Applications

Arne Roennau*, Johannes Mangler, Philip Keller, Marvin Grosse Besselmann, Nicolas Huegel and Ruediger Dillmann

Grasping and retrieving unknown hazardous objects with a mobile manipulator

https://doi.org/10.1515/auto-2022-0061 Received April 28, 2022; accepted September 20, 2022

Abstract: Robots show impressive flexibility and reliability in various applications. This makes them suitable to help and support humans in hazardous environments. They can handle dangerous, unknown objects with no risk for the operator's health. In this work we present a shared operation approach for the identification and localization of unknown hazardous objects as well as a 3D mapping approach for mobile robots in challenging environments. A shared control force-based grasping approach complete these two components and makes it easy for a human operator to grasp and retrieve unknown hazardous objects. Including the human expertise in the operation and control is additionally supported by providing intuitive visualization on different levels of abstraction. The presented approach was successfully evaluated with two different mobile robots within a field test.

Keywords: AI-enabled robotics; field robots; humanrobot-interaction; mapping; robotics in hazardous fields.

1 Introduction

1.1 Motivation

Manipulators in industrial production as well as mobile transport platforms in logistics are robust and mature. Hence, robot arms and mobile robots account for an important and big part of the value creation in many countries by moving, transporting and manipulation goods. The ability to move and handle objects reliably has been the most important ability of robots for many years.

3Open Access. © 2022 the author(s), published by De Gruyter. 🐨 🖙 This work is licensed under the Creative Commons Attribution 4.0 International License.

Recently, a change in value creation is taking place, away from mechatronics and towards more intelligence and autonomy based on advanced algorithms. The robot skills are improving and new applications are made possible outside industry and logistics.

In these new application fields like health care, public and private assistance and inspection the ability to quickly adapt to dynamic, changing environments and new tasks is essential. Therefore, reusable, modular and adaptive architectures are needed to comply with these requirements. Within this work, the focus is to help and support humans in hazardous environment or with hazardous objects and reduce environmental and health risks by applying intelligent mobile manipulators. Most other works are dedicated to a subset of the challenges that have to be addressed. Many works focus on grasping and handling of known or unknown objects. Others are more dedicated to advanced image or sensor data processing and interpretation. And there are works enabling robots to map unknown environments and enable them to safely navigate. Within this work it is necessary to address all challenges. The resulting issues require a shift in thinking - shared perception and control to reach the desired robustness und reliability.

This work presents a modular, reusable hardware and software approach that is based on standard off-the-shelf components together with Open Source libraries, machine learning models and frameworks like Google Cartographer, PyTorch and ROS. The use of powerful established implementations in combination with new software components is the basis of this work. These new components like the 3D mapping framework OpenVBD_Mapping are made available to the Community as Open Source repositories. The combination of easy to train instance segmentation networks with powerful 3D mapping and path planning functions makes this system highly flexible. On the other hand, the consequent integration of human expertise throughout the whole pipeline put a strong focus on the development to intuitive visualizations, operation and control concepts makes it easy to interact with the robot on all levels. Consequently, this contributes to a reusable

^{*}Corresponding author: Arne Roennau, FZI Forschungszentrum Informatik, Karlsruhe, Baden-Württemberg, Germany,

e-mail: roennau@fzi.de. https://orcid.org/0000-0002-6090-607X Johannes Mangler, Philip Keller, Marvin Grosse Besselmann, Nicolas Huegel and Ruediger Dillmann, FZI Forschungszentrum Informatik, Karlsruhe, Baden-Württemberg, Germany. https://orcid.org/0000-0002-2049-8219 (R. Dillmann)

and transferable shared grasping and retrieving approach for unknown, hazardous objects with a mobile robot platform.

1.2 Related work

The variety of autonomous mobile safety robots is large. Some of the robots enhance human safety by searching for landmines [1] or performing bridge inspections to detect cracks [2]. In the field of hazardous or nuclear objects the robot systems are almost exclusively teleoperated [3, 4]. Mapping the environment is one of the fundamental challenges in the use of autonomous mobile robots. Sensor-based sensing for mobile robots is typically performed using laser scanners, RGB (Red Green Blue) and RGB-D (Red Green Blue -Depth) sensors [5-8]. While simultaneous localization and mapping (SLAM) methods in two dimensional structured environments, such as office buildings, have been established for some time [9], three dimensional unstructured environments, e.g., outdoors, pose a particular challenge and are the subject of current research [10-12]. For the navigation of the robot, a distinction is made between global navigation with information about the environment, which can e.g. consist of a map, and local navigation without information about the environment [5]. One possibility for planning paths in unknown environments are so-called bug algorithms [13]. These move the robot in the direction of the target and react to too close objects by short path changes. Other work deals with the identification of dangerous spots based on images, e.g. to avoid rocks or soft terrain on a Mars expedition [14]. To support faster path finding, multiple robots can be used to share the acquired information with each other [15]. Grasping and handling objects in unknown environments based on visual and partly also haptic perception is an essential part of different domains, especially in the handling of hazardous materials. This is also an important topic in current research [16-20]. For the detection of obstacles and manipulable objects, segmentation and identification of related objects [21] plays an important role. Recent work uses e.g. well-known object models [22] or Deep Learning to narrow down the grasp selection [23].

