

Sobi: An Interactive Social Service Robot for Long-Term Autonomy in Open Environments

Marvin Stuede, Konrad Westermann, Moritz Schappler and Svenja Spindeldreier¹

Abstract—Long-term autonomy in service robotics is a current research topic, especially for dynamic, large-scale environments that change over time. We present Sobi, a mobile service robot developed as an interactive guide for open environments, such as public places with indoor and outdoor areas. The robot will serve as a platform for environmental modeling and human-robot interaction. Its main hardware and software components, which we freely license as a documented open source project, are presented. Another key focus is Sobi's monitoring system for long-term autonomy, which restores system components in a targeted manner in order to extend the total system lifetime without unplanned intervention. We demonstrate first results of the long-term autonomous capabilities in a 16-day indoor deployment, in which the robot patrols a total of 66.6 km with an average of 5.5 hours of travel time per weekday, charging autonomously in between. In a user study with 12 participants, we evaluate the appearance and usability of the user interface, which allows users to interactively query information about the environment and directions.

I. INTRODUCTION

Social service robots can now be found performing tasks such as providing information, guiding, transport or entertainment in various contexts, e.g. retail, airports [1] or hotels [2]. Despite their commercial availability, long term autonomy (LTA) without human intervention, especially in unstructured and dynamic environments, remains a current research challenge. Long-term autonomous operations thereby pose special demands to robot design, both in terms of robust and reliant software and hardware.

In this paper we present Sobi, a social service robot for information provision and guiding in open environments (see Fig. 1). Sobi is designed for use on a campus inside and outside buildings and answers voice- and touch-based requests for e.g. directions, room plans, canteen menus or small talk and provides a guiding functionality.

The robotic tour guide and information terminal scenario is a popular use case and has been considered in various contexts in research. Early developments focused mostly on robust localization and navigation, such as the RHINO robot which was deployed for 6 days in a museum [3]. Later implementations reached cumulative operating times of several weeks. Sacarino is a service robot that was deployed for multiple weeks to provide information and do navigation tasks for guests in a hotel [4]. In the CoBot project, cumulative travelled distances of more than 1000 km were achieved with multiple robots [5]. The project fo-



Fig. 1: Sobi interacting with users during one of the presentations at public events (Hannover city hall).

cused on long-term mapping, navigation and human-robot interaction (HRI), especially to proactively ask humans for help in problematic situations (e.g. operating a lift without physical manipulators). Between autonomous operations, however, manual intervention by supervisors is necessary, for example for the charging process. In [6], it is described that autonomous charging capabilities as well as deliberate requests for supervisor assistance can significantly lengthen the deployment time. The EU project STRANDS focused on the development of specific methods for environmental modeling and HRI for LTA operations. Their robots reached multiple weeks of uninterrupted autonomy in security and care contexts [7] and tour-guide scenarios [8] in subsequent works. Other projects focus on types of interaction e.g. by voice and facial recognition [9] [10], specific environments (e.g. outdoors [11] or elder care [12]), or perception and social navigation [1].

However, research is often based on the use of commercially available robots, usually supplemented by only a few hardware components, or the systems are designed for either exclusive indoor [4]–[8], [10], [12] or outdoor use [11], and the used sensors and actuators would make them unsuitable for the respective other area. In contrast, Sobi is aimed at long term autonomous use in unsupervised open spaces, multiple days and weeks indoors and frequent partially supervised deployments outdoors. Contrary to other LTA systems, we provide not only the complete robot's software as open source, but also make the system available as an open source hardware project, including 3D-model and design data, mechanical and electrical drawings and parts lists¹.

One key aspect of LTA is monitoring system variables and executing defined recoveries in the event of a failure, which

¹All authors are with the Leibniz University Hannover, Institute of Mechatronic Systems, D-30823 Garbsen, Germany, marvin.stuede@imes.uni-hannover.de
978-1-6654-1213-1/21/\$31.00 ©2021 IEEE

¹<https://marvinstuede.github.io/Sobi>

has been explored particularly for areas outside of service robotics, such as underwater or extraterrestrial applications. This can involve model-based methods that compare the system and component behavior with a nominal target [13] or data-driven approaches for outlier rejection [14]. Although various descriptions for LTA applications in service robotics note that monitoring is an important aspect [6], [7], [10], the specific methods used are not presented.

Therefore, this paper firstly introduces the main design and components of Sobi, with an emphasis on LTA aspects such as sensor and algorithmic setup for robust localization indoors and outdoors. Furthermore, a novel type of stateless reactive monitoring system based on Behavior Trees (BTs) as part of Sobi is presented, which detects faults and reacts accordingly by restarts or recoveries. In the course of the evaluation, we show first results on LTA with a 16-day deployment in which the robot patrols a total of 66.6 km within a building and notifies people of compliance to hygiene regulations. The external appearance as well as the human-robot interface are examined in a user study, that includes verbal and touch requests for places of interest, public transport and meal plans. To summarize, the contributions of this paper are 1) a novel social service robot under free license for the use in indoor and outdoor environments, 2) a stateless reactive monitoring system based on BTs for LTA of the robot, 3) experimental evaluation of the robot in a scenario aiming at LTA and in a user study.

