https://www.imavs.org/pdf/imav.2016.32

Real-time Simultaneous Localization and Mapping for UAV: A Survey

Jiaxin Li, Yingcai Bi, Menglu Lan, Hailong Qin, Mo Shan, Feng Lin, Ben M. Chen
National University of Singapore

http://www.imavs.org/

ABSTRACT

Simultaneous Localization and Mapping (SLAM) refers to the problem of using various sensors like laser scanner, RGB cameras, RGB-D cameras, etc, to estimate the position of the robot, and concurrently construct the 2D/3D map of the environment. The SLAM community has made great progress in the past few decades. So far the 2D SLAM problem with range finders is considered as solved, while the real-time 3D SLAM, especially robust and high quality visual SLAM on UAVs, remains an open problem. This article aims to give a picture of the evolution and very recent development of SLAM algorithms, and emphasis is given on real-time SLAM methods that are suitable for Unmanned Aerial Vehicles (UAVs).

1 INTRODUCTION

Autonomous UAVs, robots, vehicles are receiving more and more attention in academia and industry. Although the industry has noticed the potential applications of unmanned systems, including driver-less cars, building inspection, surveillance etc, lots of problems remain unsolved. Most of problems are caused by the failure, inaccuracy or instability of perception. In outdoor environment, GPS provides accuracy localization and partly solved the perception problem. In situations that GPS is unavailable, or high accuracy of localization or mapping is needed, accurate and robust SLAM algorithms are required. SLAM is especially difficult for UAVs because of the strict requirement of real-time processing, and the limitation of UAV's payload.

1.1 Localization

During navigation, a robot's positions at discrete time instants are related by rigid motion transformation $T \in \mathbb{R}^{4\times4}$. Aiming to solve T and concatenate T into a trajectory, localization have always been a hot topic in both research and application, and the emergence of Global Positioning System (GPS) have partly solved this problem in outdoor environments. For GPS denied situations, solutions like Wifi/Ultra Wide Band (UWB) positioning, Motion Capture Camera System (VICON) are proposed. These solutions require extern hardwares, making them impractical or too expensive for most indoor applications. Therefore, on-board localization, which utilizes only on-board sensors to estimate positions, becomes a popular and feasible solution for most robots. Localization is especially indispensable for autonomous UAVs because they can not "stop" in the air like ground robots. Localization is sometimes called odometry in robotics. In particular, Visual Odometry (VO), i.e. odometry using cameras, is experiencing rapid growth recently.

1.2 Mapping

The significance of mapping comes mainly in three aspects. First of all, a map supports tasks like path planning and obstacle avoidance, which are basic requirements of autonomous navigation. Secondly, the map itself is the objective for many robot applications. Besides providing intuitive visualization, it allows further analysis of the explored space, including dimension evaluation, object recognition, etc. Thirdly, proper mapping will improve the accuracy and robustness of localization. One of the most important features of mapping is loop closure, which allows robots to recognize a place visited previously and optimize it's estimated trajectory accordingly, therefore drift is reduced or even rejected. Also, the ability of recognizing a place in the existing map enables robots to recover from odometry failure.

In the early ages, localization and mapping are considered separately, but later researchers found that SLAM is kind of a "chicken and egg" problem, i.e., localization and mapping are highly interrelated. A map is needed for accurate localization, while localization is needed for mapping. Or in another word, mapping or localization can be solved if one of them is known accurately. Therefore, current Visual Odometry algorithms typically include the mapping function as well, although the map may not be suitable for path planning or other applications. The only difference between modern odometry and SLAM is whether loop closure, or global map optimization is available [1].

Classical SLAM methods in Section 4 prefer to jointly estimate pose and map. Later more advanced methods in Section 5 usually employ the interleave idea of Parallel Tracking and Mapping (PTAM) [2] and put localization and mapping into two parallel threads.