1.3 Challenges within the context of ROBDEKON

Within the National Competence Center on Robotic Systems for Decontamination in Hazardous Environments – ROBDEKON – the research is evaluated in different application-oriented scenarios. Each of the scenarios was intensively discussed with industrial stakeholders from industry dealing with nuclear decommissioning and emergency on-call service experts for nuclear facilities. The handling and retrieval of unknown hazardous objects is a key use case in one of the scenarios. It comes with a number of challenges that have to be addressed:

- 1. Develop a mobile robot with robot arm and required sensor setup.
- 2. Create an online 3D map of the environment with no GPS localisation information.
- 3. Localise the mobile robot within this 3d Map with no GPS localisation information.
- 4. Identify and localise hazardous objects with only little information about the objects.
- 5. Navigate autonomously in the 3D Environment to desired goals.
- 6. Grasp and retrieve the objects in a shared operation and control approach.

All these functions are presented as follows. In Section 2.1 the development of the multi-purpose robot Husky is described. Section 2.2 outlines the shared force-based grasping approach for hazardous objects. Then Section 2.3 further describes the developed object identification and localisation concept. The highly efficient online 3D mapping component for mobile robots is explained in Section 2.4. Section 3 provides details on the experiments and evaluation of the previously presented components. In contrast to the other parts, the navigation and path planning component is only described briefly within the Section 3. Finally, in the last Section 4 the contributions of this work are summarized and discussed.

2 Approach

2.1 Hardware- and software-architecture

The work on grasping and recovering hazardous goods was executed and evaluated on the mobile robot Husky, a UGV (Unmanned Ground Vehicle) from Clearpath Robotics (see Figure 1). The robot was initially equipped with a 6 DOF (Degrees Of Freedom) robot arm (Universal Robots UR5) with a payload capacity of 5 kg as well as a Velodyne HDL-32E LIDAR (LIght Detection And Ranging) and a DGPS (Differential Global Positioning System) from NovAtel (SMART6-L GPS). The LIDAR is used together with an additional MicroStrain 3DM-GX3-25 IMU (Inertial Measurement Unit) for the 3D live mapping of the environment and at the same time localization within this map (SLAM).



Figure 1: Mobile robot husky with 6 DOF arm (left) and mobile ground-station with six monitors (right).

Up to now, the DGPS system has not been used, as the availability of GPS signals cannot be guaranteed in the foreseen scenarios. The official supported ROS interface from Clearpath Robotics to all components makes it possible to easily access and add new components.

First, a force-torque sensor (Schunk FTN-Delta SI-660-60) and a highly flexible three finger-gripper (Robotiq 3-Finger Adaptive Robot Gripper) were mounted on the robot arm. Moreover, a 3D camera (Intel RealSense d435i) was connected to the end of the arm giving an optimal view to potential objects close to the gripper. A second camera was mounted on the top of the platform. This 360° camera (Ricoh Theta V) provides a high-resolution panorama of the surroundings of the robot. It helps the operator to get a better overview of the current environment but is also used by the instance segmentation network to automatically detect potentially hazardous goods. Some hardware modifications improve the overall performance of the platform. The new LiPo (Lithium-Polymer) batteries improved the runtime of the robot, now allowing autonomous operation of up to 6 h. The off-road tires were changed to tires with less profile that create a smoother motion on hard surfaced like concrete or asphalt. If needed the tires can be switched within a few minutes, making it possible to adapt to specific mission requirements. Storage bins have been added to the robot platform to safely retrieve items during a mission. For this purpose, a general storage container in the form of an open-top box and for special (simulated) hazardous goods and an extra metal container have been mounted on top and front of the robot. In order to detect objects under insufficient external illumination, two high power LEDs were mounted on the front of the robot.