The remainder of this paper is structured as follows: The next section II gives an overview of the main hardware and software components of Sobi. The monitoring system is described in section III. Subsequently, section IV describes the long-term test and user study. In section V we give some remarks about the lessons learned throughout the development process. Finally, in section VI we provide a summary and an outlook on further work.

II. ROBOT DESIGN AND COMPONENTS

This section describes the design, hardware and software components of Sobi, starting with a brief overview of the core requirements:

The robot is supposed to operate in various buildings as well as in the outdoor area of the newly built campus of the Faculty of Mechanical Engineering of the Leibniz University Hannover. Within the robot's home building, fully autonomous operation should be possible by specifically using a charging station when needed. In other areas, the robot should operate autonomously as long as the batteries are charged and will then be manually transported back to the home building. It should be able to operate for several hours with robust localization and navigation in these dynamic environments, and also enable multi modal perceptual sensing for applications such as person recognition and tracking. Since the robot should look appealing, one of the development priorities is an approachable outer design, which at the same time protects the robot's components from environmental influences, and an intuitive user interface. The robot should therefore have basic splash water protection

to withstand spills or brief drizzles (according to the IP21 protection class to protect against touch and vertically falling water). However, it should be noted that we used the IP rating only as a guideline and no industrial grade tests were performed.

A. Exterior Shape and Design Concept

The shape and color concept of the robot was developed in close cooperation with the Hanover University of Applied Sciences and Arts. Sobi's design is based on a futuristic appearance and clear lines, which is intended to have a friendly and inviting effect on users. The shape is based on simple geometric bodies offset with chamfers, which can be found throughout the design of the robot. The bevelled design and the use of straight lines are furthermore intended to avoid water build-up and to ease sealing of transition points. One individual design feature is the circumferential aluminum-bracket that connects the upper body to the base and was CNC-machined. Since the robot is relatively large with a height of 1.56 m and width of 0.66 m, this feature is supposed to give an impression of lightness. Humanoid features are the torso and head with eyes, which are indicated as rings, as well as movable arms and ears, which can be illuminated in color by LEDs. Flat elements (e.g. for arms and ears) were laser-cut from aluminum and ABS plastic. The outer white covers of the robot, with the exception of the base cover, are laser sintered and coated to repel water.

B. Hardware Components

The described externally visible elements of the robot are supported by a structure made of aluminum square-profiles, which is rigidly connected to a **wheeled platform**. All components mentioned in this section are shown in Fig. 2. The platform (Neobotix MP-500: differential drive with one caster wheel) has an onboard computer and **2D-Lidar**, which can trigger low-level emergency stops for collision avoidance. It includes two 12 V AGM batteries in series with a total capacity of 50 Ah, which we extended by two smaller batteries for a total capacity of 75 Ah. This gives a minimum time of 4–5 hours until a recharge is needed for heavy usage and over twelve hours standby time.

The embedded **main computer** used for controlling the robot is mounted on the platform (Vecow EVS-1010: Intel i7-7700T, 16 GB RAM, GeForce GTX1050 GPU). It has two built-in WiFi modules to be permanently connected to the internet and to provide a WiFi hotspot for external access in field operation. The robot includes the following sensors for localization, navigation and interaction:

- Xsens MTi-30 **IMU/AHRS** for odometry calculation
- Bosch Parkpilot URF7 **sonar sensors** for collision avoidance (3×)
- SEED ReSpeaker v2.0 **microphone array** for speech recognition
- Velodyne Puck **3D-Lidar** for localization, people perception, and collision avoidance
- Intel D435 **RGBD cameras** (2×, mounted frontal and dorsal) for localization and people/object perception.

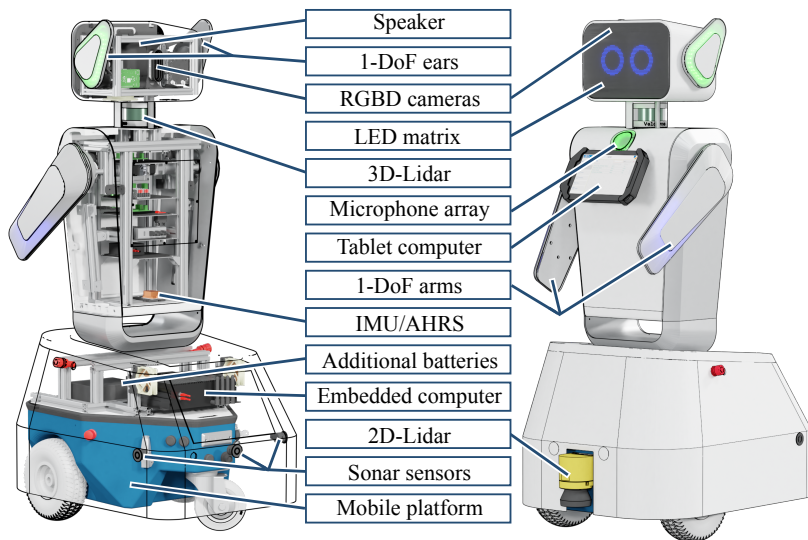


Fig. 2: Hardware components and their placement in Sobi.