2 RELATED WORKS

Proposed by the recent review by Cadena et al. [1], the history of SLAM can be roughly divided into three ages. In the classical age, 1986 - 2004, the mainstream of the community is the probabilistic formulation and filtering techniques. Reviews of first 20 years' research were published by Durrant

http://www.imavs.org/

Whyte and Bailey [3, 4]. The subsequent period, 2004-2015, is called the algorithm analysis age, where the fundamental properties, including observability, convergence and consistency, were investigated [5]. Many theories from the Computer Vision community, such as the SfM, were widely used by SLAM algorithms. Optimization based methods were developed, and believed to outperform the classical filtering based SLAM. The community was creating many open-source projects and pushing SLAM to practical applications. Currently the community is entering the third period, the robust perception age, where robust performance and high-level environment understanding are the focal points.

At the crossing of the algorithm analysis age and the robust perception future, emerging SLAM algorithms are marching towards the edge of real world application. The ORB-SLAM and LSD-SLAM published in the last two year are regarded as the most promising SLAM that can be applied on UAVs. Current reviews of SLAM are either outdated to include such recent progress [3–6], or focus on the theory development [1]. A survey introducing the real-time SLAM algorithms will provide the researchers in robotics with a clear picture, and help to apply or improve SLAM in the context of UAV.

This paper focuses on the milestones that can achieve real-time robust performance on UAV platforms, using either laser scanners or cameras. We starts by describing the modern architecture of SLAM in Section 3. Section 4 and 5 follows the anatomy of [1], to summarize the the classical and modern SLAM algorithms.

3 MODERN FRAMEWORK

A SLAM system can be partitioned into two parts, the front-end and back-end, shown in Figure 1. Usually the front-end requires techniques of computer vision and signal processing, in order to abstract the geometry information into mathematical model and feed it into the back-end. The back-end is in charge of optimizing the model, often called factor-graph, to refine the pose and map.

The sensor dependent front-end takes raw data from the sensors and pre-processes it. The pre-processing includes feature extraction, short term and long term data association, etc. In the case of featured-based visual SLAM in Section 5.2, detected image feature points are associated to 3D geometrical points. For direct tracking methods in Section 5.3, tracking between frames are conducted by front-end as well.

In Section 5, the output of front-end is usually modeled as a factor graph, where positions and space structures serve as nodes, and the rigid body transformation T as the edges connecting the nodes. Mathematically, this is a Maximum A Posteriori (MAP) problem formulated as 1. Measurements $Z = z_k : k = 1, \cdots, m$ are expressed as a function of $X = x_k : k = 1, \cdots, m$, $z_k = h_k(x_k) + \epsilon_k$, where x_k represents the unknown variable like position or space structure, i.e. the nodes in the factor graph. Under the assumption of zero mean Gaussian noise ϵ_k, (1) becomes (2), which can be optimized with iterative Gauss-Newton or Levenberg-Marquardt algorithms. The detailed formulation of the MAP problem can be found in [1].

$$X^* = \arg\max_X \mathbb{P}(X|Z) = \arg\max_X \mathbb{P}(Z|X)\mathbb{P}(X) \tag{1}$$

$$X^* = \arg\min_X \sum_{k=0}^{m} \frac{1}{2} \|h_k(x_k) - z_k\|^2_{\Omega_k} \tag{2}$$

For classical SLAM in Section 4, instead of constructing such MAP problem, nonlinear filtering approaches including Extended Kalman Filter (EKF) and Particle Filter (PF) are used to jointly estimate the position and map. A systematic review can be found in [3, 4].

According to some research [7], the accuracy of MAP estimation is believed to be better than that of the filtering method. But some advanced filtering systems may have equivalent performance as well, such as the Multi-State Constraint Kalman Filter [1].

4 FILTERING BASED METHODS

The idea of explicitly formulating SLAM as the probabilistic problem originated at the 1986 IEEE Robotics and Automation Conference [3]. Similar to the modern architecture of Section 3, the features and landmarks are extracted by the front-end, and the back-end is the joint estimation of localization and landmark maps. Later, EKF and PF were widely used in the back-end.

