ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/318795640

Analysis of ROS-based Visual and Lidar Odometry for a Teleoperated Crawler-type Robot in Indoor Environment

Conference Paper · July 2017

DOI: 10.5220/0006420603160321

CITATIONS
0

READS
62

3 authors:

Maxim Sokolov
Innopolis University

3 PUBLICATIONS **4** CITATIONS

SEE PROFILE

Oleg Bulichev
Innopolis University

1 PUBLICATION **0** CITATIONS

SEE PROFILE

Ilya Afanasyev
Innopolis University

50 PUBLICATIONS **83** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project

Laser-camera systems and algorithms to sense the environment for robotic applications View project

Project

Development of methods for identification of interaction models of robot with adaptive compliance with the human and environment View project

All content following this page was uploaded by Ilya Afanasyev on 31 July 2017.

The user has requested enhancement of the downloaded file.

Analysis of ROS-based Visual and Lidar Odometry for a Teleoperated Crawler-type Robot in indoor environment

Maxim Sokolov[1], Oleg Bulichev[1] and Ilya Afanasyev[1]
[1]*Institute of Robotics, Innopolis University, Universitetskaya str. 1, 420500 Innopolis, Russia*
(m.sokolov, o.bulichev, i.afanasyev)@innopolis.ru

Keywords: Monocular SLAM, ROS, visual odometry, lidar odometry, crawler robot, ORB-SLAM, LSD-SLAM

Abstract: This article presents a comparative analysis of ROS-based monocular visual odometry, lidar odometry and ground truth-related path estimation for a crawler-type robot in indoor environment. We tested these methods with the crawler robot "Engineer", which was teleoperated in a small-sized indoor workspace with office-style environment. Since robot's onboard computer can not work simultaneously with ROS packages of lidar odometry and visual SLAM, we used online computation of lidar odometry, while video data from onboard camera was processed offline by ORB-SLAM and LSD-SLAM algorithms. As far as crawler robot motion is accompanied by significant vibrations, we faced some problems with these visual SLAM, which resulted in decreasing accuracy of robot trajectory evaluation or even fails in visual odometry, in spite of using a video stabilization filter. The comparative analysis shown that lidar odometry is close to the ground truth, whereas visual odometry can demonstrate significant trajectory deviations.

1 INTRODUCTION

Over the last decade visual odometry has become the valuable tool for estimation of a vehicle's pose, orientation and trajectory through analysis of corresponding onboard camera images recorded during vehicle motion. However, visual odometry methods are sensitive to illumination conditions and can fail in case of insufficiency of visual features since a scene requires enough texture to let explicit motion be estimated. It makes essential to combine visual odometry with other measurements like wheel odometry, lidar odometry, global positioning system (GPS), inertial measurement units (IMUs), etc. (Scaramuzza and Fraundorfer, 2011), (Zhang and Singh, 2015), (Sarvrood et al., 2016). Nevertheless, the other types of onboard odometry can have own drawbacks. For instance, lidar odometry at robot motion includes fluctuations in positions of point clouds over time. Therefore, onboard sensor odometry should be provided with ground truth[1].

Since we utilize a single onboard camera, we are interested in monocular visual odometry, which is characterized by computing both relative vehicle motion and 3D scene structure from 2D video data. Very often, researchers define three categories of monocular visual odometry: (1) feature-based, (2) appearance-based (also known as direct), and hybrid methods (Scaramuzza and Fraundorfer, 2011). Appearance-based methods estimate intensity of all image pixels with the following direct image-to-image alignment, whereas feature-based methods extract appreciable and repeatable features that can be tracked through the frames. Finally, hybrid methods apply to a combination of both previous approaches. We need to emphasize that in our investigation we exploit visual simultaneous localization and mapping (V-SLAM) methods, which are originally focused on calculating a global map and robot path while tracking onboard camera position and orientation. However, in our case we study a robot motion in small indoor workspace, thereby we extend the results of V-SLAM to visual odometry, neglecting a difference between these definitions and considering the workspace map as a global map. In our research we use both feature-based and appearance-based methods for robot path recovery by choosing two monocular V-SLAM: ORB-SLAM (Mur-Artal et al., 2015) and LSD-SLAM (Engel et al., 2014) recognized as enhanced and robust methods. Both of these methods have demonstrated good results in tracking, mapping, and camera localization for outdoor applications, but there are some uncertainties with robust-