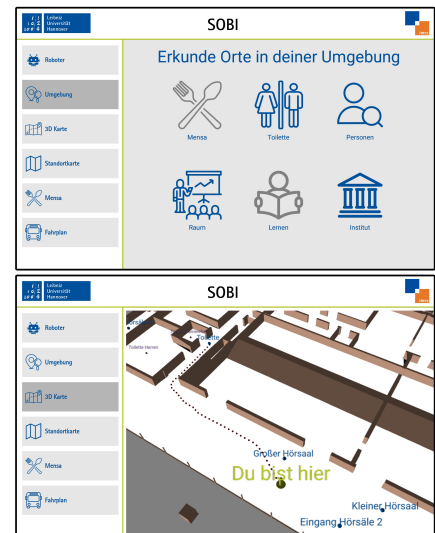


Fig. 3: GUI of the tablet. Places of interest (POI) can be chosen by category (top). A 3D map then shows the path from the current position (bottom).

The cameras are connected to a Nvidia Jetson Nano inside the head, which provides the synchronized image data via Ethernet and thus reduces the load on the main computer.

Apart from the platform, the only actuators are BLDC motors to move the arms and servo motors for the ears. These are not used for physical interaction or manipulation, but for social interaction to make the appearance of the robot more natural by supporting the emotions to be displayed (e.g. wiggling ears to depict happiness or lowering ears for sadness). Besides a speaker and tablet (Fig. 3), another main component for interaction is the 64×32 px RGB **LED matrix** that displays still images or animations. The speaker, ears and LED panel are controlled by a Raspberry Pi, which communicates with the other computers via Ethernet.

C. Software Architecture

All four computers in Sobi run Linux (Ubuntu) and the Robot Operating System (ROS) as a framework for communication and control. The software is partially based on freely available ROS packages, some of which have been adapted for the specific robot setup, but mainly consists of self-developed solutions. The layered architecture is shown in Fig. 4. All relevant ROS messages are continuously stored to an external server running a MongoDB instance. The central control unit in the application layer is a Behavior Tree (BT), partly composed of sub-trees of the behavior layer, which can also be executed independently. Since the robot must act purely reactively on immediate requests, modelling via BTs is well suited and the use of a task scheduler is not necessary. However, recent advances in BT research would also enable more sophisticated use cases which incorporate planning capabilities for BT synthesis [15], [16].

In the following sections, the individual software functions of Sobi are presented in more detail.

1) People and Object Perception

People perception and tracking is conducted through the 3D Lidar and RGBD cameras. YOLO v3 [17] is used for people and object detection in the RGB images. The centroids of resulting bounding boxes are registered with a median filter based on the distance either with the 3D Lidar data or the depth data of the cameras. We use a modified version of [18] for 3D point cloud segmentation, extended by the approach proposed in [19] for 2D clustering to decrease the false negative rate. The aggregation and tracking is performed with the SPENCER framework [20], which enables tracking in a circular area with a radius of up to 5 m through the multimodal combination of different sensors.

2) Localization and Navigation

A basic requirement for a mobile robot is to know its own position in relation to the environment. Especially in dynamically changing environments this poses a challenge. As Sobi is designed to be used in different areas, both inside and outside of buildings, a powerful SLAM solution is one of the most important requirements. RTAB-Map (real-time appearance based mapping) [21] is used as the basic SLAM method, as it allows flexible use of different sensor types (e.g. RGBD cameras, Lidar). RTAB-Map comprises a powerful framework that can be used for long-term, large-scale and multi-session mapping. As input for the SLAM front-end we use the wheel odometry, fused with the measurement data of the IMU via an extended Kalman filter, as odometry source and both cameras and 3D Lidar as sensors. Since the different types of environment place different demands on the SLAM system, we developed a method that automatically selects predefined SLAM configurations for different environments depending on various criteria, such as distances to walls. For example, for outdoor or large-scale environments the depth information of the 3D Lidar is used instead of the cameras,

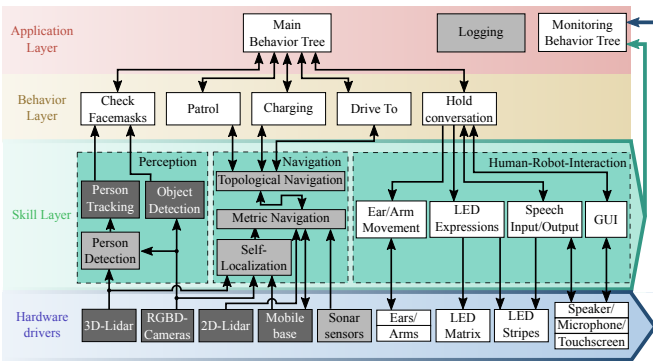


Fig. 4: Layered software structure of the robotic system. White boxes indicate custom self-developed programs, dark grey boxes third-party ROS programs and light grey boxes modified or extended third-party ROS programs.

and for indoor environments the maximum range for map generation is reduced [22].

One of the biggest challenges of visual SLAM methods like RTAB-Map are changes in illumination or appearance over time. To increase the number of loop closures, we added to RTAB-Map the possibility to use 3D point clouds for loop closure detection by a trained classifier of global point cloud descriptors and then register them in a multi-step process. Especially in outdoor environments and environments poor in visual features, localization could thus be significantly improved. Further information and experimental results can be found in the corresponding paper [23].