4.1 Probabilistic Formulation

At time instant k, the unknown variables are explicitly defined into position and orientation x_k, landmark location m_k. Control vector u_k defines the drive of transition from x_{k-1} to x_k. Measurement z_k is the observation of m_k at the state of x_k. In addition, the variable sets are defined in (3). Under the probabilistic formulation, the SLAM is implemented in an iterative time-update (4) and measurement update (5) fashion, where motion model is described by $P(x_k|x_{k-1}, u_k)$ and observation model described by $P(z_k|x_k, m)$. The visualization of the filtering inference is shown in Figure 2(a).

$$\begin{aligned} X_k &= \{x_0, x_1, \cdots, x_k\} = \{X_{k-1}, x_k\} \\ U_k &= \{u_0, u_1, \cdots, u_k\} = \{U_{k-1}, u_k\} \\ m &= \{m_1, m_2, \cdots, m_n\} \\ Z_k &= \{z_0, z_1, \cdots, z_k\} = \{Z_{k-1}, z_k\} \end{aligned} \tag{3}$$

$$\begin{aligned} &P(x_k, m|Z_{k-1}, U_k, x_0) = \\ &\int P(x_k|x_{k-1}, u_k) P(x_{k-1}, m|Z_{k-1}, U_{k-1}, x_0) dx_{k-1} \end{aligned} \tag{4}$$

$$\begin{aligned} &P(x_k, m|Z_k, U_k, x_0) = \\ &\frac{P(z_k|x_k, m) P(x_k, m|Z_{k-1}, U_k, x_0)}{P(z_k|Z_{k-1}, U_k)} \end{aligned} \tag{5}$$

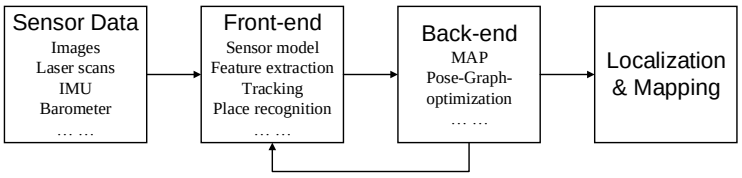

Figure 1: Modern architecture of SLAM consists of front-end and back-end.

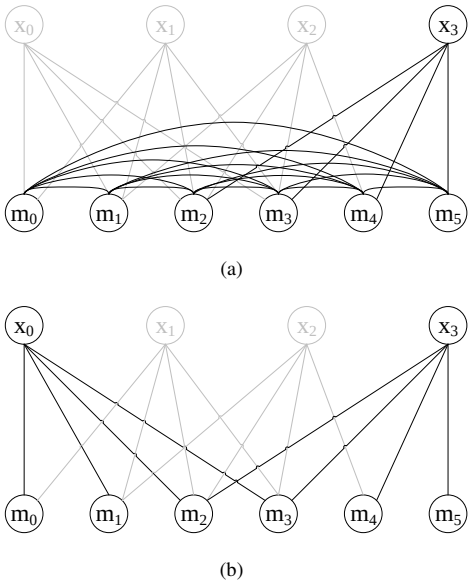

Figure 2: (a) Visualization of inference under the filtering framework. All landmarks m, and their correlation are maintained and updated. (b) Inference of the keyframe-based method. Only motion and map of keyframes are maintained.

4.1.1 EKF SLAM

The most intuitive way to simplify (4) and (5) is to employ EKF with the linear Gaussian assumption. Motion model and observation model are linearized so that EKF is applicable, and real-time performance can be achieved. However, the linear Gaussian assumption is usually not the practical case, and may lead to divergence, or significant estimation error. Also, EKF filtering is fragile to incorrect data association.

4.1.2 FastSLAM

To overcome the fragile linear assumption of EKF SLAM, FastSLAM was introduced by Montemerlo et al. [8]. Instead of linearizing the motion model $P(x_k|x_{k-1}, u_k)$, the non-Gaussian motion model is estimated with Monte Carlo sampling, i.e. Particle Filter. Direct sampling with m and X_k is computational infeasible because of their high dimension. Rao-Blackwellization is used to partition the joint estimation of m and X_k into product of independent Gaussian distributions, so that the sampling is greatly accelerated. The detailed derivation of FastSLAM can be found in [3, 8].