[1]"Ground truth" is defined as a reference tool or a set of measurements that are known to be much more accurate than measurements from a system under investigation

ness and feasibility to indoor robot navigation in typical office-style environment with monotone-painted walls (Buyval et al., 2017). Except V-SLAM we also provided ROS-based onboard lidar odometry in online mode, getting more accurate information about robot motion trajectory. Since crawler robot motion suffers significant shakes, vibrations and sharp turns, it brings deviations in onboard sensor outputs, decreasing accuracy of trajectory evaluation or even failing visual odometry. Therefore, robot path verification is provided by an external camera-based monitoring system, which is not affected by vibrations.

Our main goal of this study is to compute a crawler-type robot trajectory via different odometry methods realized in ROS[2], providing: (1) visual odometry acquisition for robot motion within the workspace by using onboard camera and two V-SLAM packages: feature-based ORB-SLAM and direct LSD-SLAM; (2) comparison of onboard visual and lidar odometry; (3) a verification of both visual and lidar odometry with the real robot trajectory measured by the external camera-based monitoring system as the ground truth. To summarize, this paper analyzes and compares different monocular visual odometry with lidar odometry and the ground truth-based robot path estimation. Since onboard computer of our robot did not permit a simultaneous run of ROS packages for lidar and two visual SLAM, we launched ROS-based lidar odometry online and recorded onboard and external cameras' video in synchronous mode. Then, we processed onboard video offline with ORB and LSD-SLAM, comparing visual and lidar odometry with ground truth-based robot trajectories.

The rest of the paper is organized as following. Section 2 introduces system setup, Section 3 presents ROS-based visual and lidar odometry, and Section 4 describes indoor tests and robot trajectories evaluation. In Section 5 we analyze visual and lidar odometry estimation. Finally, we conclude in Section 6.

2 SYSTEM SETUP

2.1 Robot system configuration

The crawler-type robot "Engineer" (Fig. 1) is designed and manufactured by "Servosila" company[3] to overcome complex terrain while navigating in confined spaces and solving a number of challenging tasks for search and rescue missions. The robot has tracks, flippers, and a robotic arm, that support capabilities for climbing stairs, traversing doorways and narrow passages, negotiating obstacles, leveling itself from sideways/upside down positions, etc. Moreover, the robotic arm is equipped by a head with sensors, controllers and grippers, that allows to have a flexible remote control for lifting heavy loads, opening different doors, plus grasping, pushing or pulling objects. The robot sensors pack can contain of a laser scanner, an optical zoom camera, a thermal vision camera, a pair of stereo cameras, IMU and GPS navigation tools. The detailed information about our set of robot vision system is presented in Table 1.

Since our robot' sensor pack does not have a built-in laser scanner, we mounted the scanning laser rangefinder Hokuyo URG-04LX-UG01[4] on the robot head (see, Fig. 2). This laser is often used for autonomous robots, as far as it has light weight (160g), low-power consumption (2.5W), convenient for indoor application scanning range from 2 cm to 5.6 m, wide field of view up to 240°, angular resolution of 0.36°, refresh rate of 10Hz, and accuracy up to 3% relatively to measured distance.

2.2 Robot vision system

To validate onboard visual and lidar data, we used external ground truth system based on Basler acA2000-50gc camera[5] (see, characteristics in Table 2), which was hung up under the workspace on the height of about 3m and connected to PC (Fig. 3). Before tests the camera was calibrated against a chessboard with "*camera_calibration*" ROS package[6].

[2]Robot Operating System (ROS), which is a set of software libraries and tools for robot applications, www.ros.org

[3]"Servosila" company, www.servosila.com/en/

[4]Hokuyo Automatic Co. 2D lidar, www.hokuyo-aut.jp

[5]Area Scan Camera from Basler AG company, www.baslerweb.com/en/products/cameras/

[6] "*camera_calibration*" ROS package is based on OpenCV library: wiki.ros.org/camera_calibration

Figure 1: The Servosila "Engineer" crawler robot. Courtesy of Servosila company

Table 1: The Servosila "Engineer" robot vision system

Type	Quantity	Direction	Resolution	Palette	Night vision	Zoom	Rate
Mono	one	front	1280*720	RGB	No	Yes	60fps
Stereo pair	two	front	640*480	YUYV	Yes	No	30fps
Mono	one	rear	640*480	YUYV	No	No	30fps