For navigation on metric maps, respectively local and global path planning, standard techniques of the ROS framework (i.e. *MoveBase*) are used. The 3D map is projected onto the ground plane to create an occupancy grid map that can be used with standard planners. Above this level, we use navigation on a topological map, where nodes represent either relevant locations, such as specific rooms or facilities, or waypoints between which the robot can either navigate directly or move using problem-specific planners. An example for problem-specific navigation planning is docking and undocking at the charging station. The former uses a triangular landmark that can be detected by the 2D laser scanner. It plans a Dubins path consisting of circular- and linear segments and follows the path with a pure pursuit controller [24]. These approaches are not complex in their computation and have resulted in robust docking processes in initial evaluations.

3) Human-Robot Interaction

The main task of Sobi is to provide campus-specific information as well as guiding applications. For intuitive accessibility, the human-robot interface therefore consists of speech processing as well as touch operation. Speech processing is based on a combination of Google's Speech-to-Text and Text-to-Speech services and the Natural Language Processing pipeline Dialogflow (see further information in [25]). Since WiFi coverage is available throughout the robot's area of operation, the latencies of the speech processing pipeline are short enough to allow smooth interaction. In

addition, no offline backup solution is required, because the internet connection was available throughout the entire development and test period. All information can be accessed via both the speech and touch interfaces. The interface includes the following functions: display and queries of the canteen menu, public transport timetables, staff offices and room locations as well as options for small talk. By connecting the topological map of the robot with the environmental structure, it is also possible to locate nearby places (e.g. restrooms, seminar rooms or offices). The path to the requested location can then be displayed on a 3D map for destinations on the same floor (Fig. 3). A daytime-specific greeting and thus the start of an interaction occurs when a person is detected in front of the robot or the touch display is activated. To further enhance HRI, animations on the LED panel, varying colors of the LED strips, and movements of the ears are also used. Blinking eyes indicated as rings are displayed in normal operation, as well as various animations for specific situations (e.g. *laughing*, *sad* or *sleeping* during the charging process). When the voice input is activated by a button on the tablet, the robot's LEDs turn green, the ears move forward, and the indicated eyes widen. Unanswerable requests, on the other hand, are underlined by lowered ears and the depiction of a sad face.

III. MONITORING FOR LONG-TERM AUTONOMY

Monitoring system variables of hardware and software applications is an essential building block for achieving long-term autonomy. We therefore developed a framework for Sobi, which monitors various system variables and reacts to faults. The monitoring system consists of separate applications that monitor hardware- and ROS framework parameters. The hardware monitors include CPU, RAM and network load monitoring as well as measurement of time differences of host computers based on Network Time Protocol (NTP). ROS nodes are monitored by continuously checking if they are pingable and that essential topic publishing rates are within a tolerance band. Furthermore, we implemented monitors that continuously check whether there is a valid loop closure in the localization system and if there are error cases in the navigation system. Each monitor includes an individual warning and error range and respective messages are aggregated on one single ROS topic. The aggregated values are then used as input to an arbiter based on the Behavior Tree framework, that deterministically reacts to the different error cases. Although Behavior Trees are mainly used for sequential control of autonomous agents, their advantages in terms of reactivity and modularity combined with intuitive modelability are also applicable for system monitoring. Due to the statelessness of BT and their reactive structure, it is thus possible to react immediately to errors that occur, without the need for explicit state transition modeling as in the case of finite state machines, for example. The structure of the monitoring system is shown in Fig. 5.

The system entities (i.e. programs/topics) to monitor are organized in configurations, that can be switched autonomously or manually and are created for different use

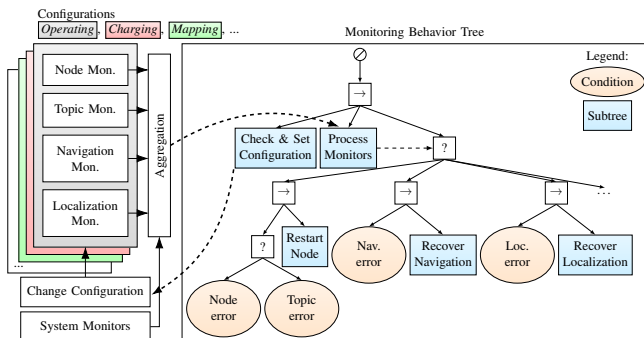


Fig. 5: Structure of the monitoring framework. Configuration dependent monitored values are processed in a BT. The composition of the tree defines the priority of error handling.

TABLE I: Overview of the utilized monitors with detection signal and corresponding reaction.

Name	Detection	Reaction
Node monitor	Node not pingable	Restart node
Topic monitor	Publish frequency not in tolerance band	Restart publishing node
CPU, RAM, NTP, Network Monitor	Not in tolerance band	Send message
Navigation monitor	No global path found and <i>MoveBase</i> recoveries failed	1) Wait and retry 2) Move backwards 3) Ask Supervisor
Localization monitor	No loop closure detected	1) Slow rotate 2) Restart localization 3) Slow rotate 4) Ask Supervisor

cases, e.g. for normal operation, charging or mapping. All unneeded programs from other configurations are terminated when a configuration change is made. Whether an error is present is determined on a monitor-specific basis via detection signals. Table I summarizes these signals together with the associated recovery reaction.

