On Autonomous Spatial Exploration with Small Hexapod Walking Robot using Tracking Camera Intel RealSense T265

Jan Bayer Jan Faigl

***Abstract*— In this paper, we report on the deployment of the combination of commercially available off-the-shelf embedded visual localization system and RGB-D camera in an autonomous robotic exploration performed by small hexapod walking robot. Since the multi-legged walking robot is capable of traversing rough terrains, the addressed exploration problem is to create a map of an unknown environment while simultaneously performing the traversability assessment of the explored environment to efficiently and safely reach next navigational waypoints. The proposed system is targeted to run onboard of small multi-legged robots, and therefore, the system design is focused on computationally efficient approaches using relatively lightweight components. Therefore, we take advantages of the recently introduced tracking camera Intel RealSense T265 and RGB-D camera Intel RealSense D435 that are deployed to our developed autonomous hexapod walking robot that is equipped with adaptive locomotion control. Together with the proposed computationally efficient data representation and traversability assessment, the developed system supports onboard mapping and online decision-making within the exploration strategy even on a platform with low computational capabilities. Based on the reported experimental evaluation of the tracking camera, the developed system provides sufficiently accurate localization, and the robot has been able to explore indoor and outdoor environments fully autonomously.**

I. INTRODUCTION

Spatial robotic exploration is a problem to create a map of the reachable area by a mobile robot. Many approaches, such as those mentioned in the survey [1], address the exploration by extending the idea of the frontier-based exploration introduced in [2]. Frontiers are borders between known and unknown parts of the environment and represent locations towards which robots can be navigated to acquire new information about unexplored parts of the environment [3].

The explored environment can be represented by the spatial map but also by more complicated models if the purpose of the exploration is to measure an additional physical quantity or if the exploration is driven by a different mechanism than frontiers, e.g., by entropy [4]. In addition to spatial mapping, traversability assessment [5] is an important part of the autonomous decision-making for exploration of rough terrains and unstructured environments that can be found in search and rescue [6] or extraterrestrial missions [7], [8].

Regarding such missions, not only the particular exploration approach affects the mission performance, but also

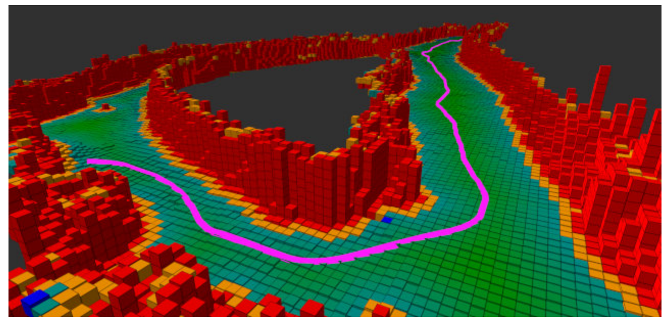


Fig. 1. Visualization of the elevation map and path planned.

the particular robot type might influence the navigation, localization precision, and exploration capabilities. Various robotic platforms can be used, but multi-legged robots can be considered as a prominent robot type for operating in rough terrains because of their locomotion capabilities [9].

The fundamental part of an autonomous robotic exploration system is related to the sensor equipment and computational resources that are also related to the energy sources that need to be available on-board of the robot. Multi-legged walking robots have been already deployed in robotic exploration tasks, and reports on existing solutions can be found, e.g., in [10]. Possible utilization of existing approaches for localization and mapping needed in autonomous exploration is related to the capacity and size of the robot.

Generally, sufficiently large robots such [11], but also LAURON V [12] and ANYmal [13], have paid load capacity to be equipped with relatively large and heavy equipment (powerful computers, laser range finders, and appropriate power source). On the other hand, smaller multi-legged robots, such as Messor [14] or CRABOT [15] have limited options for the possible equipment. Thus the development of the autonomy for these robots has to take into account these limitations, which might not allow running state-of-the-art approaches like ORB-SLAM2 [16] at high frame rates using only onboard computationally resources.