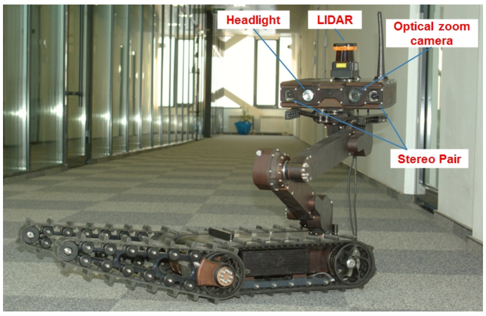


Figure 2: The "Engineer" robot head's sensor system

Figure 3: Workspace with the ground truth camera above

Table 2: Basler acA2000-50gc camera characteristics

Parameter	Configuration
Image Sensor	CMV2000 CMOS
Video resolution	2MP, 2046*1086
Frame rate	50 fps
Mono/Color	Color
Shutter	Global shutter

3 ROS-BASED VISUAL AND LIDAR ODOMETRY

3.1 ROS-based ORB SLAM odometry

Oriented FAST (Rosten and Drummond, 2006) and Rotated BRIEF (Calonder et al., 2010) algorithm, which form ORB SLAM[7] method is a feature-based real-time SLAM library for monocular, stereo and RGB-D cameras (Mur-Artal et al., 2015). It is able to build a sparse 3D scene and to compute an on-board camera trajectory, thereby recovering a vision-based robot odometry. In cases of heterogeneous environment, ORB-SLAM demonstrates robustness to complex motion clutter, performing wide baseline loop detection and real time automatic camera re-localization. To process live monocular streams, a library with a ROS node is used.

The ORB-SLAM adapts main ideas of previous SLAM works, such as Parallel Tracking and Mapping (PTAM, (Klein and Murray, 2007)) and Scale Drift-Aware Large Scale Monocular SLAM (Strasdat et al., 2010). Thus, it uses advanced approaches to localization, loop closing, bundle adjustment, keyframe selection, feature matching, point triangulation, camera localization for every frame, and relocalization after tracking failure. Moreover, ORB-SLAM surpasses PTAM algorithm, providing camera tracking with ORB-features extraction (Rublee et al., 2011), scale-aware loop closing, co-visibility information for large scale operation, occlusion handling, and invariance to viewpoint at relocalization (Mur-Artal et al., 2015). ORB-SLAM key properties include:

- the same features for tracking, mapping, re-localization and loop closing;
- real time loop closing, which is based on the optimization of a pose graph;
- invariant relocalization of real time camera to viewpoint and illumination.

ORB-SLAM is a real-time SLAM library in ROS, which provides both necessary calculations and graphical user interface (GUI) for visualization of a camera trajectory, a sparse 3D reconstruction, and real-time features on video frames. To launch ORB-SLAM algorithm in ROS, the following operations should be executed:

- building a node for Monocular SLAM;
- running Monocular Node:
  rosrun ORB_SLAM2 Mono path_to_vocabulary path_to_webcam_settings_file

[7]ORB-SLAM2 package is available at https://github.com/raulmur/ORB_SLAM2

Monocular SLAM node maintains three parallel threads: Tracking (to help in a camera localization), Local Mapping (to build a new map) and Loop Closing (to obtain a closed-loop trajectory).

3.2 ROS-based LSD-SLAM odometry

Large-Scale Direct Monocular SLAM[8] (LSD-SLAM) creates a real-time global, semi-dense map in a fully direct mode without using keypoints, corners or any other local features (in opposite to feature-based methods like ORB-SLAM). Direct featureless tracking is performed by image-to-image alignment using a coarse-to-fine approach with a robust Huber weights (Engel et al., 2014). Depth is estimated only using pixels near image boundaries and semi-dense maps (which are denser than maps of feature-based methods) are created continuously. LSD-SLAM forms both a camera trajectory and a semi-dense 3D scene reconstruction, where a global mapping is built by performing a pose graph optimization. The advantage of LSD-SLAM over feature-based methods is in reconstruction of a more complete 3D scene with textured smooth surfaces, which could be missed by feature-based algorithms.

