Comparison of ROS-Based Monocular Visual SLAM Methods: DSO, LDSO, ORB-SLAM2 and DynaSLAM

Eldar Mingachev[1(✉)], Roman Lavrenov[1(✉)], Tatyana Tsoy[1(✉)], Fumitoshi Matsuno[2], Mikhail Svinin[3], Jackrit Suthakorn[4], and Evgeni Magid[1]

[1] Laboratory of Intelligent Robotic Systems (LIRS), Intelligent Robotics Department, Higher Institute for Information Technology and Intelligent Systems, Kazan Federal University, Kazan, Russian Federation
ermingachev@stud.kpfu.ru, {lavrenov,tt,magid}@it.kfu.ru
[2] Department of Mechanical Engineering and Science, Kyoto University, Kyoto 615-8540, Japan
matsuno@me.kyoto-u.ac.jp
[3] Information Science and Engineering Department, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan
svinin@fc.ritsumei.ac.jp
[4] Biomedical Engineering Department, Mahidol University, 4, 999 Phuttamonthon, Salaya 73170, Thailand
jackrit.sut@mahidol.ac.th
http://robot.kpfu.ru/eng

Abstract. Stable and robust path planning of a ground mobile robot requires a combination of accuracy and low latency in its state estimation. Yet, state estimation algorithms should provide these under computational and power constraints of a robot embedded hardware. The presented study offers a comparative analysis of four cutting edge publicly available within robot operating system (ROS) monocular simultaneous localization and mapping methods: DSO, LDSO, ORB-SLAM2, and DynaSLAM. The analysis considers pose estimation accuracy (alignment, absolute trajectory, and relative pose root mean square error) and trajectory precision of the four methods at TUM-Mono and EuRoC datasets.

Keywords: Simultaneous localization and mapping · Visual SLAM · Monocular SLAM · Visual odometry · State estimation · Path planning · Benchmark testing · Robot sensing systems

1 Introduction

Simultaneous localization and mapping (SLAM, [8]) is an ability of an autonomous vehicle to start in an unknown location of an unknown environment and then, using only relative observations, to incrementally construct a

A. Ronzhin et al. (Eds.): ICR 2020, LNAI 12336, pp. 222–233, 2020.
https://doi.org/10.1007/978-3-030-60337-3_22

map of the environment [25] while simultaneously using the map to compute a bounded estimate of the vehicle location [22]. Nowadays, SLAM is applied to state and pose estimation problems in various domains, from virtual and augmented reality to autonomous vehicles and robotics [12,15]. The field has reached a mature level [7] that causes proprietary SLAM algorithms utilizing in many commercial products as well as public availability of a number of open-source SLAM software packages [6]. Yet, due to sensor price and robot weight concerns, currently the prevailing type of SLAM is a monocular approach [5].

One of the main features of a monocular SLAM is a scale-ambiguity [10], which states that a world scale could not be observed and drifts over time, being one of the major error sources. Being both a challenge and a benefit, it allows switching seamlessly between differently scaled environments [14], while stereo or depth cameras do not allow such flexibility, having a limited range where they can provide reliable measurements [21]. This paper offers a comparative analysis in terms of a pose estimation accuracy and a trajectory precision of the most recent and popular robot operating system (ROS) based open-source monocular SLAM methods considering power constraints of mobile ground robots [18]. The four selected SLAM methods are DSO [9], LDSO [13], ORB-SLAM2 [19,20] and DynaSLAM [2]

2 Related Work

2.1 The Selected SLAM Methods

Direct Methods can estimate a completely dense reconstruction by a direct minimization of a photometric error and optical flow regularization. Some direct methods focus on high-gradient areas estimating semi-dense maps [2]. The presented study compares:

- DSO, which is a state-of-the-art pure direct method [9],
- LDSO, which is DSO's latest revision with a loop closure ability and a global map optimization [13].