In addition, a mobile ground station was built to facilitate field tests. A stable, transportable box was constructed and equipped with two powerful PCs (one with GPU), six full-HD screens and a LiPo battery charger. Due to the large number of screens, different views, sensor data, 3D maps and status reports can be displayed in parallel. In the future, it is planned to expand the system to include internal batteries that can be used to operate the control centre without an external power supply, at least for a short time. The wireless network setup was improved regarding bandwidth by adding a high-performance IEEE 802.11ac access point to the ground station and antenna to the robot.

The built hardware does not claim to be used in a real scenario with radiating sources or a nuclear environment in general. The hardware was built to develop and test various software components and assistance functions. For the use in e.g. nuclear environments there are special robot systems e.g. from Telerob GmbH, which are not really useful for scientific development and evaluation.

The software architecture regarding the interaction with the human operator, the interaction between the components as well as the shared control and operation modes is visualized in Figure 2.

All skills described later in this work, the shared manipulation strategy, instance-based object detected, 3D mapping and path planning, were fully integrated and evaluated on this mobile husky platform.

2.2 Shared force-based grasping of unknown objects

The shared autonomy approach has the goal to increase the speed and success rate of complex robotic manipulation tasks with hazardous objects or in hazardous environments. Even in dark environments, humans have no problems grasping objects for which the positions are only roughly known. By touching the objects, these inaccurate positions are instantaneously corrected and it is possible to find a stable grasp. The grasping strategy follows this



Figure 2: Software architecture with shared operation and control approach.

concept by combining information from a human operator and a force-torque sensor on the robot. The operator takes over the object detection and the selection of a suitable grasping strategy. This strategy consists of a gripper mode, a grasp mode and the position as well as the orientation of the gripper (gripper pose) with respect to the object. A good gripper pose for the object is essential. In literature, there are many solutions to find suitable grasps for all kinds of objects. In the industrial environment, this challenge is typically called bin picking and has been solved in many applications. Nevertheless, the calculation and selection of suitable positions and orientations of the gripper for a successful grasp of objects is still a challenge if these objects are unknown, deformable or with sparse sensor data. Therefore, this task is highly dependent on visibility and lighting conditions. A deterioration of the sensor data quality directly leads to a more difficult object recognition and thus to worse grasping results. On the other hand, this task is trivial for a human. Even little information about the object is sufficient for a human to select and perform a successful grasp.

Two different grasp modes were implemented for grasping objects. The first grasp mode was developed for unknown objects of all kinds. Within this grasp mode, the gripper is first moved close to the target pose. Then the gripper is moved to the desired target pose using a force-sensitive compliance controller. If a physical contact is detected, a set of different reactions is automatically triggered. Depending on the triggered situation, the gripper pose is automatically corrected or the gripper is closed. The position correction compensates for inaccurate operator inputs and ensures that the gripper is positioned in a way that the object can be grasped and picked up successfully. If the fingers are touched by the object, the gripper target pose is shifted in a positive or negative direction along the *X*-axis of the gripper. This process is repeated multiple times, if further contacts are detected. The first grasp mode is visualized in a flow chart (see Figure 3).

A second grasp mode was developed for grasping objects with handles. For this purpose, additional steps were introduced into the algorithm and existing ones were modified to enable the touching of handles. More grasp modes were introduced for the easy handling of valves and picking up objects, but not further discussed in this work.

A gravity compensation of the gripper was implemented and subtracted from the measured values. The resulting forces and torques are only the contact forces acting on the wrist. The gripping force of the fingers cannot be adapted with the gripper used in this setup. The applied adaptive compliance control is based on previous work from Scherzinger et al. [24].

The shared manipulation tasks are operated via a graphical interface. The graphical interface displays the current environment close to the robot by means of a 3D point cloud from a RGB-D camera. The current position and kinematic configuration are also displayed using a 3D model of the robot. A virtual representation of the gripper is used to simply define the target gripper pose. The grasp mode is selected via a context menu. The grasp is then executed autonomously by the robot. Besides this shared control approach, the operator has a variety of assistance functions, in order to perform a wide range of tasks (see Figure 4). For the search-and-recovery tasks, for example, the following assistance functions are made available to the operator: 3D mapping of the environment, visual recognition of predefined objects, collision-free path planning on the 3D map, detailed images in colour and precise 3D point clouds, shared grasping of objects and supported picking and placing of objects.

2.3 Instance segmentation for object identification and localisation

The challenge regarding the recovery of unknown, hazardous objects is about recognizing hazardous objects in an unknown, cluttered environment and localize these objects with a mobile robot. This work is based on an initial visual detection using the camera data with deep neural networks. The robot is equipped with different cameras: one RGB-D camera on the robot arm and one 360° camera on the sensor frame on the back of the modified Husky robot. The camera on the robot arm provides colored depth information and has great flexibility in terms of its orientation. It is mainly intended for the perception of objects close to the robot, which can be inspected from different



Figure 3: Flow chart for the grasp mode for unknown objects. The user inputs are shown in the top blue box. The user initially selects the object and grasping strategy. The robot in the bottom orange box then automatically executes the force-adaptive grasping process.



