Autonomous Assistance for Versatile Grasping with Rescue Robots

Marius Schnaubelt, Stefan Kohlbrecher, and Oskar von Stryk

Abstract— **The deployment of mobile robots in urban search and rescue (USAR) scenarios often requires manipulation abilities, for example, for clearing debris or opening a door. Conventional teleoperated control of mobile manipulator arms with a high number of degrees of freedom in unknown and unstructured environments is highly challenging and error-prone. Thus, flexible semi-autonomous manipulation capabilities promise valuable support to the operator and possibly also prevent failures during missions. However, most existing approaches are not flexible enough as, e.g., they either assume a-priori known objects or object classes or require manual selection of grasp poses. In this paper, an approach is presented that combines a segmented 3D model of the scene with grasp pose detection. It enables grasping arbitrary rigid objects based on a geometric segmentation approach that divides the scene into objects. Antipodal grasp candidates sampled by the grasp pose detection are ranked to ensure a robust grasp. The human remotely operating the robot is able to control the grasping process using two short interactions in the user interface. Our real robot experiments demonstrate the capability to grasp various objects in cluttered environments.**

I. INTRODUCTION

Mobile ground robots provide remote presence to human operators and enable them to perceive and act from a safe distance. This allows the operator to perform tasks in USAR missions which otherwise would pose high risks to human response forces, e.g., due to radiation, toxic fumes, dangerous materials or collapsing buildings. The deployment of a mobile robot manipulator can strongly help to minimize the involved risks. Search and rescue robots in disaster scenarios are typically facing an a-priori unknown environment which might also be degraded. Frequently required complex manipulation tasks include debris removal, closing or opening valves as well as opening doors. Teleoperated control of a remote mobile robotic manipulator with a high number of degrees of freedom (DOF) under such challenging conditions is slow, error-prone and also requires suitable low-latency, high-bandwidth communication between robot and operator. Thus, supporting the human operator via robot onboard autonomous functions is desirable. On the other hand, fully autonomous manipulation is also likely to fail under such conditions. Therefore, the usage of autonomous manipulation approaches under the supervision of the remote human operator aims at increasing the quality and speed of manipulation control while simultaneously reducing the operator workload and interactions needed to control the USAR robot. Furthermore, the flexibility and control to handle a large variety of manipulation tasks including previously unknown ones is maintained. This paper focuses on the relevant use case of debris removal which requires the ability to grasp previously unknown objects as most existing approaches are not flexible enough. The contributions of this paper include:

1) A method for grasping a-priori unknown objects with an easy-to-use operator interface by combining a segmented Truncated Signed Distance Function (TSDF) representation of the scene with a CNN-based grasp pose detection.
2) A geometric segmentation approach with increased robustness compared to existing approaches.

All authors are with the Simulation, Systems Optimization and Robotics Group, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany. {schnaubelt, kohlbrecher, stryk}@sim.tu-darmstadt.de

II. RELATED WORK

Currently deployed mobile rescue robot manipulators are teleoperated with direct joint control or end-effector control in Cartesian space using inverse kinematics (IK). Brüggemann *et al.* [1] introduce a more intuitive way of control by coupling movements of the operator's arm to movements of the manipulator arm with the help of multiple inertial measurement units (IMUs) mounted on the operator's arm. However, these approaches require an experienced operator and are highly demanding due to limited situational awareness. Klein *et al.* [2] support the operator by segmenting possible grasp objects in RGB images using a saliency-based approach. The manual selection of a suitable grasp pose in the 2D image for the automated grasp execution requires the operator to have expert knowledge.

Romay *et al.* [3] use a semi-autonomous approach which represents a-priori known objects using object templates containing suitable grasp poses and affordance axes. Object templates provide an efficient representation of robot manipulation capabilities. However, they need either an experienced operator or a (semi-)automated fitting method to align the template with the object using the sensor data provided by the robot.

Klamt *et al.* [4] expand the concept of grasp templates to a fully autonomous manipulation approach by using a semantic segmentation for object detection supplemented with an object pose estimation. Additionally, grasp poses are transferred from the canonical model to novel instances of the same category. This enables the manipulation of known and similarly shaped objects, but the generalization capabilities are limited.