We consider a small hexapod walking robot with the body of size $10\,\mathrm{cm} \times 20\,\mathrm{cm}$, and paid load capacity $1\,\mathrm{kg}$ (including battery) employed with adaptive locomotion control in autonomous exploration missions. Therefore, our main design focus is on development of computationally efficient solutions for the full navigation system including localization, mapping, planning, and decision-making in exploration missions that are targeted to be run onboard of the robot. We take advantages of the recently introduced Intel RealSense T265 device that is deployed as the main localization system that is accompanied by the RGB-D camera Intel

Authors are with the Czech Technical University, Faculty of Electrical Engineering, Technicka 2, 166 27, Prague, Czech Republic {bayerja1|faiglj}@fel.cvut.cz

The presented work has been supported by the Czech Science Foundation (GAČR) under research project No. 19-20238S. The authors acknowledge the support of the OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics".

978-1-7281-3605-9/19/$31.00 ©2019 IEEE

RealSense D435 for mapping of the robot surroundings. The new sensory equipment enabled us to build an elevation map to support computationally efficient online decision-making in exploration missions held in rough terrains. Based on the experimental results of the developed system in indoor and outdoor environments, the system is capable of operating fully autonomously, and therefore, the herein reported results are mainly motivated to share the gained experience and demonstrate capabilities of the relatively small multi-legged robot with the nowadays sensory system.

The rest of the paper is organized as follows. A brief overview of the related work is presented in the following section to support the made selection of the used approaches. The developed system is described in Section III and achieved results in three autonomous exploration scenarios are reported in Section IV, together with an overview of the performance properties of the developed solution. The concluding remarks and ideas for future work are in Section V.

II. Related Work

The presented deployment of the Intel RealSense T265 in the autonomous exploration follows the frontier-based exploration [2], which can be considered as a de facto standard approach for the robotic exploration. Several extensions and improvements of the regular frontier-based exploration have been proposed in the literature, e.g., see surveys in [17], [1]. The spatial exploration relies on mapping and localization of the robot during the autonomous mission, and therefore, explicit consideration of the pose estimation in the exploration might improve the precision of the build map, e.g., using adaptive sensing to maximize the map information combined with minimization of the uncertainty in the robot pose estimation [18]. In [4], the authors employ entropy to combine information about the robot pose and environment to trade-off pose estimation improvements and mapping by a selection of the next navigational waypoint in a unified way.

Although advanced exploration strategies can be employed with the embedded localization sensor [19], the herein presented evaluation aims to show the performance of the localization sensor in the autonomous exploration regardless of the localization, and thus the localization is considered as an independent system that should provide the best possible pose estimation during the whole exploration mission. Therefore, we follow the conventional frontier-based exploration [2] but improving techniques to decrease computational requirements are employed to enable deployment of the exploration system on a small hexapod walking robot.

The frontier-based exploration is principally composed of: mapping, detection of the frontiers in the map, determination of the possible next goal locations and selection of the particular navigational goal towards which the robot is navigated [20]. All the particular parts can have different implementations, but most important is a suitable data representation of the map. The traditional probabilistic approach for a map representation [21] is the occupancy grid [22] which is, however, unable to capture the uneven shape of the terrain traversable by a walking robot. On the other hand, the full 3D extension such as the OctoMap [23] can be computationally (and memory) demanding in comparison to the elevation map as it is reported in [24]. The elevation map has been successfully employed in exploration with walking robots, e.g., in [5], and authors of [25] further extend the elevation map by a cost map based on distance from obstacles to avoid the risk of driving robot close to obstacles. Based on these deployments, we chose an elevation map representation and also the cost map approach [25] in the developed exploration system.

In our work, the Intel RealSense T265 is intended as a replacement of the vision-based localization with lightweight sensors that can be carried on small robots. The state-of-the-art feature-based visual localization ORB-SLAM2 [16] has been reported in several works [26], [27] as relatively accurate. On the other hand, one of the main advantages of T265 is its energy effectiveness, especially in comparison to the traditional solution using visual SLAM and dedicated computational power.