Feature-Based Methods rely on matching key points and can only estimate a sparse reconstruction [3], mostly providing a good trade-off between an accuracy and a runtime. The current study presents a comparison of:

- ORB-SLAM2 [20] state-of-the-art visual SLAM method that tracks ORB features in real-time. It has a same monocular core as the original ORB-SLAM [19] but is featured with an improved and optimized workflow.
- The recently proposed DynaSLAM [2] method, which adds a front-end stage to the ORB-SLAM2 system to have a more accurate tracking and a reusable map of a scene. It outperforms the accuracy of the standard visual SLAM baselines in highly dynamic scenarios.

2.2 Benchmarks

There are several publicly available datasets for the SLAM benchmark purposes, however, some of the existing ones are not suitable to benchmark monocular SLAM algorithms due to a low precision of groundtruth data [16]. The current study considers the two most suitable datasets, TUM-Mono and EuRoC.

TUM-Mono. Schubert et al. [24], Engel, Usenko, and Cremers [11] developed a dataset for evaluating a tracking accuracy of a monocular visual odometry [17] and SLAM methods. The dataset includes 50 indoor and outdoor sequences, which start and end in the same position and contain groundtruth only for these start and end trajectory segments. All dataset sequences are photometrically calibrated and provide exposure times for each frame as reported by a sensor, a camera response function, and a dense lens attenuation factors. This allows evaluating a tracking accuracy via an accumulated drift and a reliably benchmark direct methods.

EuRoC. Burri et al. [4] proposed a visual-inertial dataset aiming at evaluation of localization and 3D environment reconstruction algorithms. The dataset consists of 11 sequences, recorded with two monocular cameras onboard a micro-aerial vehicle. The datasets range from slow flights under good visual conditions to dynamic flights with motion blur and poor illumination. Each sequence contains synchronized stereo images, extrinsic and intrinsic calibrations, an inertial unit (IMU) measurements, and an accurate groundtruth (approximately 1 mm) recorded using a laser tracker and a motion capture system. Compared to the TUM-Mono benchmark, the sequences in EuRoC are shorter and have less variety as they only contain recordings inside a single machine hall and a single laboratory room.

2.3 Metrics

TUM-Mono. To evaluate the TUM-Mono benchmark results, we used proposed by Engel, Usenko, and Cremers metrics [11], an *alignment Root Mean Square Error (RMSE)* - a combined error measure, which equally takes into account an error caused by scale, rotation and translation drifts over an entire trajectory. It is the RMSE between tracked trajectories when aligned to start and end segments.

EuRoC. The EuRoC includes entire groundtruth camera trajectories, which allows using an *absolute trajectory RMSE (ATE)*, a measure of a global trajectory accuracy, and a *relative pose RMSE (RPE)*, which is a measure of a local pose accuracy, proposed by Sturm, Engelhard, Endres, Burgard, and Cremers [26]. Overall, the RPE metric provides an elegant way to combine rotational and translational errors into a single measurement, while the ATE only considers translational errors. As a result, the RPE is always slightly larger than the ATE (or equal if there is no rotational error). However, rotational errors typically also manifest themselves in wrong translations and are thus indirectly also captured by the ATE. From a practical perspective, the ATE has an intuitive visualization, which facilitates a visual inspection. Nevertheless, as the authors noted, the two metrics are strongly correlated.

Trajectory Detail Level. In contrast to the reviewed metrics, which mainly focus on measuring a difference between corresponding frames, a trajectory detail level measures a difference between a length of trajectories, being an estimated and a groundtruth trajectories length ratio. The metric can be used to benchmark effectiveness of hardware capabilities usage and even estimate limits of a detail level of a particular SLAM algorithm while running on various hardware configurations. In addition, it could be useful in determining a suitable trade-off between an accuracy and output data detail level.