Figure 4: Intuitive and fast parametrization directly in rviz where also the RGBD sensor data are visualized.

perspectives and to parametrize the force-based grasping strategy. The camera on the sensor frame, is a statically mounted 360° camera to provide a fast overview during the exploration phase. It can provide initial indications of where potentially hazardous objects are located (see Figure 5).

Many state-of-the-art works in object detection rely on artificial deep neural networks. But usually large data



Figure 5: Masked 360° image with the hazardous goods container detected by the mask R-CNN. Without the masking of the robot, false detections could occur on the robot itself.

sets are required to detect arbitrary objects using such approaches. However, the requirements for the scenario of retrieval of hazardous materials are special. In most cases, no large data sets are available for the target objects to be detected, e.g. specific barrel types or other rare hazardous material containers. In addition, many false detections can occur if the corresponding operational environment is not known to the model.

A Mask R-CNN architecture is used for this purpose. Pre-trained on the comprehensive MS-COCO dataset, a fine-tuning of the detection components in the higherlevel layers with a few annotated images of the specific categories is sufficient to enable an accurate detection or segmentation of the objects. The Mask R-CNN architecture for neural networks, first presented in Mask R-CNN [25], is an extension of Faster R-CNN [26]. Parallel to the bounding box prediction of the objects, an exact mask representing all pixels that belong to the respective object is predicted. This mask provides more information about the pose of the object, but in return the computational effort for processing the images increases.

Once a small data set has been created for the hazardous material to be detected, it can be augmented as desired and is then suitable for training the designed neural detection architectures (see Figure 6). In addition, the background of the training images can be replaced by background images that are expected in the mission scenario. However, augmentation strategies also have their limitations when it comes to augmenting smaller data sets. There is a danger of overfitting. This refers to the process when the model parameters are optimized too much for the existing training data, so that the system's detection capabilities degrade. In order to prevent this, a transfer learning method is applied on the model parameters of the network architecture, which was pre-trained on large open available data sets (e.g. MS-COCO). These large data sets enable basic capabilities in the model with respect to image recognition and visual information processing. Subsequently, the parameters are adapted to the concrete, very specific hazardous goods in the second training phase, the so-called fine-tuning. The task-oriented adaptation takes place in the higher, more abstract layers of the network architecture and requires significantly less data.

Two neural network architectures are currently applicable in this work: Mask-R-CNN and CenterMask2-Lite [27]. Both perform very well in the COCO benchmark and show robust capabilities, also compared to many approaches in the literature. For example, the CenterMask2-Lite, a one-stage model, has been designed to be as lightweight as possible to be suitable for mobile computing devices. The Mask-R-CNN, a two-stage model, is designed for detection quality and requires more computational effort. More common architectures usually only provide bounding boxes for each detection. However, masking provides additional



Figure 6: Object identification and localization pipeline based on instance segmentation architectures and 3D maps.

information about the orientation and the exact position, which is of great importance in robotic manipulation tasks. The two currently used architectures (Mask-R-CNN and CenterMask2-Lite) differ particularly in their complexity and the computational requirements on the hardware. As default, the images are processed on the ground station, where sufficiently powerful hardware is available. For missions, where no continuous communication with the ground station can be guaranteed, the lightweight CenterMask2-Lite architecture could serve as an onboard fallback. Then the image processing can take place locally on the mobile robot Husky. But in general, the Mask R-CNN has proven to be significantly more robust regarding the generalization capabilities with respect to unknown lighting conditions and new unseen instances of known categories.

A further challenge is to extract the 3D position of the target objects. Other classical vision or image processing approaches are able to recognize the 6D pose of an object in addition to the object category [28–30]. The Mask R-CNN approach as well as the CenterMask2-Lite do not provide any spatial information of the detected objects except the pixel coordinates. There are new NN architectures such as Mesh R-CNN [31] that address this lack. This approach is not used, because it requires a very large amount of training data and still does not provide the accuracy required for mobile manipulation tasks.