Similar to [6], we follow the approach that maximum robustness may not be achieved by full autonomy alone, but by planned interventions of supervisors in case of failure. Therefore, the last resort for navigation and localization errors is a predefined request to a list of supervisors, in which the robot sends an instant message with an URL. On the linked website, a teleoperation can then be performed based on the camera views, or the current position on the map can be specified. This simple system intervention usually takes less than a minute of the operator's time, but prevents a total failure of the system. Once the problem has been fixed, this is confirmed by the operator and an all-clear is sent to the other supervisors.

IV. EVALUATION

The evaluation of the robot consists of two parts: In the first part, we test the LTA capabilities in a continuous 16 day deployment and in the second part, we conduct a user study on human-robot interaction to evaluate the user interface and robot's appearance.

A. Long-Term Autonomy

The goal of this evaluation is to determine the robustness of the localization, navigation and monitoring system as

well as the individually created algorithms, e.g. for docking, and to identify possible weaknesses. The robot's task is to permanently patrol within one floor during office hours (9 am–5 pm) on working days. Outside these hours and if necessary in between, Sobi autonomously approaches the charging station and performs the charging process. The environment mainly consists of a hallway with several offices and an entrance area. The metric and topological maps of the environment are shown in Fig. 6. Due to hygiene restrictions in place at the time of the evaluation, passers-by are not allowed to access the robot by touch and are also required to wear a face mask while in the building. The robot is therefore tasked with detecting whether people are wearing a face mask, by analyzing the image of the frontal camera every 0.5 s and using face detection. The underlying YOLO network was trained with 600 images containing both *mask* and *no mask* classes. A test set of 100 images results in an average precision of 92.9% for the *mask* and 82.34% for the *no mask* class. If the *no mask* class is detected, the robot verbally asks the person to put on a mask. Given the total number of detections in the evaluation period, the average precision can give a rough estimate about how many people were encountered directly in front of the robot. This therefore also serves as a measure of the environment dynamics, since crowded environments pose a greater challenge for long-term autonomous systems. Together with typical metrics for long-term autonomy the number of detections is summarized in Table II. In total, Sobi was undocked from the charging station for 71.2 h, of which 65.4 h were spent in motion, so that a distance of 66.6 km was covered during the evaluation period. The time in motion indicates the time during which the robot was able to perform the patrolling task and was not in an error or recovery state. With respect to an 8-hour workday of the robot, this results in an autonomy percentage ($A\%$) of 69%, which is an indicator of the percentage of the available time that the robot actually uses to perform its services (patrolling). In our case, this indicative parameter is mainly influenced by the average additional charging time of 2 h during the day.

An overview of the recovery behaviors performed is shown in Table III. As in [7], a recovery behavior is considered successful if no further recovery behavior needs to be performed within one minute. For the recovery behaviors for navigation, a further requirement is that no additional recovery behavior must be performed within a circle with a radius of 1 m within this time. From the table it can be seen that many problems can already be solved by waiting and going back (in case of navigation errors) or slow rotation and restarts (in case of localization errors). Requested remote access was necessary in a total of nine cases. These occurred more frequently when there were major changes to the environment and, for example, many doors were open that had been closed during the mapping process, or when closed fire doors or obstacles blocked the way.

A typical quantity for evaluating robustness is the total system lifetime (TSL), which specifies the time interval between interventions by supervisors in the event of a failure,

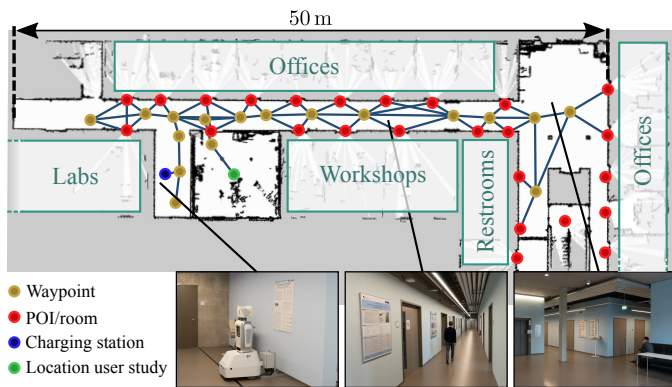


Fig. 6: Environment used for validation. Dots indicate nodes and blue lines indicate edges of the topological map. The robot continuously patrols between the shown waypoints in a random fashion.

which were not specifically requested by the robot. During the evaluation period, a total of four of such interventions were necessary, resulting in the TSL values shown. Two of the interventions were due to software errors that resulted in the signal to approach the charging station not being sent, requiring manual intervention to avoid shutdown. On one other occasion, various programs were permanently restarted within a deadlock state, making it impossible to continue the process. This deadlock situation mainly influences the 62.5% success rate of the node restart recovery, because ten node restarts occurred in a timespan of less than three minutes. The fourth intervention was necessary when a docking attempt failed, so that the robot performed an emergency shutdown and had to be switched on again manually. A total of 28 docking attempts were made during the evaluation period, with all other 27 attempts carried out successfully. Except for this last error, all problems could be recovered via remote SSH access to the system and no hardware errors occurred. One drawback of the TSL, and also the $A\%$ is, that it does not validate how *useful* the services of the robot actually are for users. In the next section, we therefore present a separate study on how users perceive the robot and its info-terminal services.