3 Comparative Analysis

Mur-Artal and Tardós [20] proposed running each sequence five times and showed median results to account for a non-deterministic nature of a system. Bescos, Fácil, Civera and Neira [2] extended this approach by increased the number of runs up to 10 times, as dynamic objects are prone to increase a non-deterministic effect. In light of the above, the current study also utilizes the extended approach.

3.1 Hardware Setup

This study focuses on SLAM methods usage with mobile ground robots that implies a restriction on energy consumption and absence of strict constraints on a mobile robot weight, which, for example, are critical for SLAM usage with a UAV. The selected hardware platform with balanced computational resources and power consumption is the *HP Omen 15-ce057ur* laptop with the technical specifications briefly described in Table 1.

Table 1. Hardware specifications.

CPU	Intel Core i7-7700 HQ, 2800 MHz
RAM	16 GB, DDR 4, 2400 MHz
Weight	2.56 kg
Battery	70 Wh Li-Ion
Power consumption	80 W (avg. load)

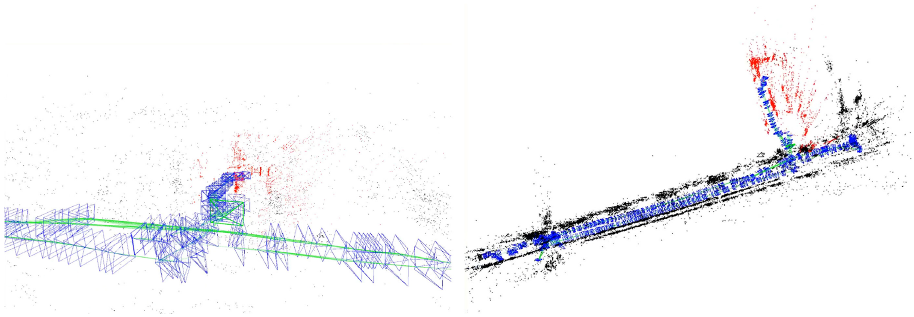

Fig. 1. The experimental sequence - the loop start and end (left) and the global trajectory overview (right).

3.2 TUM-Mono

To prove the effectiveness of the proposed approach [11], we expanded the TUM-Mono dataset sequences with a new real-world sequence (Fig. 1) collected with PAL Robotics TIAGo Base ground mobile robot [1] with a single monocular camera onboard [23].

The sequence presents 13 min of video and about a 100-m length trajectory in a gradually changing environment - from a narrow indoor corridor to a wide indoor corridor, which moved the robot from illuminated scenes to dark scenes. The sequence starts and ends in the same place with slow loopy motion allowing a correct initialization of the SLAM algorithms. The groundtruth for the entire trajectory was recorded with the ORB-SLAM2 [20] algorithm, in contrast to the other sequences groundtruth, which was provided by LSD-SLAM [10] only for the start and end segments.

We have evaluated the metrics over DSO, LDSO, ORB-SLAM2, and DynaSLAM methods on the expanded TUM-Mono dataset, running the dataset sequences forward and backward, with the loop closure feature being disabled, following the dataset authors' recommendations.

Figure 2 presents the cumulative error graphs – accumulated translational, rotational, and scale drifts along with the RMSE when aligning the estimated trajectory start and end segments with the provided groundtruth trajectory. The figure depicts the number of runs in which the errors are below the corresponding x-values - the closer to the top left, the better. It is important to note the difference in magnitude – the RMSE within start and end segments is about 100 times less than the alignment RMSE.

Due to the groundtruth nature and the similarity of the experimental results, Engel et al. [11] concluded that almost all of the alignment errors originate not from the noise in the groundtruth, but from the accumulated drift. Our experiments confirmed this conclusion, which means that these metrics could be used for any benchmark with a groundtruth of any accuracy as a reference, even the one collected with SLAM algorithms. Figure 3 shows the color-coded alignment RMSE ranging from 0 (blue) to 10 m (red) for each dataset sequence.