Regarding deployment of the ORB-SLAM2 [16] in different scenarios [27], [28], we experienced it requires at least CPU with the computational power equivalent to the Intel i5 to run about 7 Hz, which we found sufficient to localize the employed hexapod walking robot [27]. Hence, the thermal dissipation power is about 15 W in such a case, which requires input power in tens of watts. Contrary to that, T265 has input power only 1.5 W, and it does not require any additional computational power. Therefore, comparison of T265 and ORB-SLAM2-based localization system is reported in Section IV-A.

III. Developed Autonomous Exploration System

The embedded localization system has been integrated into the developed system for autonomous exploration that consists of four main modules: the *sensors*, *mapping module*, *exploration module*, and *path following module*. Since the utilized sensors the Intel RealSense D435 and T265 directly provide point clouds and localization, respectively, we do not explicitly consider a localization module. However, the *mapping module* is needed to build an elevation map of the environment from the provided robot pose and point clouds. The *exploration module* then uses the build map and robot pose to determine possible next goal locations to explore unknown parts of the environment. Based on the employed exploration strategy, the next navigational goal is

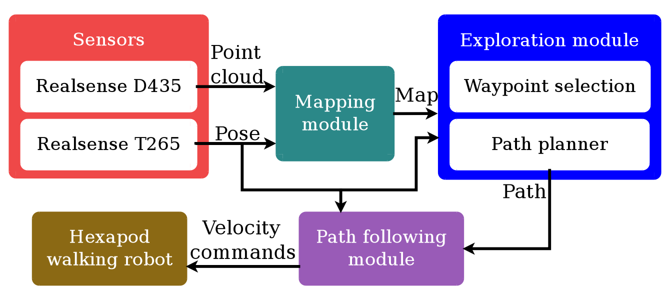


Fig. 2. Architecture of the proposed system

determined, and a path is planned that is then passed to the *path following module*, which is responsible for controlling the robot towards the next waypoint safely. The exploration system is developed in ROS [29], and an overview of the system architecture is visualized in Figure 2. The individual modules are further described in the rest of this section.

1) Sensors: The RGB-D camera Intel RealSense D435 (further referred as D435) is utilized for collecting 3D point cloud of the nearby surroundings of the robot that is directly used with the pose estimation of the robot in the mapping module. The localization is powered by the Intel RealSense T265 (further denoted T265) that runs onboard visual SLAM which takes advantage of stereo fisheye camera and fusion with the inertial measurement unit (IMU).

2) Mapping module: In the *mapping module*, the captured point cloud is integrated into an elevation map using the estimated robot pose. The elevation map is represented as a grid map $map(i, j)$ where each cell (i, j) represents a height of the corresponding terrain area. The map is built incrementally from the point clouds provided by D435, in which points not belonging to the top modeled surface of the terrain are filtered out. Similar to [30], [31], [24], one dimensional Kalman filter is utilized for merging new measurements of the terrain height h_k and its variance $\sigma^2_{h,k}$ of the k-th input point cloud. Thus, a new measured height z_k with the variance $\sigma^2_{z,k}$ is estimated utilizing the linear sensor model [30], and it is fused with the height h_{k-1} of the elevation map at the corresponding location (i, j) by

$$h_k = \frac{\sigma^2_{z,k} h_{k-1} + \sigma^2_{h,k-1} z_k}{\sigma^2_{z,k} + \sigma^2_{h,k-1}} \tag{1}$$

$$\sigma^2_{h,k} = \frac{\sigma^2_{z,k}\sigma^2_{h,k-1}}{\sigma^2_{z,k} + \sigma^2_{h,k-1}}. \tag{2}$$

Cells of the elevation map that correspond to not yet explored parts are marked as *unknown*.

The build elevation map is further assessed to mark *untraversable* parts of the environment based on the heights of the cells. The map cell $map(i, j)$ is considered traversable if local height differences $g_h(i, j)$ are lower than the threshold g_{max} estimated from the particular robot kinematics, where $g_h(i, j)$ is determined as

$$\begin{aligned} g_h(i, j) = \max(\{&|h(i, j) - h(i-1, j)|,\\ &|h(i, j) - h(i+1, j)|,\\ &|h(i, j) - h(i, j-1)|,\\ &|h(i, j) - h(i, j+1)|\}). \end{aligned} \tag{3}$$

Then, the untraversable cells are grown by the radius of the robot shape circumference to consider the physical dimensions of the robot. Moreover, we follow the cost map approach [25], and the distance transform [32] is applied to compute cost $d(i, j)$ of each traversable cell (i, j) based on its distance to the closest untraversable or unknown cells.