In this work, the robot platform autonomously navigates to the area where objects are initially detected in the 360° camera image. The target coordinates are determined by raycasting on the 3D map captured by the 3D LiDAR. This requires a transformation of the pixel coordinates to a ray in the global reference coordinates. The intersection point with the voxels in the 3D map are found by raycasting in the temporary position of the object, which is sufficiently accurate for rough navigation. The RGB-D camera on the robot arm is then used in the target area to re-localize the hazardous object. With this camera, the object can be localized with a higher precision and provide the desired depth information. A DBSCAN clustering algorithm (Density-Based Spatial Clustering of Applications with Noise) is used to preprocess the detections, which filters inaccuracies and prevents incorrect information from being entered into the map. The filter approach checks if the detections are occasional false detections or continuous detections with high confidence at the same or a nearby position. In this case, the detected and localized objects are additionally added to the 3D map as interactive markers to enable the operator to easily manage the found potentially hazardous objects.

2.4 Online 3D mapping for mobile robots

The 3D mapping approach is based on Google's Cartographer, which was already successfully used in various works [32]. This is an open source software that is able to perform a real-time SLAM on the basis of laser scans or point clouds. In previous works the authors have only used Cartographer for 2D indoor navigation. In this work, a 3D mapping approach was developed enhancing the previous functionality. For this purpose, different sensor setups were tested and evaluated. In contrast to the 2D case, the 3D SLAM requires the values of an inertial measurement unit (IMU) in addition to the 3D point cloud data. The IMU provides information about the orientation, as well as the linear and angular accelerations. These are used by Cartographer to transform the input point cloud into the ground plane in order to reduce the problem of scan matching by 2 dimensions (roll and pitch angles), resulting in a higher processing speed of the individual point clouds. In addition to pure scan matching, Cartographer offers the possibility of feeding various other localization sources into the algorithm, such as pre-calculated odometry or the coordinates of a global navigation satellite system (e.g. GPS). In this work, the wheel-based odometry of the Husky was used as additional input. Although the odometry has a high slip, it nevertheless brings an enormous advantage in certain scenarios compared to a setup without this information.

After establishing a stable SLAM system, the goal was to create a 3D map from all the input data. In general, the Velodyne LiDaR point clouds are rather sparse, at least vertically. Despite this sparse initial scan data, the resulting total point clouds are very dense due to the continuous movement of the robot. On average, the Velodyne Puck accumulates about 14 million points per minute. A naive approach raises several problems. On the one hand, the map occupies a huge part of the system's memory, which must also be kept available during the entire navigation. On the other hand, it is very time-consuming and inefficient to plan on the entire point cloud. Therefore, it is better to introduce an abstraction layer of the map representation instead of using the full point clouds.

For this purpose, the environment is first divided into a uniform 3D grid in which each cell covers a defined area depending on the spatial resolution and stores whether the corresponding cell is occupied or free. However, this level of abstraction is not yet sufficient because the grid grows too quickly due to its three dimensions. The main reason for this is that memory consumption does not scale directly with the number of relevant grid cells. Instead, the data structure grows proportionally to the space that embeds the data. A prominent example of this is the mapping of free areas on the map. These take up a large part of all cells, but provide relatively little information about the surrounding area.

To counteract this problem, instead of a regular complete 3D grid, more efficient tree structures are increasingly used as data management in such cases. A frequently used implementation for this is the ROS implementation OctoMap [33], which is based on octrees. This is a 3D data representation that discretises the space similar to occupancy grids and also offers efficient memory management. To insert new data into the octree, a ray casting beam is projected from the sensor to the data point through the entire data structure for each new point in a sensor scan. All traversed grid cells are classified as free or occupied with the help of a probability distribution. The result is a probabilistic occupancy map, similar to the occupancy grid maps used in the 2D case. However, one problem with the OctoMap is that the insertion of new data points and ray casting is too slow due to the data structure in relation to the sensor data rate. Since the mapping cannot insert new data fast enough, new measurements from the sensor are often ignored, resulting in a sometimes inaccurate and incomplete map.

Therefore, a new 3D mapping framework was developed that focuses on the fast and efficient integration of new data points. In this approach OpenVDB is used as an efficient data storage layer for the volumetric data [34]. OpenVDB has its origins in computer animation and is mainly used for volumetric modelling in animated movies. In this field of application, it is essential to render data structures efficiently together with optimised ray casting algorithms. However, this property can also be applied to the ray casting operations during the mapping process, which makes the OpenVDB much faster than OctoMap, see Figure 7. OpenVDB has also been optimised in terms of memory consumption. It also encodes the free space over low-resolution voxels in a memory-efficient way, similar to the described octree approach. In addition, it provides built-in compression algorithms to keep the data structure as small as possible.