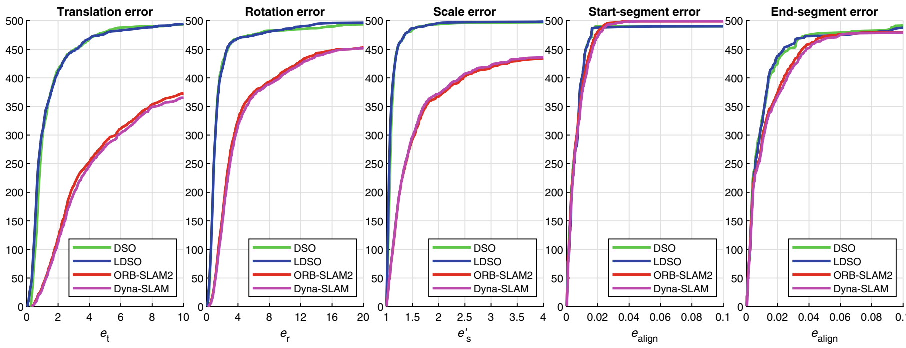

Fig. 2. Accumulated translational (e_t, m), rotational (e_r, m), and scale (e'_s, m) drifts along with the start and end segments RMSE (e_{align}, m).

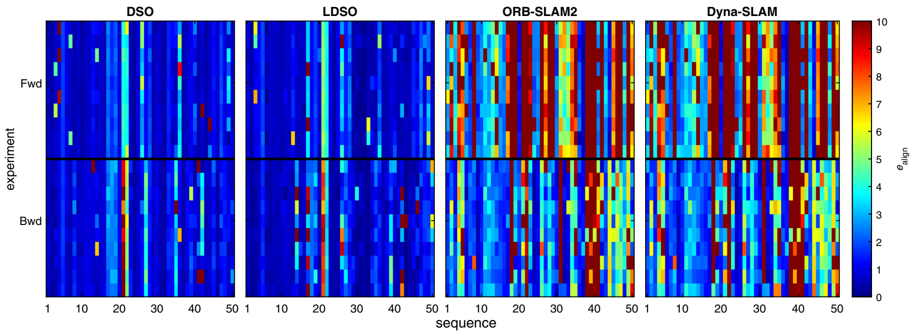

Fig. 3. Color-coded alignment RMSE (e_{align}, m) for each TUM-Mono dataset sequence. (Color figure online)

The experiments demonstrated that direct methods provide outstanding results comparing to the feature-based ones - the TUM-Mono dataset is designed especially for direct methods benchmark purposes, providing full photometric data for each frame, which greatly improves the accuracy of such methods. However, there is not that much of a difference if comparing DSO to LDSO - as we can assume, the LDSO global map optimization slightly improves the overall accuracy of the base method.

The same behaviour is observed while comparing feature-based methods - the accuracy of DynaSLAM is slightly lower comparing to ORB-SLAM2. However, the DynaSLAM initialization is always quicker than the ORB-SLAM2 initialization; in highly dynamic sequences, the ORB-SLAM2 initialization only occurs when moving objects disappear from a scene while DynaSLAM succeeds in bootstrapping the system in such dynamic scenarios.

3.3 EuRoC

We evaluated the metrics over DSO, LDSO, ORB-SLAM2, and DynaSLAM on the EuRoC dataset over all sequences for each of the two camera streams, which were interpreted as separate sequences with the same groundtruth ('.0' and '.1' notations correspond to the first and the second camera dataset respectively and are labeled on X-axis in Figs. 4 and 5). Figure 4 shows the calculated absolute trajectory RMSE (ATE, measured in metres) and the relative pose RMSE (RPE, measured in metres per second) metrics ranging from 0 (blue) to 2 (red) for all methods.