3) Exploration module is responsible for repeated determination of the next goal location towards which the robot is navigated. We follow the frontier-based exploration [2], and frontier cells are determined in the elevation map as traversable cells that are incident with unknown cells. Nearby frontier cells are clustered into similarly sized sets, and a single representative of each set is determined as in [33]. The representatives are further considered as possible goal locations from which the next goal location is selected according to the shortest expected path from the current robot location to the goal. The next goal is determined using A* with the heuristic function computed as the Euclidean distance to the waypoint. The travel cost between two neighboring nodes n and n' is computed using the eight-neighborhood that is increased by the cost $d(n)$ to penalize paths close to the obstacles. Non-negative cost $d(n)$ decreases with the D_8 distance from the closest non-traversable cell. However, the selection of the next waypoint is focused to the nearby area of the robot, and therefore, a possible goal location is considered unreachable if a path is not found in less than 20 000 expansions of A*.

The *path following module* follows the determined path (if any), and determination of the new navigational goal and path is triggered if the goal location is reached or after $T_{exp} = 8\,\mathrm{s}$. The exploration terminates if no new goal location is determined.

4) Path following module is an independent process for steering the robot motion to follow the path planned by the *exploration module*. The path is considered as a sequence of waypoints that are progressively processed, and for the current closest waypoint, the robot forward and angular velocities are determined to steer the robot towards the waypoint. The next waypoint from the sequence is processed if the robot gets less than $1\,\mathrm{cm}$ far from the current waypoint.

IV. Experimental Results

The localization performance of T265 has been firstly evaluated and compared with the ORB-SLAM2 in a dedicated laboratory experiment with the hexapod walking robot shown in Fig. 3. Then, T265 has been deployed in one indoor and two outdoor exploration scenarios.

The exploration framework described in Section III has been implemented in C++ using ROS [29] and deployed on the hexapod walking robot with the computational environment based on the Intel i5 3320M processor clocked at $2.6\,\mathrm{GHz}$ with 8 GB RAM. The same computational environment is also used for the ORB-SLAM2-based localization.

TABLE I
COMPUTATIONAL REQUIREMENTS OF EXPLORATION PROCESSES

Module / Process	CPU usage Intel i5*	CPU usage Odroid-XU4**	Update rate
Localization by ORB-SLAM2 [16]	208%	-	11 Hz
RealSense D435 ROS driver	36%	152%	-
RealSense T265 ROS driver	16%	48%	200 Hz
Elevation map building (90k valid points)	18%	40%	5 Hz
Goal determination and path planning	17%	37%	8 s
Path Following	2%	2%	10 Hz
Locomotion control	70%	78%	50 Hz

*Intel i5 has maximal CPU usage 400% (dual-core with Hyper-threading).
**Odroid-XU4 has maximal CPU usage 800% (two quad-cores).

The elevation map is represented by a quadtree structure with the size of the smallest squared cell 7.5 cm. The same cell size is used for the cost map and path planning. The threshold g_{max} is set to $g_{max} = 8\,\text{cm}$ but $g_{max} = 12\,\text{cm}$ at the grass surface. The mapping is performed at 5 Hz from point clouds provided by D435 synchronized with the localization from T265. The navigational goal is refreshed if the path becomes unfeasible or after $T_{exp} = 8\,\text{s}$, path following is run at the frequency 10 Hz. The underlying locomotion control employed tripod gait with a single gait cycle period of 5 s [34]. Computational requirements of the particular modules of the exploration framework depend on map properties and the number of the frontiers, but an average processor utilization with the corresponding update rate per particular exploration process is reported in Table I to provide an overview of the computational requirements. The localization based on the ORB-SLAM2 is not needed when T265 is used, and all ORB-SLAM2 requirements, including memory requirements, which may exceed 4 GB observed during the second experiment summarized in Table II, are saved. Hence, the proposed exploration framework with the T265-based localization can be run on the small embedded computer Odroid-XU4 with Octa-core CPU Samsung Exynos 5422, and 2 GB memory; and the computational requirements are reported in Table I.