De Gregorio *et al.* [5] grasp previously unknown objects using an industrial robot arm by creating a 3D TSDF reconstruction of the scene. Subsequently, the scene is segmented into objects using a plane-based approach. Grasp points are

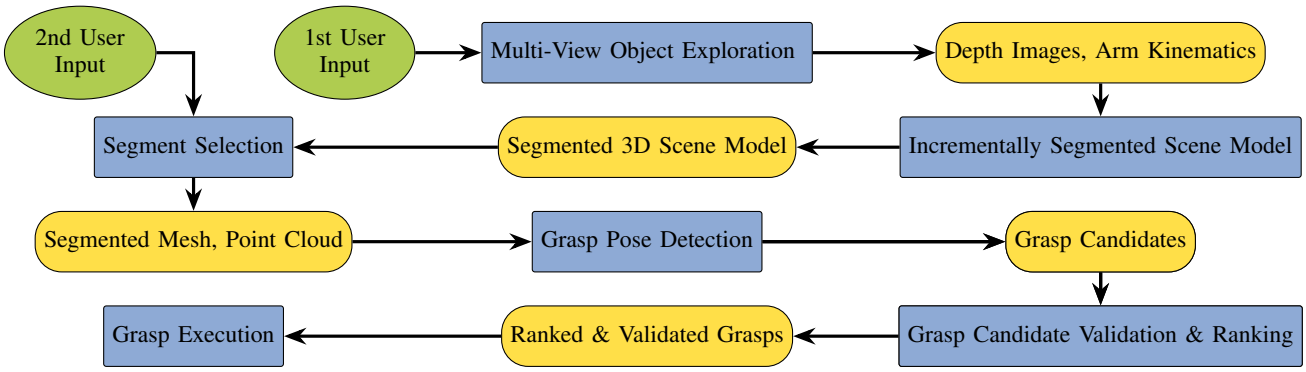
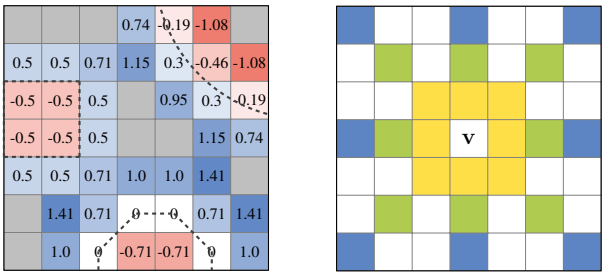


Fig. 1: An overview of the components used to enable grasping of unknown objects. The robot operator is able to select the object to grasp by selecting a segmented object. Blue objects denote data processors whereas yellow objects represent data types.



(a) (b)

Fig. 2: (a) A 2D TSDF voxel grid. Each voxel stores the truncated distance from the voxel center to the closest surface (dotted lines). Undefined voxels are marked in gray. (b) The neighborhood of $\mathbf{v}$ considered for $\rho_{\max} = 3$.

detected using a planar grasp algorithm that extracts slices of the 3D object along the principal axis. Although this approach enables the grasping of unknown objects and the TSDF representation provides a reconstruction of the scene that is robust to noise, the planar object segmentation renders it unsuitable for generic usage in rescue scenarios.

Vezzani *et al.* [6] use superquadric functions to model the object and volume that can be grasped by the hand of the humanoid robot. By utilizing superquadrics fitted to the sensor data, even occluded regions can be taken into account. Afterwards, they solve a nonlinear constrained optimization problem to find a suitable grasping pose while ensuring obstacle avoidance by using constraints. However, superquadrics lack appropriate distance functions and collision detection algorithms which makes them unappealing compared to other approaches, for example, TSDFs.

Instead of using a fully automated segmentation approach, Butler *et al.* [7] combine a human-aided segmentation approach with autonomous grasp execution. While manually segmenting the 3D sensor data is accurate, the process is time-consuming and raw sensor data is noisy.

Pas *et al.* [8] present a model-free, autonomous approach for grasping objects using data of an RGB-D camera. Antipodal grasp poses are sampled in point clouds and classified using a CNN. Using a heuristic, the best grasp sample is selected and executed which leads to a stable grasp. There is no direct control over the grasping process which renders the approach unsuitable for the direct application in the field of rescue robotics.