In addition to replacing the entire mapping data storage layer, the way in which new sensor data is integrated has been fundamentally revised. The ray casting of all points was divided into two steps. First, all rays are projected through the data structure as usual. However, instead of directly updating the probabilities, a temporary grid is first created in which it is noted whether a cell needs to be updated or not. As soon as this has been done for all points, the second step begins. The temporary grid is compared with the actual map in order to update all necessary grid occupancy probabilities. This new 3D mapping approach showed an up to ten times faster mapping speed, depending on the map resolution and maximum sensor range. Figure 7 shows in detail how the two data structures behave with respect to different resolutions and ray casting distances.

Due to these good results, this new mapping approach was made available to a larger community by publishing the open source software package vdb_mapping [35]. In addition, all algorithms like the path planning for mobile robots, which were previously based on OctoMap,



Figure 7: Comparison of the calculation time for datapoint insertion between OctoMap and our OpenVDB implementation with respect to different resolutions and raycasting distances [10].

were ported to the new map format. An example of a dense VDB map of the FZI lab building that was created with a flying drone equipped with a LIDAR is shown in Figure 8.

In many situations multiple robots are applied for a difficult task. Moreover, that map should also be available at the base station. Therefore, the maps have to be synchronised between the individual robot systems as well as the base station over the network for an effective use. This generates large, impracticable amounts of data. To counteract this effect, we have extended the approach with a remote mapping function. The idea here is to transmit only new parts of the environment that are currently detected by the lidar sensors over the wireless network instead of the whole map. For this purpose, we exploit the already existing data integration of VDB-Mapping. Once a new point cloud is acquired, ray casting through the map structure is performed for each data point to determine which cells are occupied and which are free. This information is stored in an efficient bit structure within the active mask of a VDB tree. These so-called update grids only cover the area of the map currently perceived by the LIDAR. In the approach the update grids are used to determine which occupancy probabilities in the map need to be updated to represent the current LIDAR data. As a result, it is only necessary to check within the update grid whether an update must be carried out on the corresponding cells of the map. Since the update grids do not contain any data except for the bit flag, they are relatively small and can therefore be sent over the network with a high update rate. To further reduce the size, the sparsely filled VDB tree is compressed, serialised and made available to other remote



Figure 8: OpenVDB map of the backyard between the FZI lab buildings [10].

mapping nodes as a ROS topic. These nodes then unpack the data and perform the same update step as the robot in order to obtain the same map of the environment without having to send it over the network.

3 Experiments and evaluation

The evaluation concept was developed together with the partners of the German national competence center on robotic systems for decontamination in hazardous environments – ROBDEKON. Especially, the definition of the evaluation scenario as well as the identification of the requirements were an important part, where feedback and input were needed from industrial application partners from the field of nuclear decommissioning and emergency



Figure 9: Illustration of the scenario during the evaluation (left), robot team with husky and ANYmal retrieving a metal object (right).

forces for a nuclear incident. The evaluation scenario can be described as follows: A traffic accident occurs during the transport of a medical radiation source (e.g. Iridium 192). The protective container is destroyed and a radioactive object is exposed. Shared autonomy robots are requested to find, grasp and safely retrieve the radioactive object. Two different robots should be deployed via a remote command and control center (ground station). One robot is supposed to quickly scan the area and perform a search for the radioactive object. As soon as it finds potentially hazardous objects, the first robot sends the position of these objects to the second robot with a manipulator and protective container. The second robot can then check and confirm the found hazardous objects. The operator uses the shared control approach to handle and safely store the objects with the manipulator. Both robots should be managed simultaneously by only one operator.

The scenario described was successfully demonstrated during a ROBDEKON project field test evaluation at the Fraunhofer IOSB. Figure 9 (left) shows the setup and shows the scenario sequence that was evaluated. On the right side of this Figure, the robot team, composed of the Husky and the walking robot ANYmal found a metal object.

The first task was to explore the area and locate hazardous objects. This task was performed by ANYmal because of its ability to move quickly and highly flexible in unknown environments. The second task involved grasping and retrieving the hazardous objects. This task was carried out by the Husky robot system. During the field test, the area was explored by ANYmal with a pattern defined by the operator. As soon as a hazardous material was identified and localised, the ground station was informed about the potential object. The position was automatically set as target way point for Husky. Both robots used the 3D VDB map to plan collision free paths in this previously unknown environment. The successful arrival of Husky in front of the hazardous object was visualised at the base station. Then the operator did a close inspection of the potential object with different views with the integrated RGB-D camera on the robot arm. After approval, the operator selected and parameterized the force-based grasp strategy. The object was not grasped on the first attempt. But the operator was able to readjust the grasp strategy and the hazardous object was successfully recovered. After confirmation by the operator, Husky automatically navigated to the next potentially hazardous object that was found by ANYmal. All hazardous objects were collected with the presented shared operation and control approach. By deploying the robot team, the search and recovery could be performed in parallel with only one operator. This proved to be robust and highly efficient and showed great potential for further applications with multi-robot teams.