A. Performance Comparison of T265 and ORB-SLAM2

Before the deployment of T265 as the single localization module in the autonomous exploration, we examine localization precision in comparison with the state-of-the-art localization method ORB-SLAM2 combined with an RGB-D camera, that has been already deployed in localization of the utilized hexapod walking robot [27]. The precision is measured using the standard metrics of Absolute Trajectory Error (ATE) and Relative Pose Error (RPE) [35]. Since both metrics require to compare the estimated trajectory with the reference ground truth trajectory, we employed motion tracking system AprilTag [36] to detect the pattern attached to the robot by a top placed camera. The robot is equipped with the localization sensor T265 and RGB-D camera D435 utilized by the ORB-SLAM2. The used robot is shown in Fig. 3.

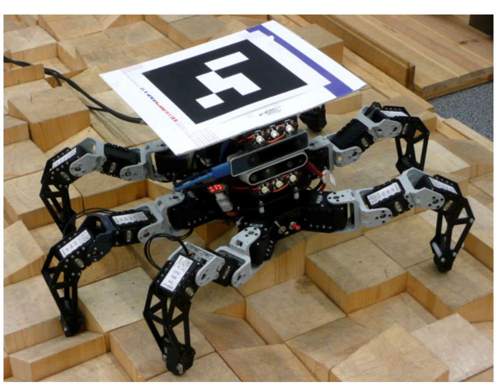


Fig. 3. The used hexapod walking robot with the attached T265 and D435 sensors and AprilTag for reference localization.

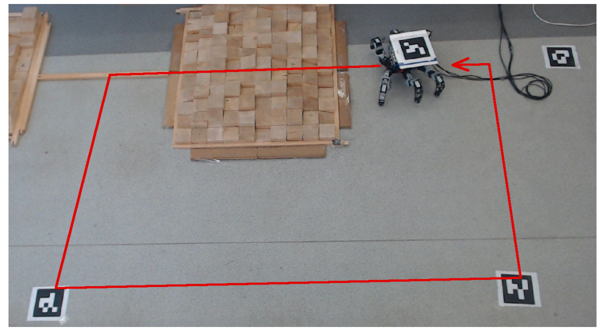


Fig. 4. Setup for the evaluation of RGB-D-based ORB-SLAM2 and T265.

The precision of the localization systems has been evaluated for an experimental deployment in the laboratory environment shown in Fig. 4. Since we are interested in the deployment of the robot in rough terrains, a part of the experimental trail is over irregularly shaped wooden blocks with different heights. The robot has been teleoperated and walked the closed trails five times (trial 1), and ten times (trial 2). Because the trails are closed and contain multiple loops, both localization systems can could take advantage of the loop closure and relocalization. In the total, the robot traversed approximately 130 m during both trails.

The captured images from the top camera (with the resolution 1920×1080 at 20 Hz) have been processed frame-by-frame to provide the reference trajectory. The RGB and depth images from D435 have been used to obtain the trajectory estimate by the ORB-SLAM2, while T265 directly provide the trajectory estimate. The achieved precision of the localization systems by means of the average values of $\overline{\text{ATE}}$ and $\overline{\text{RPE}}$ [35] are summarized in Table II.

TABLE II
PRECISION OF THE EVALUATED LOCALIZATION SYSTEMS

	Trial 1		Trial 2	
Localization system	$\overline{\text{ATE}}_t$ [cm]	$\overline{\text{RPE}}_t$ [cm]	$\overline{\text{ATE}}_t$ [cm]	$\overline{\text{RPE}}_t$ [cm]
ORB-SLAM2	10.91	1.49	10.15	1.32
Intel RealSense T265	10.27	0.55	8.31	0.45

The results summarized in Table II indicate that T265 provides localization with similar absolute error as the ORB-SLAM2 combined with RGB-D camera D435. The relative error of the localization provided by T265 is lower, which is most probably induced by sensory fusion with IMU utilized by T265. Thus, it is a premise to be employed in the exploration, which is reported in the following section.