III. METHODS

In this chapter, we present our concept for assisted grasping of unknown objects in complex scenarios, which is outlined in Figure 1. The approach combines the generation of a 3D incrementally segmented scene model (ISSM) with a grasp pose detection which classifies sampled grasp poses under the supervision of the robot operator.

The robot operator initiates the multi-view object exploration by selecting the area of interest in the user interface. Then, based on multiple view poses, an ISSM is created. By selecting an object segment in the 3D scene, the robot operator can select the object which should be grasped.

The selected object is then extracted as a point cloud from the ISSM and used to generate scored grasp candidates. Subsequently, the grasp candidates are validated and all valid grasps are ranked based on multiple criteria aiming at providing a reliable grasp. Afterwards, the best possible grasp is executed in the grasp execution stage.

A. Multi-View Object Exploration

As the environment around the robot is assumed to be unknown and populated with arbitrary objects to be potentially manipulated, the robot operator needs to initiate the robot's exploration of the area of interest. The operator is assisted by a 3D visualization of the environment generated from the robot's sensors, primarily the lidar. Using the user interface, the operator can place a 3D point in the environment which marks the center of the area of interest. Now, several joint configurations are autonomously planned for the arm such that the depth camera captures the area of interest from different viewpoints.

B. Incrementally Segmented Scene Model

The estimation of suitable grasp poses is based on a segmented model of the scene which is created following an

approach by Tateno *et al.* [9]. First, a noise-robust 3D TSDF map is built from depth data that implicitly describes object surfaces by using a discretized voxel grid which stores the truncated distance from the voxel center to the closest object surface (see Figure 2a for a 2D example). Subsequently, to update the ISSM, each depth image is segmented using a geometric segmentation approach and the resulting segments are propagated into the global ISSM. The segment labels are propagated by matching the segment labels in the segmented depth image with those in the ISSM to ensure consistent segment labels in the ISSM. Finally, the segment update stage updates the labels assigned to each voxel using a confidence-based approach.

1) Depth Image Segmentation: Instead of using the segmentation approach suggested by Tateno *et al.* [9], we extend the segmentation approach used by Rünz *et al.* [10] with increased noise robustness as proposed by Ückermann *et al.* [11]. The edge image is computed by applying a threshold to the sum of the depth-discontinuity term ϕ_d and the concavity term ϕ_c

$$\phi_d + \lambda\phi_c > \tau, \tag{1}$$

with the edge threshold τ and the depth-discontinuity weight λ. In order to reduce the influence of noisy normal estimations, we compute both terms by averaging over different window sizes. The concavity term ϕ_c is given by

$$\phi_c = \frac{1}{\rho_{\max}} \sum_{\rho=1}^{\rho_{\max}} \left(\max_{i \in \mathcal{N}_\rho} \begin{cases} 0, & (\mathbf{v_i} - \mathbf{v}) \cdot \mathbf{n} < 0 \\ 1 - (\mathbf{n_i} \cdot \mathbf{n}), & \text{else} \end{cases} \right), \tag{2}$$

where $\mathcal{N}_\rho$ is the neighborhood of the 3D point $\mathbf{v}$ with maximal window size $\rho_{\max}$ and $\mathbf{n}$ denotes the corresponding normal while $\mathbf{v_i}$ and $\mathbf{n_i}$ are the i-th neighboring point and normal. The neighborhood is composed of eight pixels along the vertical, horizontal and diagonal directions with a distance of ρ steps from the center point, see Figure 2b for a visualization. The depth-discontinuity term ϕ_d is computed using

$$\phi_d = \frac{1}{\rho_{\max}} \sum_{\rho=1}^{\rho_{\max}} \left(\max_{i \in \mathcal{N}_\rho} \left\{ \left| (\mathbf{v_i} - \mathbf{v}) \cdot \mathbf{n} \right| \right\} \right). \tag{3}$$

After the edge map is computed, the segments are extracted from the binarized edge image by using a connected components algorithm. Additionally, the unsegmented border pixels are assigned to adjacent segments by searching the pixel with minimal Euclidean distance to the respective border point [12].

C. Segment Selection

Thanks to the segmented 3D scene model, the robot operator can be provided with an immersive view of the objects in the scene. Then, the operator can choose an object segment which should be grasped by selecting it in the 3D visualization of the segmented scene. Finally, a point cloud of the selected object is extracted from the ISSM.