4 Discussion and summary

The challenges in handling hazardous unknown objects have been addressed by developing an efficient 3D mapping, an object segmentation and localisation as well as a force-based manipulation approach. All three components have been integrated in a shared operation and control architecture. Moreover, they were evaluated on the modified Husky and ANYmal robot in a field test with a reviewing external jury. The 3D mapping is based on the integration of the powerful OpenVDB data storage layer into a 3D grid map approach based on the robust and reliable Cartagrapher SLAM stack. The object segmentation and localisation combine a pre-trained 2D instance segmentation approach with a data augmentation and 3D grid map-based object localisation to make it a flexible and easy to adapt system for mobile robots. Currently, this approach is only abled to identify previously trained objects. Unknown objects are not addressed here, but could be highlighted by an anomaly detection based on the reconstruction error with an autoencoder NN. The shared force-based grasping strategies make it possible to include intelligent assistance functions with the expertise end experience of human operators.

The 3D mapping approach outperforms other SoA grid based-mapping implementations regarding the performance of online mapping with high resolution 3D sensor data. The VDB mapping framework is available as open source stack and has been successfully tested on various mobile robots and sensor setups. The object detection and localisation is based on a powerful Mask R-CNN that makes it possible to detect previously unknown objects with only a small (5-10) number of sample pictures with very high confidences despite partial occlusions. In combination with the 3D grid map it is possible to use these good segmentation results to roughly localise the hazardous objects. The force-based shared grasping strategy has been evaluated on a high number of different objects with an overall success rate of approx. 90%. In combination with the intuitive visualization of the environment, found objects and interaction control interfaces this works provides a powerful toolset for grasping and handling hazardous, unknown objects with a mobile robot like the presented Husky robot. Within the field test the shared operation and control architecture convinced the expert jury regarding its performance, robustness and flexibility. Therefore, this work contributes to make the work with hazardous materials and objects safer for humans, by including their expertise from a safe distance. In the future, we plan to extend this work by enhancing the 3D path mapping and planning capabilities as well as further improve the manipulation strategies by adding learning from experience.

Author contributions: All the authors have accepted responsibility for the entire content of this submitted manuscript and approved submission.

Research funding: The research leading to these results was supported by the ROBDEKON project funded by the German Federal Ministry of Education and Research under grant agreement No. 13N14679.

Conflict of interest statement: The authors declare no conflicts of interest regarding this article.

References

- [1] P. Kopacek, "Autonomous mobile robots," *Int. J. Autom. Austria (IJAA)*, vol. 18, pp. 53–59, 2010.
- [2] H. M. La, N. Gucunski, S. H. Kee, J. Yi, T. Senlet, and L. Nguyen, "Autonomous robotic system for bridge deck data collection and analysis," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2014, pp. 1950–1955.
- [3] R. Bogue, "Robots in the nuclear industry: a review of technologies and applications," *Ind. Robot*, vol. 38, no. 2, pp. 113–118, 2011.
- [4] D. W. Seward and M. J. Bakari, "The use of robotics and automation in nuclear decommissioning," in *Proceedings of the 22nd International Symposium on Automation and Robotics in Construction ISARC*, 2005, pp. 11–14.
- [5] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: a survey," *J. Intell. Rob. Syst.*, vol. 53, no. 3, pp. 263–296, 2008.
- [6] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the RGB-D SLAM system," in 2012 IEEE International Conference on Robotics and Automation, IEEE, 2012, pp. 1691–1696.
- [7] F. Moosmann and C. Stiller, "Velodyne slam," in *Proceedings* of the IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011, pp. 393-398.
- [8] T. Schöps, J. Engel, and D. Cremers, "Semi-dense visual odometry for AR on a smartphone," in *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality* (ISMAR), IEEE, 2014, pp. 145–150.
- [9] S. Thrun, "Robotic mapping: a survey," in *School of Computer Science*, Pittsburgh, PA, USA, Carnegie Mellon University, 2002.
- [10] M. G. Besselmann, L. Puck, L. Steffen, A. Roennau, and R. Dillmann, "VDB-mapping: a high resolution and real-time capable 3D mapping framework for versatile mobile robots," in *Proceedings of the IEEE 17th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2021, pp. 448–454.
- [11] T. Shan and B. Englot, "Lego-loam: lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 4758–4765.
- [12] P. Wolf, A. Vierling, T. Ropertz, and K. Berns, "Advanced scene aware navigation for the heavy duty off-road vehicle Unimog," in *IOP Conference Series: Materials Science and Engineering*, vol. 997, IOP Publishing, 2020, p. 012093.
- [13] N. Buniyamin, W. W. Ngah, N. Sariff, and Z. Mohamad,
 "A simple local path planning algorithm for autonomous mobile robots," *Int. J. Syst. Appl. Eng. Dev.*, vol. 5, no. 2, pp. 151–159, 2011.
- [14] M. Ono, T. J. Fuchs, A. Steffy, M. Maimone, and J. Yen,
 "Risk-aware planetary rover operation: autonomous terrain classification and path planning," in *Proceedings of the IEEE Aerospace Conference*, IEEE, 2015, pp. 1–10.