4.2 Laser SLAM

Proved by research like [9], the above probabilistic architecture is able to achieve practical performance with 2D laser scanner. A significant issue in filtering based laser SLAM, is the feature extraction and data association problem. Traditional features are modeled as lines, circles, corners, etc [10]. The major problem of such geometric features is that they are environment sensitive. For example, features relying on corners will fail in a mess environment, where walls can not be scanned by the laser. An improvement is the scan correlation, which regards the raw laser scan as the feature, and the alignment of the raw scans can be efficiently solved with Iterative Closest Point (ICP) [11].

In practice, the mapping between measurements and landmarks is rarely known [12]. In EKF SLAM, maximum likelihood estimation for each observation is commonly used to solve data association. Some enhanced algorithms solve the best association of all observations at the same time [13]. Compared to EKF, FastSLAM is much less prone to inaccurate data association. Actually, FastSLAM can be extended to sample on data associations, so that it can be estimated simultaneous with robot paths [12]. In [12], Montemerlo and Thrun showed an impressive SLAM result with FastSLAM using a single 2D laser. In the experiment of driving for over 4km, the position error was less than 10 meters. In conclusion, it is proved that FastSLAM is much better than EKF SLAM in terms of accuracy and robustness.

4.3 Visual SLAM

Compared to the success of filter based laser SLAM, vision-only SLAM remained unsolved, because of scarcity of computational power, lack of depth information, and the difficulty of extracting or associating features. Various filtering techniques, including EKF [14], Unscented Kalman Filter (UKF), Particle Filter, sparse information filter, were attempted under the filtering framework mentioned above.

Stereo cameras are used to overcome the depth problem [15]. Davison and Murray [15] achieved 5Hz SLAM using fixating active stereo, and proved that a small number of landmarks is enough to provide accurate pose estimation. In some

http://www.imavs.org/

http://www.imavs.org/

work, plane assumption was applied on downward looking cameras, to further simply the depth estimation.

Feature detector and descriptors like SIFT, SURF are widely used nowadays because of their robustness and convenience for data association, i.e. matching landmarks with image measurements. However, these features were computational too expensive before 2006. As the workaround, salient image patches, line segments were popular.

MonoSLAM by Davison et al. [14] is a significant milestone of Visual SLAM, because it is the first real-time monocular SLAM that achieve the speed of 30Hz with adequate accuracy and robustness for robotics. MonoSLAM is under the EKF frameworks where the state vector contains position, orientation, linear velocity, angular velocity, and the 3D position of landmarks. Using the famous good-feature-to-track detector by Shi and Tomasi, a landmark is represented by a salient image region, the depth of the patch, the orientation of its norm, and the uncertainty of its position. Efficient data association is achieved by searching features with a precalculated area, instead of in the whole image.

5 ANALYSIS BASED METHODS

Poor scalability is the main drawback of filtering SLAM. Because the joint distributions of all landmarks have to be maintained and updated all the time, the computational cost become unacceptable eventually, shown in Figure 2(a). This scalability problem is serious for Visual SLAM because images contain much more features than laser scans.

To mitigate such limitation, analysis based methods, or the so called optimization based methods, are proposed to maintain only a small subset of motions and maps. These subsets that represent the trajectory and map are called keyframes. Shown in Figure 2(b), although the number of nodes in the graph is larger, the interconnections between nodes remain sparse. Keyframe based graph can be optimized efficiently even if there are large number of motions and features. In computer vision, graph of Figure 2(b) is referred as Bundle Adjustment (BA). In many SLAM algorithms, the graph consists of only positions x. In that case, by applying the MAP formulation in Section 3, open-source libraries including g2o are able to optimize a graph with tens of thousands of nodes within one second.

By making a systematic comparison between filtering based and analysis based Visual SLAM methods, Strasdat et al. [7] came to the conclusion that keyframe based BA outperforms filtering SLAM in terms of accuracy, robustness and speed.