D. Grasp Pose Detection

Suitable grasp poses are detected by sampling grasp poses in the point cloud and scoring the resulting samples encoded as multi-channel images using a CNN as presented by Pas *et al.* [8]. The detected grasp poses are antipodal grasps which can guarantee a stable force-closure grasp for two-fingered hands [13].

E. Grasp Candidate Validation and Ranking

The available grasp candidates are validated by checking if the grasp pose has an IK solution that is free of collisions with itself and the environment of the robot. Afterwards, the score given by the grasp pose detection is augmented by additional grasp quality metrics in order to find the best grasp. Given the set of grasp pose candidates, we want to select the grasp which maximizes the proposed grasp quality metric Q_{grasp} consisting of three additional metrics that augment the grasp score Q_{GPD} estimated by the grasp pose detection. For each grasp pose candidate $\mathbf{x}$, we compute an IK solution $\mathbf{q} = \{q_1, q_2, \dots q_n\}$ such that the robot gripper reaches the grasp pose. The first metric applies a linear L1 loss to the Euclidean distance between the center of mass (COM) and the base link of the robot

$$Q_{\text{COM}}(\mathbf{q}) = \left\| \text{COM}(\mathbf{q}) \right\|_2, \tag{4}$$

where the function $\text{COM}(\mathbf{q})$ computes the COM of the robot resulting from the joint angle configuration $\mathbf{q}$. A COM that has a larger distance from the robot's base link is potentially less stable and therefore not desired. This is caused by the fact that the robot arm with the lifted object creates a torque which needs to be compensated by the base of the mobile robot. If the torque exceeds a limit, the robot might start tilting during the grasp resulting in possible damage to the robot.

Additionally, we seek for maximal joint range availability [14] in order to minimize the possibility that a joint will reach a mechanical limit. Therefore, we introduce a Lorentzian penalty

$$Q_{\text{lim}}(\mathbf{q}) = \frac{1}{n} \sum_{i=1}^{n} \log \left(\frac{1}{2} \left(\frac{q_i - a_i}{c(a_i - q_{i,\max})} \right)^2 + 1 \right), \tag{5}$$

$$\text{with } a_i = \frac{q_{i,\max} + q_{i,\min}}{2}, \tag{6}$$

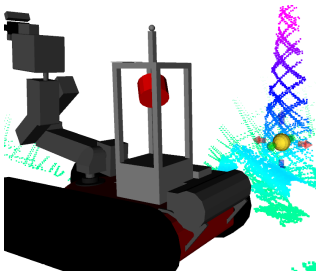
for joint angles deviating from the center between the i-th joint's limits $q_{i,\min}$ and $q_{i,\max}$, $i = 1, \dots n$. Here, c is used to scale the penalty. Finally, we want to avoid grasps near singularities because of the risk of losing one or more DOF. Thus, we introduce the distance to kinematic singularities

$$Q_{\text{sing}}(\mathbf{q}) = \sqrt{\det\left(\mathbf{J}(\mathbf{q})\ \mathbf{J}^\top(\mathbf{q})\right)} = \prod_{i=1}^{n} |\sigma_i|. \tag{7}$$

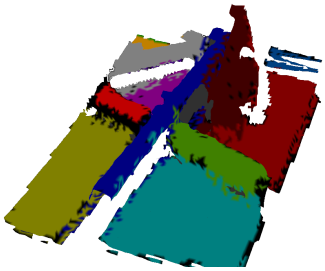
The distance Q_{sing} is the product of the singular values of the Jacobian matrix $\mathbf{J}(\mathbf{q})$ of the robot manipulator and can be regarded as a measure for the distance from a kinematic singularity [15]. This yields the final grasp quality metric

$$Q_{\text{grasp}} = Q_{\text{sing}}(\mathbf{q}) \left[Q_{\text{GPD}}(\mathbf{x}) - \lambda_1 Q_{\text{COM}}(\mathbf{q}) - \lambda_2 Q_{\text{lim}}(\mathbf{q}) \right], \tag{8}$$

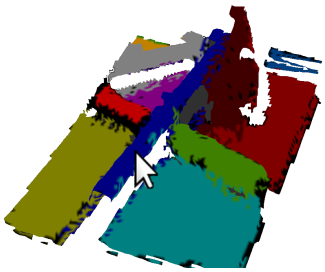
(a) Overview over a scene composed of multiple stacked pipes and a traffic cone.