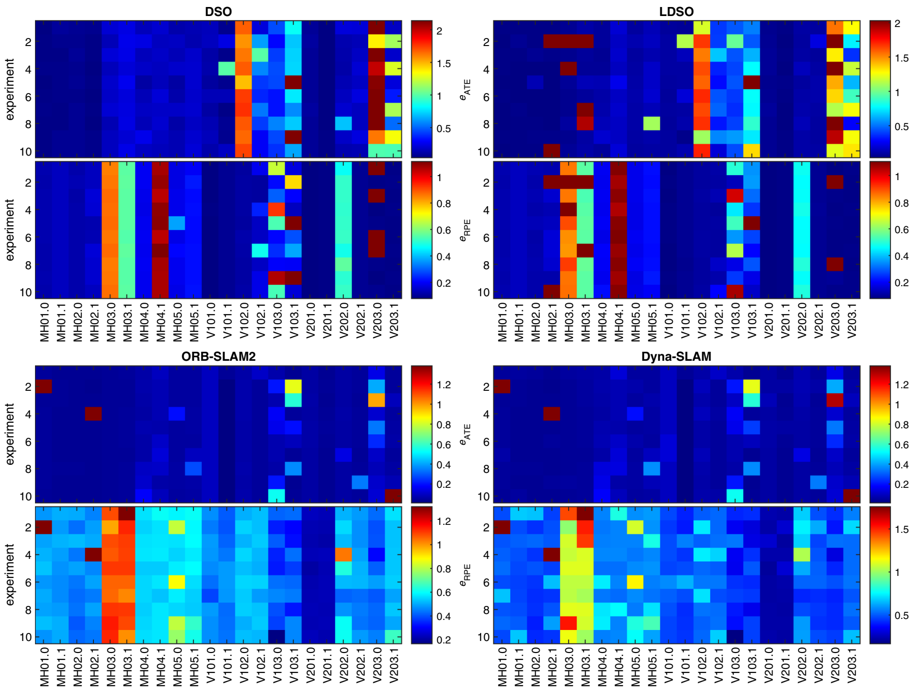

Fig. 4. Color-coded evaluation results for each EuRoC dataset sequence: absolute trajectory RMSE (ATE, m) and relative pose RMSE (RPE, m/s). (Color figure online)

As the analysis demonstrates, that in terms of the RPE, the measure of local accuracy, the direct methods generally perform significantly better than the feature-based ones, but it is still difficult for them to overcome a harsh environment with a lack of light and prevailing rotational movements (over translational movements), as shown in MH.05, V1.03, and V2.03 sequences. In terms of the ATE, the feature-based methods demonstrate a stable performance, even in the "hard" sequences. However, the accuracy of DynaSLAM is slightly lower, compared to ORB-SLAM2, since DynaSLAM succeeds in bootstrapping the system with a dynamic content and always initializes quicker than ORB-SLAM2 and thus has more frames to process (and more room for accumulating errors). Figure 5 demonstrates the calculated trajectory detail level for DSO, LDSO, ORB-SLAM2, and DynaSLAM methods for each EuRoC dataset sequence.

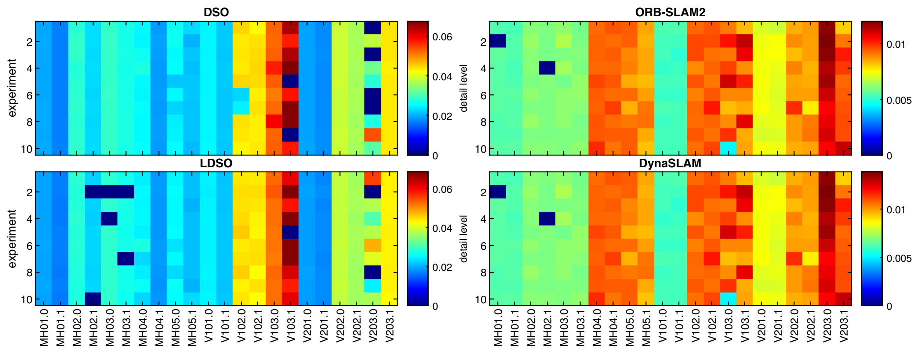

Fig. 5. Color-coded trajectory detail level for each EuRoC dataset sequence. (Color figure online)

Table 2. Median absolute trajectory RMSE (ATE, m), relative pose RMSE (RPE, m/s) & trajectory detail level ($Detail$) for each EuRoC dataset sequence.