B. User Study on Human-Robot Interaction

The user study is based on the Godspeed Questionnaire Series (GQS) [26] and is conducted with 12 participants (10 male, 2 female, all between 25 and 34 years old), who had no prior knowledge or direct contact with the robot before the study. The GQS measures a user's perceived sensations during a social interaction with a robot and is one of the most widely used assessment criteria in this area [27]. It describes the perceived impression of the robot via five categories and different bipolar adjective pairs on a scale from 1 to 5 using a semantic differential.

The interaction between Sobi and a user is conducted in the location shown in Fig. 6 and is divided into three independent phases. During the experiment, the robot remained in a static position and interaction was started by the users. The shown POI are visualized on the 3D map of the tablet accordingly. In the first phase, users were given specific tasks

TABLE II: Metrics of the deployment. Weekend days are excluded.

Metric	Value
Timespan	12 days
Mean time operating per day	5.9 h
Mean time moving per day t_m	5.5 h
$A\% = t_m/8\text{ h}$	69 %
Traveled distance	66.6 km
Mean TSL	56.7 h
Max TSL	90.6 h
Detected faces with mask	983
Detected faces w/o mask	101

TABLE III: Executed recovery behaviors sorted by category.

Type	Reaction	#	success
ROS	Restart node	32	62.5 %
Navigation	Wait and retry	194	70.6 %
	Move backwards	41	48.8 %
	Ask supervisor for teleoperation	5	100 %
Localization	Slow rotate	76	84.2 %
	Restart localization & rotate again	12	66.6 %
	Ask supervisor for relocalization	4	100 %

to perform. This included figuring out how to get to the nearest restroom as well as to the office of a given person. Users were not provided with any assistance, nor were they made aware that there was more than one possibility to get the requested information, using speech commands or the tablet GUI. After successfully visualizing the path to the destination on the tablet's 3D map, users should identify information about public transportation departure times. In the second phase of the study, five minutes were provided for free interaction, during which Sobi's abilities were to be learned in more detail by trying them out for themselves. In a concluding observation phase, Sobi gave some verbal closing remarks and insight about further possible emotions and robot states.

The results of the GQS are shown in Fig. 7. For the evaluation of reliability, the respective value of Cronbach's alpha represents a measure of internal consistency within a category. The results show that the robot likeability level felt during the interaction is not dependent on distinctive human-like characteristics of the robot. Even though human-like characteristics within the anthropomorphism category can only be partially applied to Sobi, the interaction was predominantly perceived as pleasant. Sobi's appearance and behavior are positively highlighted with overall high perceived likeability by all users. Sobi is perceived as a mechanical robot, but its dynamic behavior clearly distinguishes him from a static machine. By greeting a user automatically when he or she approaches as well as providing immediate verbal and visual feedback to user requests, the robot is perceived as lively and responsive. With high internal consistency within the corresponding category, the perceived intelligence of the robot was rated relatively positively with medium to high values, but continues to show potential for development. In many cases, speech input was used to search for locations, which were not always correctly comprehended by the robot or were fundamentally unknown to it. However, regardless of any failed attempts, the robot's behavior in responding to user instructions is considered reasonable. The outliers at values of 1 could be due to an inadvertent erroneous rating of the category by one person, since this person otherwise positively highlighted the interaction with Sobi as well as

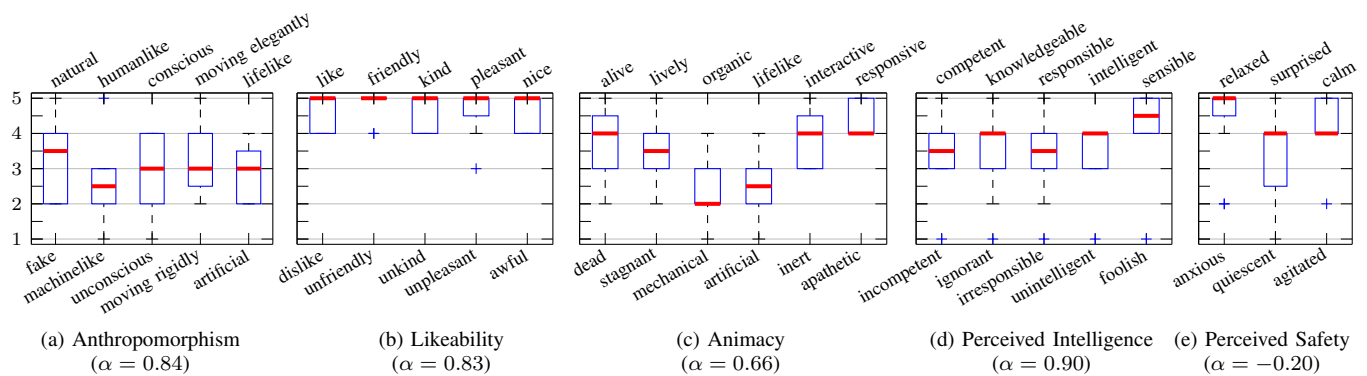


Fig. 7: Results of the user study on sensations and perceptions (Godspeed Questionnaire Series) of a user during interaction with Sobi. The consistency within a category is indicated in each case with Cronbach's α . The values for the anthropomorphism, likeability, and perceived intelligence categories are above the 0.70 threshold, indicating internal consistency. A low value in the safety category was also reported in similar research [28].

his abilities in an additional free text field that had to be answered. Other people described the operation of the robot as smooth and intuitive, but also mentioned possible scopes of knowledge that could be learned by Sobi. Overall, Sobi was perceived as a competent robot, which was able to answer most of the users' concerns. Due to its appearance as well as its active and reactive behavior, the robot was accepted as a pleasant interaction partner.