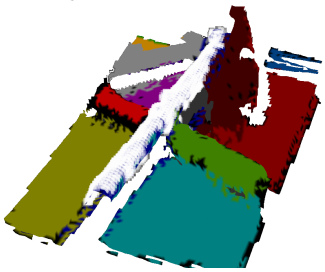
(b) As a first step, the center of the area of interest for the exploration process is marked by moving an interactive marker.



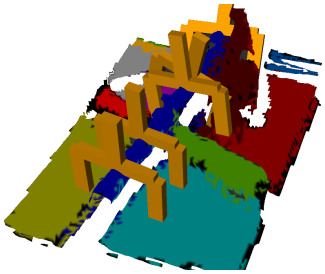
(c) The resulting segmented scene after the completed multi-view exploration.



(d) The operator can select the object to be grasped by clicking on a segment.



(e) The selected object is meshed and used for grasp pose detection.



(f) Valid grasp candidates for the selected segment.

Fig. 3: The process of grasp object selection is controlled via two operator inputs, one to mark the area of interest using the lidar data and one to select the object to grasp.

which is used to rank the grasp candidates. Here, λ_1 is the relative weight for Q_{COM} and λ_2 is the relative weight for Q_{lim}. By factoring out $Q_{\text{sing}}(\mathbf{q})$, grasps near singularities are circumvented.

F. Grasp Execution

Finally, for each ranked and valid grasp candidate, we compute the corresponding approach vector. In descending ranking score order, a collision-free trajectory is planned for each grasp candidate using *Moveit!* [16]. The planned trajectory moves the gripper into the pre-grasp pose, approaches the object along the approach vector and then grasps the object.

IV. RESULTS

In this section, we present the resulting grasping procedure including the operator interface, compare category-agnostic instance segmentation performance with two baseline methods on RGB-D images and test the proposed method for grasping in clutter.

A. Grasping Procedure & Operator Interface

In the following, we present the procedure for grasping an arbitrarily shaped object using an exemplary scene (Figure 3a) in detail. As shown in Figure 3b, the robot operator can start the grasping procedure by marking the center of the area of interest with a 3D interactive marker. The robot then explores the area by moving the arm into several view poses that point the RGB-D camera at the center of the area of interest. Using the view poses, the 3D ISSM (see Figure 3c) is created. The segmented scene representation augments the scene understanding of the robot operator obtained by the image of the RGB camera. Eventually, the robot operator can select the object which should be grasped by selecting the segment in the user interface (see Figure 3d). The selected segment is highlighted as shown in Figure 3e and passed to the grasp pose detection which returns a set of antipodal grasps. Subsequently, the best grasp of all valid grasps (see Figure 3f) is selected for the grasp execution using the grasp quality metrics.

B. Segmentation Comparison

In the following, we compare the geometric segmentation approach proposed in Section III-B.1 with two alternative segmentation approaches on depth images produced by an Intel D435 RGB-D camera. The first alternative segmentation approach is a region growing approach [17] that operates on a point cloud and is implemented in the Point Cloud Library [19]. The algorithm clusters points by checking a smoothness constraint that compares the angles between the normals of the points. The second alternative approach is the SD Mask R-CNN segmentation [18], a neural network trained for category-agnostic instance segmentation. The segmentation quality of the approaches is evaluated and compared using four example scenes shown in Figure 4. As can be seen, the segmentation quality of the proposed approach is comparable to the region growing approach but shows better robustness to noise in the last scene. Yet, the approach tends to over-segment objects, for example, the red rope in the last scene. However, the resulting segmentation quality is sufficient to command the grasp pipeline accurately.