Sequence	DSO			LDSO			ORB-SLAM2			DynaSLAM		
	ATE	RPE	Detail	ATE	RPE	Detail	ATE	RPE	Detail	ATE	RPE	Detail
MH.01	0.054	0.132	0.018	0.044	0.131	0.018	0.041	0.491	0.006	0.042	0.494	0.006
MH.02	0.063	0.134	0.025	0.044	0.139	0.025	0.035	0.458	0.006	0.036	0.465	0.007
MH.03	0.209	0.711	0.028	0.090	0.706	0.028	0.041	1.095	0.006	0.043	1.102	0.007
MH.04	0.173	0.632	0.022	0.136	0.642	0.022	0.074	0.560	0.009	0.076	0.568	0.011
MH.05	0.169	0.199	0.023	0.127	0.198	0.023	0.054	0.589	0.009	0.056	0.592	0.010
V1.01	0.104	0.088	0.023	0.099	0.089	0.023	0.054	0.454	0.005	0.054	0.459	0.006
V1.02	1.047	0.137	0.044	1.013	0.111	0.044	0.054	0.528	0.009	0.055	0.534	0.011
V1.03	0.584	0.334	0.057	0.607	0.375	0.057	0.091	0.409	0.009	0.097	0.411	0.010
V2.01	0.064	0.081	0.018	0.058	0.081	0.019	0.047	0.225	0.007	0.047	0.227	0.008
V2.02	0.162	0.306	0.037	0.106	0.281	0.038	0.051	0.508	0.009	0.053	0.552	0.009
V2.03	1.439	0.087	0.036	1.266	0.086	0.040	0.096	0.477	0.010	0.097	0.479	0.012

The obtained results suggest that the direct methods typically distinguish more keyframes and, thus, having a better trajectory detail level, show a better local (pose) accuracy, compared to feature-based methods. It is important to note the difference in the color scale of the feature-based methods plots, which is the difference in the detail level magnitude. DynaSLAM operates a slightly larger amount of frames than ORB-SLAM2 and has a slightly better trajectory detail level (due to the quicker initialization).

Table 3. Median alignment RMSE (*Align*, m), absolute trajectory RMSE (*ATE*, m), relative pose RMSE (*RPE*, m/s) & trajectory detail level (*Detail*).

Metrics	DSO	LDSO	ORB-SLAM2	DynaSLAM
Align	0.8496	0.7769	5.7571	6.1891
ATE	0.1683	0.1062	0.0507	0.0525
RPE	0.1373	0.1111	0.4765	0.4787
Detail	0.0355	0.0376	0.0086	0.0099

3.4 Summary

Tables 2 and 3 summarize the calculated metrics as a single median value for each EuRoC dataset sequence (Table 2) and the entire TUM-Mono and EuRoC datasets (Table 3).

The alignment error, accumulated drift, is in average 7.35 times lower for direct methods than for indirect:

- LDSO outperforms DSO by 8.56%
- DynaSLAM is 6.98% behind ORB-SLAM2

The absolute trajectory error, global accuracy, for direct methods is in average 2.66 times higher than for indirect:

- LDSO outperforms DSO by 36.89%
- DynaSLAM is 3.42% behind ORB-SLAM2

The relative position error, local accuracy, for direct methods is in average 3.85 times lower than for indirect:

- LDSO outperforms DSO by 19.08%
- DynaSLAM is 0.46% behind ORB-SLAM2

The level of trajectory detail for direct methods is in average 3.95 times higher than for indirect:

- LDSO outperforms DSO by 5.59%,
- DynaSLAM outperforms ORB-SLAM2 by 13.13%.

