.

[4] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.

[5] Y. Fan, J. Xing, and W. Hu. Load-balanced locality-sensitive hashing: A new method for efficient near duplicate image detection. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 53–57, Sept 2015.

[6] J. Hartmann, D. Forouher, M. Litza, J. H. Kluessendorff, and E. Maehle. Real-time visual slam using fastslam and the microsoft kinect camera. In *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, pages 1–6, May 2012.

[7] O. Miksik, Y. Amar, V. Vineet, P. Prez, and P. H. S. Torr. Incremental dense multi-modal 3d scene reconstruction. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 908–915, Sept 2015.

[8] M. Montemerlo and S. Thrun. Simultaneous localization and mapping with unknown data association using fastslam. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pages 1985–1991 vol.2, Sept 2003.

[9] M. Montemerlo, S. Thrun, and W. Whittaker. Conditional particle filters for simultaneous mobile robot localization and people-tracking. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 1, pages 695–701 vol.1, 2002.

[10] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. In *Aaai/iaai*, pages 593–598, 2002.

[11] R. Mur-Artal, J. M. M. Montiel, and J. D. Tards. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, Oct 2015.

[12] H. I. M. A. Omara and K. S. M. Sahari. Indoor mapping using kinect and ros. In *Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on*, pages 110–116, Aug 2015.

[13] Alexandr Prozorov, Alexandr Tyukin, Ilya Lebedev, and Andrew Priorov. Self-localization of mobile robot in unknown environment. In *Open Innovations Association (FRUCT), 2015 17TH Conference of*, pages 173–178. IEEE, 2015.

[14] L. Sabattini, A. Levratti, F. Venturi, E. Amplo, C. Fantuzzi, and C. Secchi. Experimental comparison of 3d vision sensors for mobile robot localization for industrial application: Stereo-camera and rgb-d sensor. In *Control Automation Robotics Vision (ICARCV), 2012 12th International Conference on*, pages 823–828, Dec 2012.