5.1 Laser SLAM

Research on 2D laser SLAM has been decreasing since 2006, mainly because of the success of algorithms like FastSLAM, lack of 3D information, and the significant progress of Visual SLAM. However, filtering based laser SLAM is mainly designed for 2D motions in structured environment. It is still impractical to apply it on UAV. New algorithms adopts the frontend-backend structure in Figure 1. Researchers concentrate on the front-end, while the back-end is kind of standardized as pose graph optimization. ICP [16] is a mature choice of front end, while it suffers from the high computational cost. Polar Scan Matching (PSM) utilizes the polar coordinates to perform scan matching. Normal Distribution Transform (NDT) aligns laser scans to a mixture of normal distributions.

Proposed by Kohlbrecher et al. in 2011 [17], HectorSLAM is currently one of the most competitive 2D laser algorithms. The front-end scan matching is achieved by aligning the laser scan with the map learned so far, using the Gauss-Newton approach. To improve convergence, pyramid like multi-resolution map is implemented with a coarse-to-fine matching scheme. Kohlbrecher et al. showed that their front-end was so accurate and robust that the back-end post-graph optimization could be neglected in most situations. To estimate the complete 6 DOF motion, a EKF is used to fuse information from sensors like IMU, with the 2D laser SLAM.

5.2 Featured Based Visual SLAM

Representing the environment with landmarks or features has been a popular choice since the 1980s when researchers began working on SLAM. Feature based map is compact, so that the requirement for computational power is significantly reduces. Also, both salient image patches and modern feature detectors are robust, and data association can be done efficiently with features. In the age of analysis based SLAM, feature detectors, including ORB [18–22], FAST [2], SURF [23], etc., were widely used in short-term tracking and loop closure.

The beginning of analysis SLAM is marked with the publish of PTAM [2], who first proposed to replace the filtering framework with a tracking thread and a mapping thread. PTAM maintains a global map composed of 3D points and the corresponding salient image patches. The tracking thread consists of data association and motion only BA. Data association, i.e. associating patches on a new frame with 3D points in map, is done by a coarse-to-fine search. After data association, motion only BA is conducted to refine the previously predicted motion. Simultaneously, the mapping thread utilizes the tracking result, to triangulate unmatched features into 3D points, which will be inserted into the global map. Then the local BA is applied to refine both the tracking result and the global map. The experiments showed that PTAM significantly outperforms the MonoSLAM. Ever since PTAM, most Visual SLAM algorithms adopted similar structure of separated tracking and mapping.

Strasdat et al. [24] presented a monocular SLAM algorithm where motion only BA was used for tracking. Within the keyframe framework, [24] proposed three dimensional information filters to initialize the 3D feature points. Loop closure were conducted with 7 DOF similarity transformation, where the scales of keyframes were considered. In [25],

http://www.imavs.org/

Strasdat et al. proposed the idea of covisibility graph, and tracking within a local map.

With stereo cameras, the depth of feature points can be triangulated, which greatly simplifies the tracking and mapping process. In libviso2 [23], SURF feature points are extracted with triangulated depth. Efficient feature point matching is achieved with coarse-to-fine searching. Visual odometry is done by minimizing the projection error of 3D feature points from the previous time instants into the current left and right frames. With the motion estimated by visual odometry, dense map construction is conducted with the ELAS stereo matching. With similar strategy, libfovis [26] achieved real-time SLAM on UAV with RGBD cameras, where loop closure was implemented using Calonder randomized tree descriptors and RANSAC.

In 2015, the paper [18] and source code of ORB-SLAM were published, and is believed to be the best feature based SLAM so far. ORB-SLAM is a complete system with functions of tracking, mapping, loop closure, relocalization from tracking lost. The ORB features, which can be extracted in milliseconds, are used throughout the system in all functions. Tracking is initialized from previous frame or global relocalization, and then refine with motion only BA in a local map extracted from the covisibility graph. Local mapping is done by inserting keyframes, optimizing using local BA, and culling keyframes. Loop closure candidates are searched using the DBoW2 approach, which is also used for relocalization from tracking lost. Similarity transformations between current frame and loop candidates are computed similar to the method of [24], and then fused into the covisibility graph and essential graph. Global map optimization is achieved over the essential graph. The ORB-SLAM has shown impressive performance with monocular and depth cameras.