B. Report on Usability of T265 in Autonomous Exploration

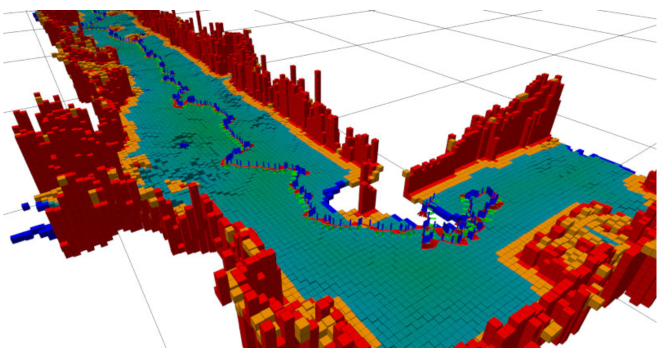
Three scenarios with an *indoor* corridor, outdoor *concrete* panels, and outdoor *grass* terrain have been considered for the examination of the usability of T265 as the sole localization sources in autonomous exploration with small hexapod walking robot. The map is built using depth images of D435, and for all scenarios, the size of the squared grid cell is 7.5 cm. For the indoor scenario, the range of the depth data has been limited to 2.5 m, to increase the frequency of changes of gaze direction, see Fig. 5d, and thus put stronger stress on the performance of the localization system.

In all scenarios, the robot walked roughly about $50\,\mathrm{m}$, but the exploration time is noticeably higher for the outdoor scenarios due to terrain difficulty. A summary of the exploration setup and performance indicators is listed in Table III. The created maps of the explored environments, together with the snapshots of the real environments, are depicted in Fig. 5, Fig. 6, and Fig. 7.

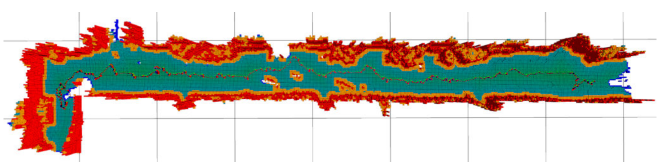
TABLE III
EXPLORATION SETUP, TIME, AND TRAVERSED DISTANCE

Terrain type	Indoor	Concrete	Grass
Map resolution [cm]	7.5	7.5	7.5
Max sensor range [m]	2.5	3.0	3.0
Exploration time [min]	28.3	40.2	44.3
Traversed distance [m]	49.1	55.5	46.0

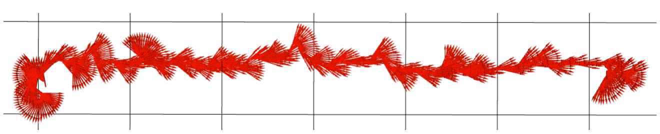
(a) Indoor corridor



(b) Elevation map build during the exploration



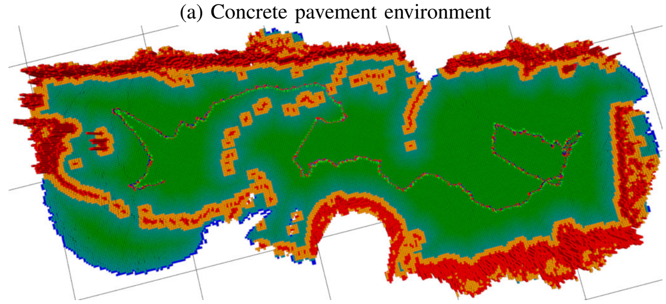
(c) Corridor top view



(d) Robot gaze direction

Fig. 5. Indoor scenario, the size of the shown gray grid is $5\,\mathrm{m}$.

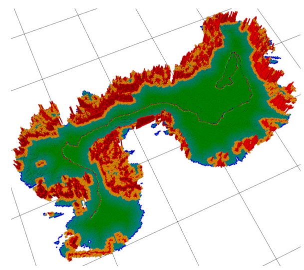
(a) Concrete pavement environment



(b) Created map

Fig. 6. Concrete pavement terrain, the size of the shown gray grid is $5\,\mathrm{m}$.