LSD-SLAM is a fully direct method, which is capable to build real-time large-scale semi-dense maps using an onboard computer. LSD-SLAM is launched in ROS with:

- Starting the camera driver:
  roslaunch webcam_camera usb_camera.launch
- Launching the LSD-SLAM viewer:
  rosrun lsd_slam_viewer viewer
- Initiation of the main node:
  rosrun lsd_slam_core live_slam
  /image:=/usb_cam_node/image_raw
  /camera_info:=/usb_cam_node/camera_info

Note that LSD-SLAM consists of two ROS packages: (1) *lsd_slam_core* that includes full SLAM system, which provides live camera operation *live_slam* using ROS input/output, and (2) *lsd_slam_viewer*, which is optionally used for 3D visualization.

3.3 ROS-based lidar odometry

To obtain ROS-based lidar odometry for "Engineer" robot motion within indoor workspace, we used the open source *hector_slam* package[9], which contains of modules related to SLAM execution in unstructured

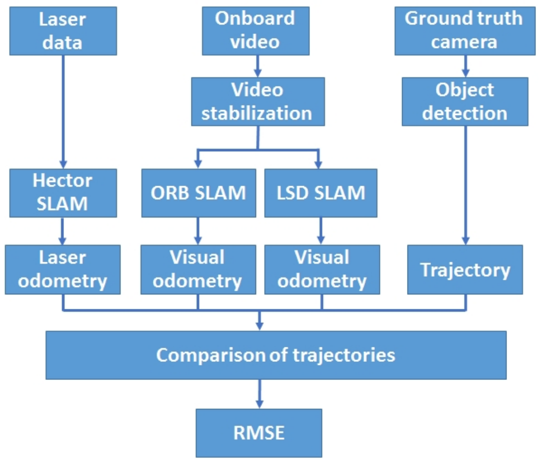


Figure 4: The block-scheme of the "Engineer" robot trajectory evaluation with onboard sensors and ground truth

environments (Kohlbrecher et al., 2013). Its algorithm relies on fast lidar data scanning and matching at full lidar refresh rate. To stabilize the laser scanner data, *hector_slam* combines with an attitude estimation and an optional pitch and roll angles, calculating probable locations and environment maps even for rugged ground terrain (Kohlbrecher et al., 2013). To recover a trajectory of moving robot with lidar data, the ROS module *hector_trajectory_server*[10] keeps tracking of multiple coordinate frames over time.

3.4 Indoor tests with crawler-type robot

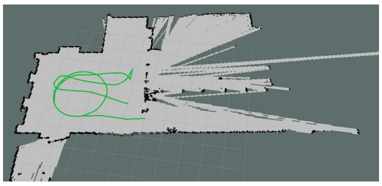
Our goals in these indoor tests with the crawler robot are: (1) to calculate visual odometry from the robot motion within the workspace by using two ROS-based V-SLAM packages: ORB and LSD-SLAM; (2) to obtain onboard lidar odometry; (3) to verify the visual and lidar odometry with a trajectory calculated by video processing of the external ground truth camera. The block-scheme of the "Engineer" robot path evaluation with onboard sensors and ground truth is shown in Fig. 4

At the initial stage of our experiments we put the external ground truth camera above the workspace, and calibrated both onboard and external cameras with a chessboard at different locations. Our tests were performed with a human-operated crawler-type robot "Engineer" followed a close-loop trajectory in a small-size indoor workspace with office-style environment and partially glass walls (Fig. 5a). The data was recorded with onboard sensors: 2D lidar and high-resolution camera with a global shutter and a

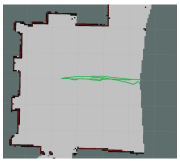
[8] LSD-SLAM package is available at https://github.com/tum-vision/lsd_slam

[9] ROS package *hector_slam* is available at https://github.com/tu-darmstadt-ros-pkg/hector_slam

[10] ROS package *hector_trajectory_server* is a part of *hector_slam* package, wiki.ros.org/hector_trajectory_server