5.3 Dense Direct Visual SLAM

Despite that feature based methods have proved themselves to be effective, they are faced with two serious problems. The first is that, the features are far too sparse to represent the environment, which hinders further applications with the environment, including semantic understanding, object recognition, human interaction, etc. Secondly, they make use of only the information of the features and discard all other pixels of the images. In 2011, dense methods were proposed to take advantages of all pixels, and build dense 3D map.

One of the pioneers is the Dense Tracking and Mapping (DTAM) by Newcombe et al [27], which achieved real-time performance with GPU support. Dense reconstruction is implemented between any keyframe and many other nearby frames. A regularized cost function is constructed with regard to the inverse depth, in order to consider both photometric error and the smoothness of the estimated depth. By introducing a coupling term, the smoothness part is optimized in a similar way of ROF image denoising, while the photometric error part can be optimized simply using exhaustive search. Paralleled with dense mapping, each RGB frame is tracked against the densely built map by minimizing the photometric error, in order to solve for the Lie algebra $se(3)$ that defines the 6 DOF motion.

The vision group from Technical University of Munich (TUM) published a series of paper to demonstrate their dense direct SLAM system. Kerl et al. [28] utilized the depth from RGBD camera, and used a similar optimization scheme with [27] to achieve odometry in 30Hz with CPU. In 2014, the Large Scale Direct Monocular SLAM (LSD-SLAM) was published by Engel et al. [29], and demonstrated impressive capacity of densely reconstructing the environment accurately. At the back-end, the map is continuously improved with pose-graph optimization, and loop closure is done with FAB-MAP [30]. Later, the stereo [31] and fisheye [32] extension of LSD-SLAM were released to make use of the known depth, or the wide view angle.

KinectFusion by Newcombe et al. [33] is another milestone for dense SLAM, though it is only real-time with GPU support. Different from previous Visual SLAM where map is represented with point clouds or grids, KinectFusion maintains a volumetric, truncated signed distance function (TSDF) representation. Tracking is conducted with pyramid ICP.

5.4 Semi-dense Methods

In between the feature based and direct SLAM, some researchers tried to take advantages of both methods. Semi-direct Visual Odometry (SVO) is a popular visual odometry algorithm put forward by Forster et al. [34]. The map is represented by 3D locations of FAST feature points, and the 4×4 image patches of the features. Whenever a keyframe is created, FAST features are extracted and inserted into a depth filter, which is in charge of updating depth and evaluating the uncertainty. In the tracking thread, there are 3 steps. The first is to initialize motion against the previous frame by minimizing the photometric error. In the second step, for each projected feature in the current frame, its location is refined by minimizing the photometric error between its patch and the closest keyframe's feature patch. In the third step, motion only BA or local BA is applied to refine the pose of the current frame. Because of the semi-direct scheme, SVO is so efficient that it can be implemented on embedded systems.

6 CONCLUSION

Over decades of development, 2D SLAM methods with laser scanners are considered mature. Modern algorithms like HectorSLAM [17] are adequately robust for UAV applications. Visual SLAM on UAV is still an open problem, mostly because algorithms are fragile to UAV's agile motion and the unknown uncertainty of the environment. Recent algorithms including ORB-SLAM and LSD-SLAM, are potential candidates that can be applied on UAV.

REFERENCES

[1] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, Jose Neira, Ian D Reid, and John J Leonard. Simultaneous

localization and mapping: Present, future, and the robust-perception age. *arXiv preprint arXiv:1606.05830*, 2016.

[2] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 225–234. IEEE, 2007.

[3] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[4] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam): Part ii. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.

[5] Gamini Dissanayake, Shoudong Huang, Zhan Wang, and Ravindra Ranasinghe. A review of recent developments in simultaneous localization and mapping. In *2011 6th International Conference on Industrial and Information Systems*, pages 477–482. IEEE, 2011.