(a) Uneven grass environment



(b) Created map

Fig. 7. Grass scenario, the size of the shown gray grid is $5\,\mathrm{m}$.

Discussion

Based on the reported results, the localization solely based on T265 seems to be sufficient in the particular deployment scenarios. The comparison results to ORB-SLAM2 indicate competitive precision, while the computational and also power requirements of T265 are significantly lower. Even though we do not report qualitative evaluation of the exploration deployments, regarding the created maps, it seems T265 provides sufficiently precise pose estimation that can be directly used for map building from the RGB-D data provided by D435. The robot has been able to autonomously navigate through the environment, perform the online-decision making, and build the map. Notice that when we deploy ORB-SLAM2 in similar scenarios, the localization sooner or later failed and the autonomous exploration had to be restarted. Such an issue has not been observed with T265. Moreover, the advantage of T265 (as a result of the lower computational requirements) is also in the increased operational time of the robot for the same battery.

V. CONCLUSION

In this paper, we report on the experimental deployment of the embedded visual localization system Intel RealSense

T265 as the sole source for the robot pose estimation in autonomous exploration with small hexapod walking robot. The herein reported results indicate that T265 provides competitive results to ORB-SLAM2, but with the decreased computational requirements. Moreover, it provides sufficiently reliable localization in the considered experimental deployments, where the ORB-SLAM2 fails. Therefore, it is a suitable choice to enable localization and autonomous missions with a small multi-legged robot. In our future work, we aim to examine the properties of the system further, quantify the achieved performance in exploration missions, but also deploy the developed system in environments with low illumination (such as tunnels and mines), where it is needed to carry an external light source, which usage is enabled by the reduced computationally, and thus power requirements.

References

[1] A. Quattrini Li, R. Cipolleschi, M. Giusto, and F. Amigoni, "A semantically-informed multirobot system for exploration of relevant areas in search and rescue settings," *Autonomous Robots*, vol. 40, no. 4, pp. 581–597, 2015.

[2] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 1997, pp. 146–151.

[3] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[4] H. Carrillo, P. Dames, V. Kumar, and J. A. Castellanos, "Autonomous robotic exploration using occupancy grid maps and graph slam based on shannon and rényi entropy," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 487–494.

[5] D. Belter, P. Labecki, and P. Skrzypczynski, "An exploration-based approach to terrain traversability assessment for a walking robot," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2013, pp. 1–6.

[6] Y. Liu and G. Nejat, "Robotic Urban Search and Rescue: A Survey from the Control Perspective," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 2, pp. 147–165, 2013.

[7] M. J. Schuster, S. G. Brunner, K. Bussmann, S. Büttner, A. Dömel, M. Hellerer, H. Lehner, P. Lehner, O. Porges, J. Reill, S. Riedel, M. Vayugundla, B. Vodermayer, T. Bodenmüller, C. Brand, W. Friedl, I. Grixa, H. Hirschmüller, M. Kaßecker, Z.-C. Márton, C. Nissler, F. Ruess, M. Suppa, and A. Wedler, "Towards Autonomous Planetary Exploration: The Lightweight Rover Unit (LRU), its Success in the SpaceBotCamp Challenge, and Beyond," *Journal of Intelligent & Robotic Systems*, vol. 93, no. 3–4, pp. 461–494, 2019.

[8] J. Oberländer, S. Klemm, G. Heppner, A. Roennau, and R. Dillmann, "A multi-resolution 3-D environment model for autonomous planetary exploration," in *IEEE International Conference on Automation Science and Engineering (CASE)*, 2014, pp. 229–235.

[9] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser, and M. Hutter, "Advances in real-world applications for legged robots," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1311–1326, 2018.

[10] P. Fankhauser, "Perceptive Locomotion for Legged Robots in Rough Terrain," Ph.D. dissertation, ETH Zurich, 2018.