While it was expected that LDSO should outperform its original source algorithm (DSO) and experiments demonstrated its better performance with regard to all measured criteria, DynaSLAM and ORB-SLAM2 have varying benefits with respect to particular criteria, and this variety should be considered when selecting a SLAM algorithm for a specific task.

4 Conclusions and Future Work

This paper presented a comparative analysis of four publicly available ROS-based monocular SLAM algorithms in terms of the state estimation accuracy and the trajectory detail level. The analysis demonstrated that the direct methods DSO and LDSO have a better accuracy while having entire photometric data available and mainly focus on the local accuracy, which is also indirectly proven by the fact that they save and operate a relatively large amount of trajectory keyframes. For these reasons, DSO and LDSO are more suitable for tasks involving a short-range operation and requiring high accuracy in a local pose estimation, e.g.., a 3D-reconstruction of an environment. At the same time, the feature-based methods ORB-SLAM2 and DynaSLAM outperform the direct ones in terms of the global accuracy, which, combined with an average trajectory detail level, makes them a universal solution for most SLAM tasks - especially the ones that require a long-range operating with stable and reliable results throughout an entire trajectory. In tasks that involve dynamic objects the accuracy of DynaSLAM will be significantly higher than ORB-SLAM2.

The current study used HP Omen 15-ce057ur laptop hardware. However, any SLAM method performance strongly correlates with available computational resources. Our ongoing work deals with expanding the obtained results and comparing the four algorithms' performance using several different robots of the Laboratory of intelligent robotic systems [1,18]. We strongly believe that such comparison could be useful to the research community in order to have a better perspective of how each metric varies depending on availability of real robots' onboard computational resources.

Acknowledgements. The reported study was funded by the Russian Foundation for Basic Research (RFBR), according to the research project No. 19-58-70002. The second author acknowledges the support by the research grant of Kazan Federal University. The forth and the fifth authors acknowledge the support of the Japan Science and Technology Agency, the JST Strategic International Collaborative Research Program, Project No. 18065977. The sixth author acknowledges the support of the National Science and Technology Development Agency (NSTDA), Thailand, Project ID FDA-CO-2562-10058-TH. Special thanks to PAL Robotics for their kind professional support with TIAGo Base robot software and hardware related issues.

References

1. Bereznikov, D., Zakiev, A.: Network failure detection and autonomous return for PMB-2 mobile robot. In: International Conference on Artificial Life and Robotics (ICAROB), pp. 444–447 (2020)
2. Bescos, B., Fácil, J.M., Civera, J., Neira, J.: DynaSLAM: tracking, mapping, and inpainting in dynamic scenes. IEEE Robot. Autom. Lett. **3**(4), 4076–4083 (2018)
3. Bokovoy, A., Yakovlev, K.: Sparse 3D point-cloud map upsampling and noise removal as a vSLAM post-processing step: experimental evaluation. In: Ronzhin, A., Rigoll, G., Meshcheryakov, R. (eds.) ICR 2018. LNCS (LNAI), vol. 11097, pp. 23–33. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99582-3_3