- [15] J. H. Jung, S. Park, and S. L. Kim, "Multi-robot path finding with wireless multihop communications," *IEEE Commun. Mag.*, vol. 48, no. 7, pp. 126–132, 2010.
- [16] R. Grimm, M. Grotz, S. Ottenhaus, and T. Asfour,
 "Vision-based robotic pushing and grasping for stone sample collection under computing resource constraints," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA*), IEEE, 2021, pp. 6498–6504.
- [17] L. Manuelli, W. Gao, P. Florence, and R. Tedrake,
 "kpam: keypoint affordances for category-level robotic manipulation," in *The International Symposium* of Robotics Research, Cham, Springer, 2019,
 pp. 132–157.
- [18] S. Ottenhaus, D. Renninghoff, R. Grimm, F. Ferreira, and T. Asfour, "Visuo-haptic grasping of unknown objects based on Gaussian process implicit surfaces and deep learning," in *Proceedings of the IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, IEEE, 2019, pp. 402–409.
- [19] C. Pohl and T. Asfour, "Probabilistic spatio-temporal fusion of affordances for grasping and manipulation," *IEEE Rob. Autom. Lett.*, vol. 7, no. 2, pp. 3226–3233, 2022.
- [20] J. C. V. Tieck, K. Secker, J. Kaiser, A. Roennau, and R. Dillmann, "Soft-grasping with an anthropomorphic robotic hand using spiking neurons," *IEEE Rob. Autom. Lett.*, vol. 6, no. 2, pp. 2894–2901, 2020.
- [21] M. Durner, W. Boerdijk, M. Sundermeyer, W. Friedl, Z. C. Marton, and R. Triebel, "Unknown object segmentation from stereo images," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (*IROS*), IEEE, 2021, pp. 4823–4830.
- [22] N. Vahrenkamp, E. Koch, M. Wächter, and T. Asfour,
 "Planning high-quality grasps using mean curvature object skeletons," *IEEE Rob. Autom. Lett.*, vol. 3, no. 2, pp. 911–918, 2018.
- [23] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *Int. J. Robot Res.*, vol. 34, nos. 4–5, pp. 705–724, 2015.
- [24] S. Scherzinger, A. Roennau, and R. Dillmann, "Forward dynamics compliance control (FDCC): a new approach to cartesian compliance for robotic manipulators," in *Proceedings of the IEEE/RSJ International Conference on*

Intelligent Robots and Systems (IROS), IEEE, 2017, pp. 4568–4575.

- [25] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.
- [26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, vol. 28, NIPS 2015, 2015.
- [27] Y. Lee and J. Park, "Centermask: real-time anchor-free instance segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13906–13915.
- [28] T. Acharya and A. K. Ray, *Image Processing: Principles and Applications*, Hoboken, New Jersey, USA, John Wiley & Sons, 2005.
- [29] E. Brachmann, F. Michel, A. Krull, M. Y. Yang, and S. Gumhold, "Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3364–3372.
- [30] A. Krull, E. Brachmann, F. Michel, M. Y. Yang, S. Gumhold, and C. Rother, "Learning analysis-by-synthesis for 6D pose estimation in RGB-D images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 954–962.
- [31] G. Gkioxari, J. Malik, and J. Johnson, "Mesh R-CNN," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9785–9795.
- [32] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 1271–1278.
- [33] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Aut. Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [34] K. Museth, "VDB: high-resolution sparse volumes with dynamic topology," ACM Trans. Graph. (TOG), vol. 32, no. 3, pp. 1–22, 2013.
- [35] M. G. Besselmann, Github.Com, 2022. Available at: https:// github.com/fzi-forschungszentrum-informatik/vdb_mapping 2022.