[11] M. Reibert, K. Blankespoor, G. Nelson, R. Playter, and the BigDog Team, "Bigdog, the rough-terrain quadruped robot," in *International Federation of Automation Control (IFAC)*, 2008.

[12] A. Roennau, G. Heppner, M. Nowicki, and R. Dillmann, "LAURON V: A versatile six-legged walking robot with advanced maneuverability," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2014, pp. 82–87.

[13] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, R. Diethelm, S. Bachmann, A. Melzer, and M. Hoepflinger, "ANYmal - a highly mobile and dynamic quadrupedal robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 38–44.

[14] K. Walas and D. Belter, "Messor-Verstatile walking robot for serach and rescue missions," *Journal of Automation Mobile Robotics and Intelligent Systems*, vol. 5, pp. 28–34, 2011.

[15] O. A. Silva, P. Sigel, W. Eaton, C. Osorio, E. Valdivia, N. Frois, and F. Vera, "CRABOT: A six-legged platform for environmental exploration and object manipulation," in *4th Congress on Robotics and Neuroscience (CRONE)*, 2019.

[16] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[17] F. Amigoni, J. Banfi, and N. Basilico, "Multirobot Exploration of Communication-Restricted Environments: A Survey," *IEEE Intelligent Systems*, vol. 32, no. 6, pp. 48–57, 2017.

[18] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2002, pp. 540–545.

[19] M. Prágr, P. Čížek, J. Bayer, and J. Faigl, "Online incremental learning of the terrain traversal cost in autonomous exploration," in *Robotics: Science and Systems (RSS)*, 2019.

[20] J. Faigl and M. Kulich, "On benchmarking of frontier-based multi-robot exploration strategies," in *European Conference on Mobile Robots (ECMR)*, 2015, pp. 1–8.

[21] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millennium*, G. Lakemeyer and B. Nebel, Eds., 2003, pp. 1–35.

[22] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[23] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.

[24] A. Souza and L. M. G. Gonçalves, "Occupancy-elevation grid: an alternative approach for robotic mapping and navigation," *Robotica*, vol. 34, no. 11, pp. 2592—2609, 2016.

[25] M. Schwarz, M. Beul, D. Droeschel, S. Schüller, A. S. Periyasamy, C. Lenz, M. Schreiber, and S. Behnke, "Supervised Autonomy for Exploration and Mobile Manipulation in Rough Terrain with a Centaur-Like Robot," *Frontiers in Robotics and AI*, vol. 3, 2016.

[26] M. Menze and A. Geiger, "Object Scene Flow for Autonomous Vehicles," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3061–3070.

[27] M. Nowicki, D. Belter, A. Kostusiak, P. Čížek, J. Faigl, and P. Skrzypczynski, "An Experimental Study on Feature-based SLAM for Multi-legged Robots with RGB-D sensors," *Industrial Robot: An International Journal*, vol. 44, no. 4, pp. 320–328, 2017.

[28] J. Bayer and J. Faigl, "Localization fusion for aerial vehicles in partially gnss denied environments," in *Modelling and Simulation for Autonomous Systems (MESAS)*, 2019, pp. 251–262.

[29] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[30] P. Pfaff, R. Triebel, and W. Burgard, "An efficient extension to elevation maps for outdoor terrain mapping and loop closing," *The International Journal of Robotics Research*, vol. 26, pp. 217–230, 02 2007.

[31] P. Fankhauser, M. Bloesch, and M. Hutter, "Probabilistic terrain mapping for mobile robots with uncertain localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3019–3026, 2018.

[32] P. F. Felzenszwalb and D. P. Huttenlocher, "Distance transforms of sampled functions," *Theory of Computing*, vol. 8, pp. 415–428, 2012.

[33] M. Kulich, J. Faigl, and L. Přeučil, "On distance utility in the exploration task," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4455–4460.

[34] J. Faigl and P. Čížek, "Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only," *Robotics and Autonomous Systems*, vol. 116, pp. 136–147, 2019.

[35] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 573–580.

[36] E. Olson, "AprilTag: A Robust and Flexible Visual Fiducial System," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3400–3407.