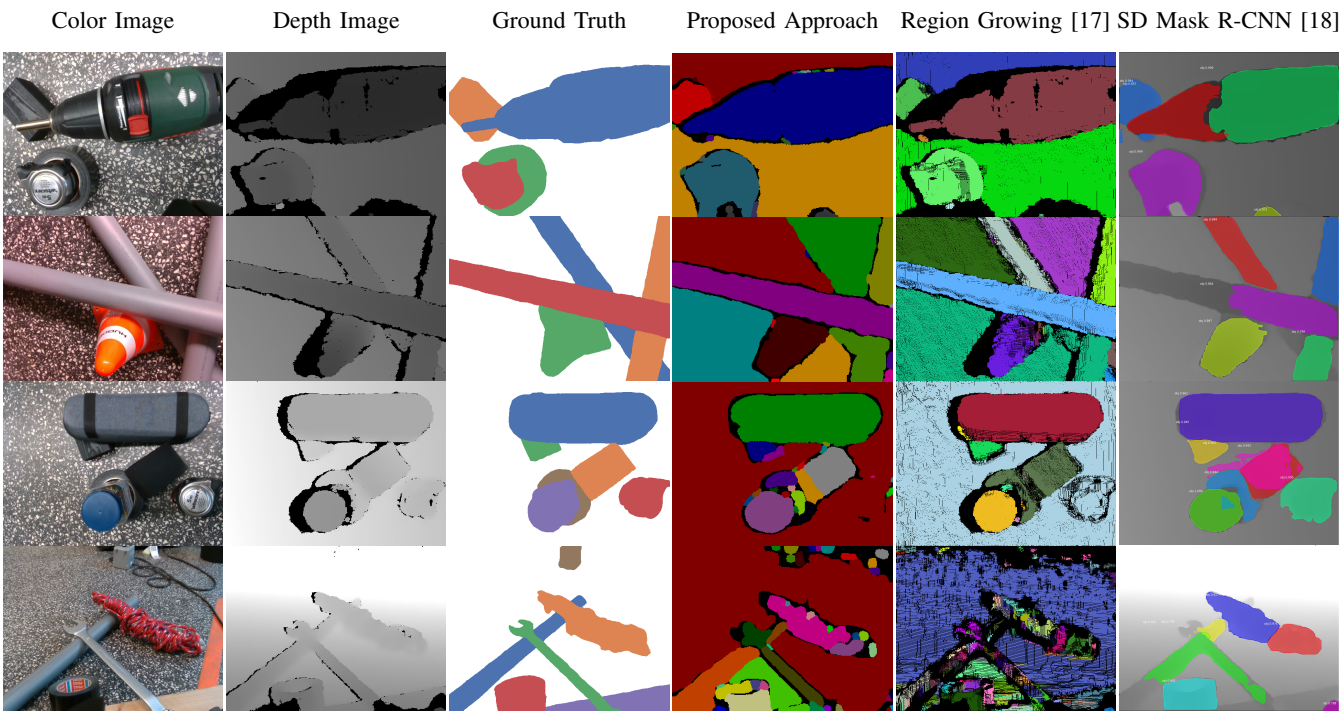


Fig. 4: Exemplary results generated by the compared segmentation approaches for four different scenes.

Figure 5 depicts the runtimes of the different segmentation approaches divided into preprocessing time and segmentation time which were measured using an Intel Core i7-6700HQ mobile processor. The preprocessing of the proposed segmentation algorithm consists of subsampling the image by a factor of 2 followed by inpainting the missing depth values and a cross-product based normal estimation. For the region growing approach, the preprocessing needed is a normal estimation that fits a plane in the local neighborhood of each point. As SD Mask R-CNN was trained on inpainted depth images, we inpaint each depth image.

In summary, the proposed segmentation performs well in comparison with the two other approaches while being more than one order of magnitude faster. However, SD Mask R-CNN is able to segment some of the more complex scenes, for example, the jar standing on the adhesive tape roll in the third scene even without retraining the neural network for our specific RGB-D camera including its noise characteristics. Hence, retraining the neural network on our own data might yield even better segmentation performance.