4. Burri, M., et al.: The EuRoC micro aerial vehicle datasets. Int. J. Robot. Res. **35**(10), 1157–1163 (2016)
5. Buyval, A., Afanasyev, I., Magid, E.: Comparative analysis of ROS-based monocular SLAM methods for indoor navigation. Ninth International Conference on Machine Vision, ICMV, vol. 10341, p. 103411K. International Society for Optics and Photonics (2017)
6. Carballo, A., Takeuchi, E., Takeda, K.: High density ground maps using low boundary height estimation for autonomous vehicles. In: 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 3811–3818. IEEE (2018)
7. Delmerico, J., Scaramuzza, D.: A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots. In: IEEE 2018 International Conference on Robotics and Automation, ICRA, pp. 2502–2509. IEEE (2018)
8. Dissanayake, M.G., Newman, P., Clark, S., Durrant-Whyte, H.F., Csorba, M.: A solution to the simultaneous localization and map building (SLAM) problem. IEEE Trans. Robot. Autom. **17**(3), 229–241 (2001)
9. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 611–625 (2017)
10. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: large-scale direct monocular SLAM. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8690, pp. 834–849. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10605-2_54
11. Engel, J., Usenko, V., Cremers, D.: A photometrically calibrated benchmark for monocular visual odometry. arXiv preprint arXiv:1607.02555 (2016)
12. Gabdullin, A., Shvedov, G., Ivanou, M., Afanasyev, I.: Analysis of onboard sensor-based odometry for a quadrotor UAV in outdoor environment. In: International Conference on Artificial Life and Robotics (ICAROB) (2018)
13. Gao, X., Wang, R., Demmel, N., Cremers, D.: LDSO: direct sparse odometry with loop closure. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pp. 2198–2204. IEEE (2018)
14. Ibragimov, I.Z., Afanasyev, I.M.: Comparison of ROS-based visual SLAM methods in homogeneous indoor environment. In: 2017 14th Workshop on Positioning, Navigation and Communications (WPNC), pp. 1–6. IEEE (2017)
15. Lavrenov, R., Matsuno, F., Magid, E.: Modified spline-based navigation: guaranteed safety for obstacle avoidance. In: Ronzhin, A., Rigoll, G., Meshcheryakov, R. (eds.) ICR 2017. LNCS (LNAI), vol. 10459, pp. 123–133. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66471-2_14
16. Martínez-García, E.A., Rivero-Juárez, J., Torres-Méndez, L.A., Rodas-Osollo, J.E.: Divergent trinocular vision observers design for extended Kalman filter robot state estimation. Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng. **233**(5), 524–547 (2019)
17. Martinez Garcia, E.A.: 4WD robot posture estimation by radial multi-view visual odometry. Instituto de Ingeniería y Tecnología (2018)
18. Moskvin, I., Lavrenov, R., Magid, E., Svinin, M.: Modelling a crawler robot using wheels as pseudo-tracks: model complexity vs performance. In: 7th International Conference on Industrial Engineering and Applications (ICIEA), pp. 235–239. IEEE (2020)
19. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans. Robot. **31**(5), 1147–1163 (2015)
20. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. IEEE Trans. Robot. **33**(5), 1255–1262 (2017)

21. Nagahama, K., Nishino, T., Kojima, M., Yamazaki, K., Okada, K., Inaba, M.: End point tracking for a moving object with several attention regions by composite vision system. In: International Conference on Mechatronics and Automation, pp. 590–596. IEEE (2011)
22. Rodriguez-Telles, F.G., Mendez, L.A.T., Martinez-Garcia, E.A.: A fast floor segmentation algorithm for visual-based robot navigation. In: 2013 International Conference on Computer and Robot Vision, pp. 167–173. IEEE (2013)
23. Safin, R., Lavrenov, R., Tsoy, T., Svinin, M., Magid, E.: Real-time video server implementation for a mobile robot. In: 2018 11th International Conference on Developments in eSystems Engineering (DeSE), pp. 180–185. IEEE (2018)
24. Schubert, D., Goll, T., Demmel, N., Usenko, V., Stückler, J., Cremers, D.: The TUM-mono VI benchmark for evaluating visual-inertial odometry. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pp. 1680–1687. IEEE (2018)
25. Simakov, N., Lavrenov, R., Zakiev, A., Safin, R., Martínez-García, E.A.: Modeling USAR maps for the collection of information on the state of the environment. In: 2019 12th International Conference on Developments in eSystems Engineering (DeSE), pp. 918–923. IEEE (2019)
26. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of RGB-D SLAM systems. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 573–580. IEEE (2012)