(a) Test 1: complex robot motion



(b) Test 2: forth-and-back motion

Figure 5: Lidar map for different robot trajectories (green color) with ROS *hector_slam* package in RViz viewer

frame rate of 60 fps. Since robot's onboard computer can not work simultaneously with ROS packages of lidar odometry and two visual SLAM, we launched ROS-based *hector_slam* package online and recorded onboard and external cameras' video in synchronous mode. Then, we processed onboard video offline with ORB and LSD-SLAM, comparing visual and lidar odometry with ground truth-based robot trajectories. The main drawbacks of the crawler robot motion are significant vibration and sharp turns that typically lead to poor visual SLAM results or fails of these ROS packages. To minimize vibration effects on visual odometry, we provided a series of experiments, moving robot forth-and-back at slow speed in teleoperation mode without turns to either side (Fig. 5b). However, in spite of our precautions, strong vibration forced us to stabilize onboard video during post-processing stage offline by OpenCV video stabilization filter described in (Grundmann et al., 2011).

4 ANALYSIS OF VISUAL AND LIDAR ODOMETRY

Lidar odometry. To process Hokuyo 2D Laser Scanner data, we used ROS-based *hector_slam* package, which calculated 2D map of our indoor environment and robot trajectory with the following visualization in RViz[11] (Fig. 5). However, glass walls nearby the workspace were transparent for lidar (Fig. 5a) that can create a problem for autonomous navigation of the robot in such type of indoor environment.

ORB-SLAM tests. Since ORB-SLAM is feature-based method, its 3D point cloud is very sparse. However, ORB-SLAM performs map reconstruction over selected keyframes with calculation of camera positions after every recall in all video dataset (see, Fig. 6), building as well 2D robot motion trajectory.

LSD-SLAM tests. In our experiments with crawler robot, LSD-SLAM has frequently failed in conditions of camera vibrations and sharp turns, even after video stabilization by the OpenCV filter.

[11] RViz is 3D visualization tool for ROS, wiki.ros.org/rviz

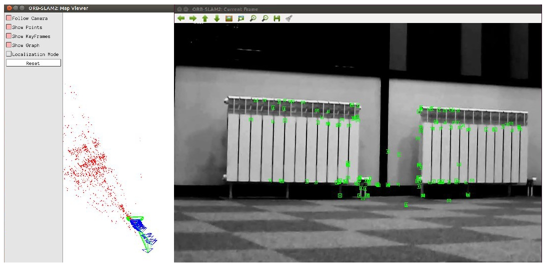


Figure 6: ORB-SLAM map viewer with 3D point cloud (left), and an image with extracted features (right)

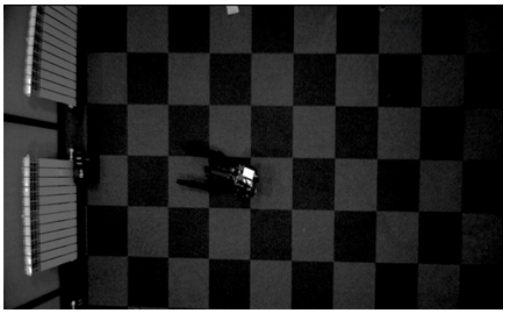


Figure 7: The contrasting white label on the robot's head from the ground-truth camera

Ground truth-based path evaluation. To monitor the robot motion from the height of about 3m, we used the external high resolution camera Basler (see, Fig. 3), thus established ground truth. To clearly identify the robot motion within the workspace, we placed white label on the robot's head. Thereby, from a grayscale video of the ground truth camera we easily recognized the contrasting white label and robot contour (see, Fig. 7), computing the robot geometrical center and its corresponding trajectory.

Analysis of trajectories. Finally, we computed and scaled visual and lidar-based trajectories for the following comparative analysis. Typical trajectories are shown in Fig. 8, where green, red and blue (dot) curves mean robot path evaluation calculated by monocular ORB-SLAM, ground truth camera and lidar data correspondingly. We used the root mean square error (RMSE) as metrics of robot trajectory evaluation accuracy. The robot path evaluated by external ground truth camera was used as a base path to rescale all other trajectories in terms of ground truth axes. Therefore, we developed a software to estimate a correspondence between every point of the base path and the nearest points of another trajectory by computing the square root of mean of squared distances between these points. Thus, the maximum RMSE reached 1.96 cm between the ground truth and lidar-based trajectories, and 12.6 cm between the ground truth and ORB-SLAM trajectories.