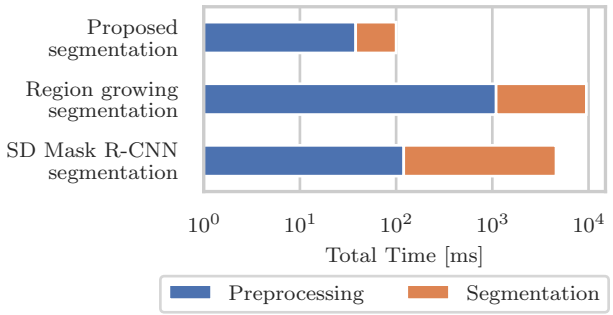


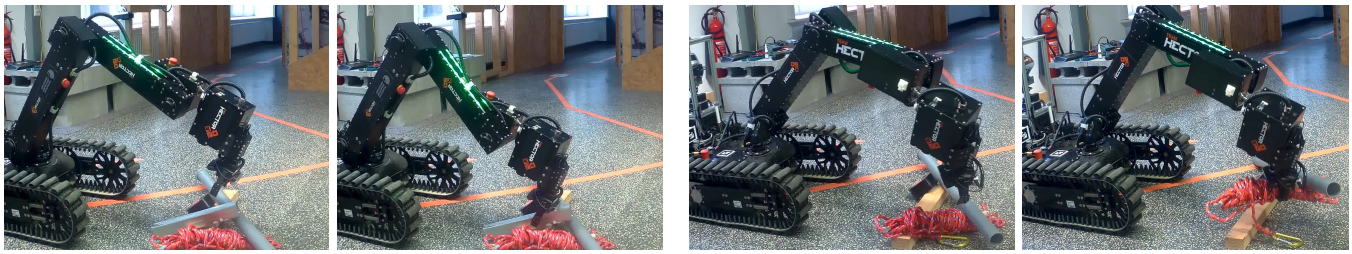
Fig. 5: Average runtimes for the different segmentation approaches composed of the preprocessing of input images with size 640×480 and the segmentation itself averaged over four images.

C. Grasp Experiments

For testing and evaluating grasping in clutter, the scene shown in Figure 6 consisting of a pipe laying on a rope and a wooden block, an adhesive tape roll, and a metal carrier is used. The experimental platform is equipped with a 6 DOF manipulator arm and has an Intel D435 RGB-D camera mounted on the wrist. First, the operator selects the metal carrier to be grasped. The robot approaches the object from above and safely extracts the object. Subsequently, the operator commands to extract the rope. Again, the planned grasp is successful after an additional object exploration. The segmentation quality is sufficient for the extraction of the objects, yet the grasping of the tape roll is not possible due to an oversegmentation of the object. Additionally, the TSDF reconstruction of the adhesive tape roll is deformed despite the usage of a voxel resolution of $5\,\mathrm{mm}$. This is caused by a noisy depth image of the RGB-D camera, most likely because of light reflections due to a sub-optimal viewing angle. Thus, for the reconstruction and grasping of small objects and objects with very noisy sensor data, the approach is not perfectly suited at this point.

V. DISCUSSION AND FUTURE WORK

In this paper, a method has been presented that enables grasping of unknown objects while maintaining the possibil-



(a) Grasping of a metal carrier stacked on top of the clutter.

(b) Grasping of the red rope laying below the pipe.

Fig. 6: Grasping in clutter.

ity to select the object to grasp. By avoiding only pre-defined object classes, we are able to grasp arbitrary rigid objects. Thanks to the efficient interface requiring no training, the method is usable for emergency responders as well. By detecting antipodal grasps that are ranked using additional metrics, we aim for robust and stable grasps using a two-fingered hand. The approach is suitable for all robots that are able to approach a 6D grasp pose, that have a depth camera, ideally mounted on the manipulator arm, and that either have a two-fingered gripper or approximate it. There are several ways in which the system could be extended. Currently, the geometric segmentation approach – despite being fast – tends to oversegment objects in the scene. By improving the category-agnostic segmentation approach, for example by incorporating a deep learning-based approach, the overall performance of the method could be improved even further. In the current state, the object exploration procedure generates random view poses for the manipulator arm. Instead, planning the view poses such that the area of interest is explored with the minimal number of poses could reduce the time needed for object exploration. Additionally, executing the grasps with closed-loop control could compensate for a slightly decalibrated arm as well as small localization errors and therefore could increase the robustness of the presented approach.

VI. ACKNOWLEDGMENT

Research presented in this paper has been supported in parts by the German Federal Ministry of Education and Research (BMBF) within the subproject "Autonomous Assistance Functions for Ground Robots" of the collaborative A-DRZ project (grant no. 13N14861).
The authors gratefully acknowledge the contributions by and fruitful cooperation with all members of Team Hector.