[6] Josep Aulinas, Yvan R Petillot, Joaquim Salvi, and Xavier Lladó. The slam problem: a survey. In *CCIA*, pages 363–371. Citeseer, 2008.

[7] Hauke Strasdat, José MM Montiel, and Andrew J Davison. Visual slam: why filter? *Image and Vision Computing*, 30(2):65–77, 2012.

[8] 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.

[9] MWM Gamini Dissanayake, Paul Newman, Steve Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on robotics and automation*, 17(3):229–241, 2001.

[10] Juan Nieto, Tim Bailey, and Eduardo Nebot. Scan-slam: Combining ekf-slam and scan correlation. In *Field and service robotics*, pages 167–178. Springer, 2006.

[11] Feng Lu and Evangelos Milios. Robot pose estimation in unknown environments by matching 2d range scans. *Journal of Intelligent and Robotic Systems*, 18(3):249–275, 1997.

[12] Michael Montemerlo and Sebastian 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. IEEE, 2003.

[13] José Neira and Juan D Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on robotics and automation*, 17(6):890–897, 2001.

[14] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007.

[15] Andrew J Davison and David W Murray. Simultaneous localization and map-building using active vision. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):865–880, 2002.

[16] Zhengyou Zhang. Iterative point matching for registration of free-form curves and surfaces. *International journal of computer vision*, 13(2):119–152, 1994.

[17] Stefan Kohlbrecher, Oskar Von Stryk, Johannes Meyer, and Uwe Klingauf. A flexible and scalable slam system with full 3d motion estimation. In *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, pages 155–160. IEEE, 2011.

[18] Raul Mur-Artal, JMM Montiel, and Juan D Tardós. Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[19] Raúl Mur-Artal and Juan D Tardós. Probabilistic semi-dense mapping from highly accurate feature-based monocular slam. *Proceedings of Robotics: Science and Systems, Rome, Italy*, 1, 2015.

[20] Raúl Mur-Artal and Juan D Tardós. Orb-slam: Tracking and mapping recognizable features,. In *MVIGRO Workshop at Robotics Science and Systems (RSS), Berkeley, USA*, 2014.

[21] Raúl Mur-Artal and Juan D Tardós. Fast relocalisation and loop closing in keyframe-based slam. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 846–853. IEEE, 2014.

[22] Dorian Gálvez-López and Juan D Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, 2012.

[23] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 963–968. IEEE, 2011.

[24] Hauke Strasdat, JMM Montiel, and Andrew J Davison. Scale drift-aware large scale monocular slam. *Robotics: Science and Systems VI*, 2010.

[25] Hauke Strasdat, Andrew J Davison, JMM Montiel, and Kurt Konolige. Double window optimisation for constant time visual slam. In *2011 International Conference on Computer Vision*, pages 2352–2359. IEEE, 2011.

[26] Albert S Huang, Abraham Bachrach, Peter Henry, Michael Krainin, Daniel Maturana, Dieter Fox, and Nicholas Roy. Visual odometry and mapping for autonomous flight using an rgb-d camera. In *International Symposium on Robotics Research (ISRR)*, volume 2, 2011.

[27] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *2011 international conference on computer vision*, pages 2320–2327. IEEE, 2011.

[28] Christian Kerl, Jürgen Sturm, and Daniel Cremers. Robust odometry estimation for rgb-d cameras. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3748–3754. IEEE, 2013.

[29] J. Engel, T. Schöps, and D. Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision (ECCV)*, September 2014.

[30] Mark Cummins and Paul Newman. Fab-map: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[31] J. Engel, J. Stueckler, and D. Cremers. Large-scale direct slam with stereo cameras. In *International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[32] D. Caruso, J. Engel, and D. Cremers. Large-scale direct slam for omnidirectional cameras. In *International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[33] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011.

[34] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. Svo: Fast semi-direct monocular visual odometry. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15–22. IEEE, 2014.

http://www.imavs.org/