V. LESSONS LEARNED

During the development period and the evaluation runs we had the opportunity to test Sobi, identify errors and obtain feedback. We will therefore briefly discuss the most important insights and challenges of this process. The most prominent factor in the first months of development of the robot was the choice of main hardware components (i.e. the mobile base, sensors and computers). Since the selection of reliable hardware components is the basis for long-term autonomous use, we mainly use established components with maintained ROS support, full integration and standard transmission interfaces (e.g. USB or Ethernet). The signal lines are kept as short and shielded as possible, since in our experience, especially with data-intensive transmission (e.g. camera data), line losses or interference lead to errors as the operating time increases. To develop the SLAM functionality of the robot, a functional model with a simple aluminum frame structure was used early in the development process. This allowed the exact positioning of sensors to be easily varied. After these initial tests, we started working with the design team early on to conceptualize and develop the eventual design of the robot appearance. By defining this at an early stage, it was thus possible to avoid redesigns of the mechanical structure and provide a guideline for the placement of the other robot components.

The robot initially contained only the main computer and the computer in the mobile platform to run the software. However, it has proven useful to outsource individual functions to function-specific embedded computers. The execution of many hardware drivers can already lead to a considerable system load, which can be counteracted by

distributing them over several computing units. Thus, for example, the camera data is read out and synchronized by the Jetson computer, so that only the compressed RGBD data has to be made available to the ROS system via Ethernet. An essential prerequisite for this is that the system clock times of all computers are synchronized with each other to avoid inconsistencies in ROS message processing. A common approach is to use Network Time Protocol (NTP), with the main computer acting as the NTP server in our case, so time synchronization is possible regardless of an internet connection.

In terms of software development and implementation for long term autonomy, we share the findings from similar efforts in the literature [6], [7], [10]. Especially the design principles that all programs should tolerate the temporary unavailability of other programs and that program restarts should transition to a clean state are the basis for the design of our software and the monitoring system. In terms of localization, our system differs from the aforementioned literature in the use of multiple RGBD cameras in conjunction with a 3D laser scanner. Since the localization programs are terminated during the charging process, it is necessary to quickly regain a valid localization estimate when resuming operation. For this purpose, we hung up a poster with many images and text directly above the charging station, so that a large number of distinctive features leads to a loop closure detection directly after undocking (see Fig. 6). It has also proven useful to cover a large field of view with the different perceptual sensors in order to prevent the robot from losing localization if, for example, people are standing around it. On the other hand, the fields of view of the cameras and the 3D laser scanner overlap, so that the depth information of the scanner can be registered with the RGB images. This is particularly advantageous for large distances, since the depth data from the cameras is only useful for distances of a few meters.

We also had to discard some original plans during the course of the project. The implementation of actuators requires a significant amount of time and therefore we did not implement a rotatable head and a tiltable tablet as

originally intended. Furthermore, the plan to let the robot move autonomously between indoor and outdoor areas is prevented by door sills. These are too high (20–30mm) for the robot to pass, due to its mass of 150 kg with a wheel diameter of 260 mm. This problem arises from an incorrect assumption regarding accessibility, as the campus was still under construction during the project period.

VI. CONCLUSION AND FUTURE WORKS

In this work we presented our approach to building a long-term autonomous robot in terms of design, hardware components and software structure. The robot's monitoring system was presented and tested in a 16-day evaluation with regard to long-term autonomy. In a user study with 12 participants, the functionalities of the HRI were assessed in terms of impression and usefulness. Participants described the services offered by the robot as useful and perceived the appearance as very pleasant. During the long-term evaluation the robot was in motion for 65.4 h, traveling a total distance of 66.6 km. These first results show that the robot is suitable for long-term autonomous tasks, since the main sources of error were due to the developed software. In addition to correcting these errors, the robot will be tested in further outdoor deployments in the future to test the capabilities for long-term autonomy under even more dynamic environmental conditions. Within the deployment on our campus, further insights are to be gained through a larger number of users and recording of time- and location-dependent long-term usage patterns. This information will then be used to create data-driven models of people occurrence and usage patterns, which could then be employed to actively improve the services offered by the robot.

ACKNOWLEDGMENT

This work was funded by the Faculty of Mechanical Engineering of the Leibniz University Hannover. Special thanks go to the product design team of the Hannover University of Applied Sciences and Arts.