REFERENCES

[1] B. Brüggemann, B. Gaspers, A. Ciossek, *et al.*, "Comparison of Different Control Methods for Mobile Manipulation using Standardized Tests," in *2013 IEEE Int. Symposium on Safety, Security, and Rescue Robotics (SSRR)*, IEEE, 2013, pp. 1–2.

[2] D. A. Klein, B. Illing, B. Gaspers, *et al.*, "Hierarchical Salient Object Detection for Assisted Grasping," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 2230–2237.

[3] A. Romay, S. Kohlbrecher, and O. von Stryk, "An Object Template Approach to Manipulation for Humanoid Avatar Robots for Rescue Tasks," *KI-Künstliche Intelligenz*, vol. 30, no. 3-4, pp. 279–287, 2016.

[4] T. Klamt, D. Rodriguez, M. Schwarz, *et al.*, "Supervised Autonomous Locomotion and Manipulation for Disaster Response with a Centaur-Like Robot," in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 5483–5490.

[5] D. De Gregorio, F. Tombari, and L. Di Stefano, "RobotFusion: Grasping with a Robotic Manipulator via Multi-view Reconstruction," in *European Conf. on Computer Vision*, Springer, 2016, pp. 634–647.

[6] G. Vezzani, U. Pattacini, and L. Natale, "A Grasping Approach Based on Superquadric Models," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 1579–1586.

[7] D. J. Butler, S. Elliot, and M. Cakmak, "Interactive Scene Segmentation for Efficient Human-In-The-Loop Robot Manipulation," in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, IEEE, 2017, pp. 2572–2579.

[8] A. ten Pas, M. Gualtieri, K. Saenko, *et al.*, "Grasp Pose Detection in Point Clouds," *The Int. Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017.

[9] K. Tateno, F. Tombari, and N. Navab, "When 2.5D is not enough: Simultaneous Reconstruction, Segmentation and Recognition on dense SLAM," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 2295–2302.

[10] M. Rünz and L. Agapito, "MaskFusion: Real-Time Recognition, Tracking and Reconstruction of Multiple Moving Objects," in *IEEE Int. Symposium on Mixed and Augmented Reality (ISMAR)*, IEEE, 2018.

[11] A. Ückermann, C. Elbrechter, R. Haschke, *et al.*, "3D Scene Segmentation for Autonomous Robot Grasping," in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, IEEE, 2012, pp. 1734–1740.

[12] A. Ückermann, R. Haschke, and H. Ritter, "Real-Time 3D Segmentation of Cluttered Scenes for Robot Grasping," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS Int. Conf. on*, IEEE, 2012, pp. 198–203.

[13] I.-M. Chen and J. W. Burdick, "Finding Antipodal Point Grasps on Irregularly Shaped Objects," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 4, pp. 507–512, 1993.

[14] A. Liegeois, "Automatic Supervisory Control of the Configuration and Behaviour of Multibody Mechanisms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 7, no. 12, pp. 868–871, 1977.

[15] G. Marani, J. Kim, J. Yuh, *et al.*, "A real-time approach for singularity avoidance in Resolved Motion Rate Control of Robotic Manipulators," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, IEEE, vol. 2, 2002, pp. 1973–1978.

[16] S. Chitta, I. Sucan, and S. Cousins, "MoveIt! [ROS Topics]," *IEEE Robotics Automation Magazine*, vol. 19, no. 1, pp. 18–19, Mar. 2012, ISSN: 1070-9932. DOI: 10.1109/MRA.2011.2181749.

[17] T. Rabbani, F. van den Heuvel, G. Vosselman, *et al.*, "Segmentation of point clouds using smoothness constraints," in *Int. Society for Photogrammetry and Remote Sensing (ISPRS)*, vol. 35, 2006, pp. 248–253.

[18] M. Danielczuk, M. Matl, S. Gupta, *et al.*, "Segmenting Unknown 3D Objects from Real Depth Images using Mask R-CNN Trained on Synthetic Point Clouds," *arXiv preprint arXiv:1809.05825*, 2018.

[19] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, IEEE, 2011, pp. 1–4.