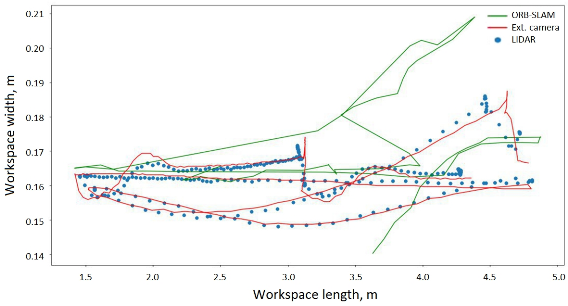


Figure 8: The robot trajectories computed by monocular ORB-SLAM (green curve), external ground-truth camera (red curve), and lidar data (blue dot curve)

5 CONCLUSION AND FUTURE WORK

In this paper we analyze and compare robot trajectories acquired by onboard 2D lidar and monocular camera, and evaluated by ROS-based visual odometry, lidar odometry and the external ground truth camera. We used two visual SLAM methods for the robot path recovery: feature-based ORB-SLAM and appearance-based LSD-SLAM, because they usually demonstrate good results in tracking, mapping, and camera localization. Our tests were performed with a human-operated crawler-type robot "Engineer" followed a close-loop trajectory in a small-sized indoor workspace with office-style environment and partially glass walls. Since onboard computer of our robot can not work simultaneously with ROS packages of lidar odometry and two visual SLAM, we used ROS-based *hector_slam* package online and recorded onboard and external cameras' video in synchronous mode. Then, we processed onboard video offline with ORB and LSD-SLAM, comparing visual and lidar odometry with ground truth-based robot trajectory. The main drawbacks of the crawler-type robot motion are significant vibration and sharp turns that result in poor results or even fails in ROS-based visual SLAM packages. To minimize vibration effects on visual odometry, we moved robot forward and backward at slow speed in teleoperation mode without sharp turns to either side, and used OpenCV video stabilization filter offline as post-processing stage. However, in spite of our precautions the vibrations were so strong that LSD-SLAM odometry frequently failed and lost the robot trajectory during its motion.

The comparative analysis of trajectories computed by ORB-SLAM, lidar and external ground truth camera data allowed to make the following conclusions: (1) lidar odometry is close to the ground truth path evaluation; (2) ORB-based visual odometry continues working in spite of strong camera vibration during crawler motion, but its trajectory shows significant deviations in comparison with the ground truth.

Our future plans deal with realization of other visual SLAM and odometry methods, providing tests in more complex environment and improving experimental technique for better accuracy estimation.

REFERENCES

Buyval, A., Afanasyev, I., and Magid, E. (2017). Comparative analysis of ros-based monocular slam methods for indoor navigation.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). *BRIEF: Binary Robust Independent Elementary Features*, pages 778–792. Springer Berlin Heidelberg, Berlin, Heidelberg.

Engel, J., Schöps, T., and Cremers, D. (2014). *LSD-SLAM: Large-Scale Direct Monocular SLAM*, pages 834–849. Springer International Publishing, Cham.

Grundmann, M., Kwatra, V., and Essa, I. (2011). Auto-directed video stabilization with robust l1 optimal camera paths. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 225–232, Washington, DC, USA. IEEE Computer Society.

Klein, G. and Murray, D. (2007). Parallel tracking and mapping for small ar workspaces. In *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 225–234.

Kohlbrecher, S., Meyer, J., Graber, T., Petersen, K., Klingauf, U., and von Stryk, O. (2013). Hector open source modules for autonomous mapping and navigation with rescue robots. In *Robot Soccer World Cup*, pages 624–631. Springer.

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

Rosten, E. and Drummond, T. (2006). *Machine Learning for High-Speed Corner Detection*, pages 430–443. Springer Berlin Heidelberg, Berlin, Heidelberg.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571.

Sarvrood, Y., Hosseinyalamdary, S., and Gao, Y. (2016). Visual-lidar odometry aided by reduced imu. *ISPRS International Journal of Geo-Information*, 5(1):3.

Scaramuzza, D. and Fraundorfer, F. (2011). Visual odometry [tutorial]. *IEEE Robotics Automation Magazine*, 18(4):80–92.

Strasdat, H., Montiel, J., and Davison, A. J. (2010). Scale drift-aware large scale monocular slam. *Robotics: Science and Systems VI*.

Zhang, J. and Singh, S. (2015). Visual-lidar odometry and mapping: low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2174–2181.

View publication stats