REFERENCES

- [1] R. Triebel *et al.*, "SPENCER: A Socially Aware Service Robot for Passenger Guidance and Help in Busy Airports," *Springer Tracts in Advanced Robotics*, pp. 607–622, 2016.
- [2] S. H. Ivanov, C. Webster, and K. Berezina, "Adoption of Robots and Service Automation by Tourism and Hospitality Companies," *Revista Turismo & Desenvolvimento*, vol. 27, no. 28, pp. 1501–1517, 2017.
- [3] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "The Interactive Museum Tour-Guide Robot," in *AAAI Conference on Artificial Intelligence*, 1998, pp. 11–18.
- [4] R. Pinillos, S. Marcos, R. Feliz, E. Zalama, and J. Gómez-García-Bermejo, "Long-term Assessment of a Service Robot in a Hotel Environment," *Robotics and Autonomous Systems*, vol. 79, pp. 40–57, 2016.
- [5] J. Biswas and M. Veloso, "The 1,000-km Challenge: Insights and Quantitative and Qualitative Results," *IEEE Intelligent Systems*, vol. 31, no. 3, pp. 86–96, 2016.
- [6] W. Meeussen, E. Marder-Eppstein, K. Watts, and B. Gerkey, "Long Term Autonomy in Office Environments," in *IEEE International Conference on Robotics and Automation (ICRA), Workshop on Long-term Autonomy*, 2011.
- [7] N. Hawes *et al.*, "The STRANDS Project: Long-Term Autonomy in Everyday Environments," *IEEE Robotics & Automation Magazine*, vol. 24, no. 3, pp. 146–156, 2017.
- [8] F. D. Duchetto, P. Baxter, and M. Hanheide, "Lindsey the Tour Guide Robot - Usage Patterns in a Museum Long-Term Deployment," in *28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, pp. 1–8.
- [9] M. de Jong *et al.*, "Towards a Robust Interactive and Learning Social Robot," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2018, pp. 883–891.
- [10] S. Wang and H. I. Christensen, "TritonBot: First Lessons Learned from Deployment of a Long-Term Autonomy Tour Guide Robot," in *RO-MAN*, 2018, pp. 158–165.
- [11] K.-H. Chiang, S.-H. Tseng, Y.-H. Wu, G.-H. Li, C.-P. Lam, and L.-C. Fu, "Multisensor-based Outdoor Tour Guide Robot NTU-I," *SICE Annual Conference*, pp. 1425–1430, 2008.
- [12] M. Luperto, J. Monroy, J. R. Ruiz-Sarmiento, F.-A. Moreno, N. Basilico, J. Gonzalez-Jimenez, and N. A. Borghese, "Towards Long-Term Deployment of a Mobile Robot for at-Home Ambient Assisted Living of the Elderly," *European Conference on Mobile Robots (ECMR)*, pp. 1–6, 2019.
- [13] G. Antonelli, "Fault detection/tolerance strategies for AUVs and ROVs," in *Underwater Robots*. Springer, 2014, pp. 101–116.
- [14] E. Khalastchi and M. Kalech, "On Fault Detection and Diagnosis in Robotic Systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, p. 9, 2018.
- [15] M. Colledanchise, D. Almeida, and P. Ögren, "Towards Blended Reactive Planning and Acting using behavior trees," in *ICRA*, 2019, pp. 8839–8845.
- [16] F. Rovida, B. Grossmann, and V. Krüger, "Extended Behavior Trees for Quick Definition of Flexible Robotic Tasks," in *IROS*, 2017, pp. 6793–6800.
- [17] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [18] Z. Yan, T. Duckett, and N. Bellotto, "Online Learning for Human Classification in 3D LiDAR-based Tracking," *IROS*, pp. 864–871, 2017.
- [19] I. Bogoslavskyi and C. Stachniss, "Fast range image-based segmentation of sparse 3D laser scans for online operation," in *IROS*, 2016, pp. 163–169.
- [20] T. Linder, S. Breuers, B. Leibe, and K. O. Arras, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *ICRA*, 2016, pp. 5512–5519.
- [21] M. Labbé and F. Michaud, "RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.
- [22] S. F. G. Ehlers, M. Stuede, K. Nuelle, and T. Ortmaier, "Map Management Approach for SLAM in Large-Scale Indoor and Outdoor Areas," *ICRA*, pp. 9652–9658, 2020.
- [23] T.-L. Habich, M. Stuede, M. Labbé, and S. Spindeldreier, "Have I been here before? Learning to Close the Loop with LiDAR Data in Graph-Based SLAM," *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. Accepted for publication. *arXiv preprint arXiv:2103.06713*, 2021.
- [24] R. C. Coulter, "Implementation of the Pure Pursuit Path Tracking Algorithm," Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, Tech. Rep., 1992.
- [25] M. Stuede, J. Wilkening, S. Tappe, and T. Ortmaier, "Voice Recognition and Processing Interface for an Interactive Guide Robot in an University Scenario," *2019 19th International Conference on Control, Automation and Systems (ICCAS)*, pp. 1238–1242, 2019.
- [26] C. Bartneck, D. Kulić, E. Croft, and S. Zoghbi, "Measurement Instruments for the Anthropomorphism, Animacy, Likeability, Perceived Intelligence, and Perceived Safety of Robots," *International Journal of Social Robotics*, vol. 1, no. 1, pp. 71–81, 2009.
- [27] A. Weiss and C. Bartneck, "Meta Analysis of the Usage of the Godspeed Questionnaire Series," in *RO-MAN*, 2015, pp. 381–388.
- [28] Z.-H. Tan *et al.*, "iSocioBot: A Multimodal Interactive Social Robot," *International Journal of Social Robotics*, vol. 10, no. 1, pp. 5–19, 2018.