RoboCup Rescue 2017 Team Description Paper Hector Darmstadt

Stefan Kohlbrecher, Christian Rose, Dorothea Koert, Paul Manns, Kevin Daun, Alexander Stumpf and Oskar von Stryk

Info

Team Name:	Hector Darmstadt
Team Institution:	Technische Universität Darmstadt
Team Leader:	Stefan Kohlbrecher
Team URL:	

http://www.sim.informatik.tu-darmstadt.de/teamhector/

RoboCup Rescue 2017 TDP collection:
https://to-be-announced.org

***Abstract*—This paper describes the approach used by Team Hector Darmstadt for participation in the 2016 RoboCup Rescue Robot League competition. Participating in the RoboCup Rescue competition since 2009, the members of Team Hector Darmstadt focus on exploration of disaster sites using autonomous Unmanned Ground Vehicles (UGVs).**

We provide an overview of the complete system used to solve the problem of reliably finding victims in harsh USAR environments. This includes hardware as well as software solutions and diverse topics like locomotion, SLAM, pose estimation, human robot interaction and victim detection. In 2016, the team focuses on improving 3D perception, autonomous rough terrain locomotion and manipulation capabilities. As a contribution to the RoboCup Rescue community, major parts of the used software have been released and documented as open source software for ROS.

***Index Terms*—RoboCup Rescue, Team Description Paper, Simultaneous Localization and Mapping, Manipulation, Urban Search and Rescue.**

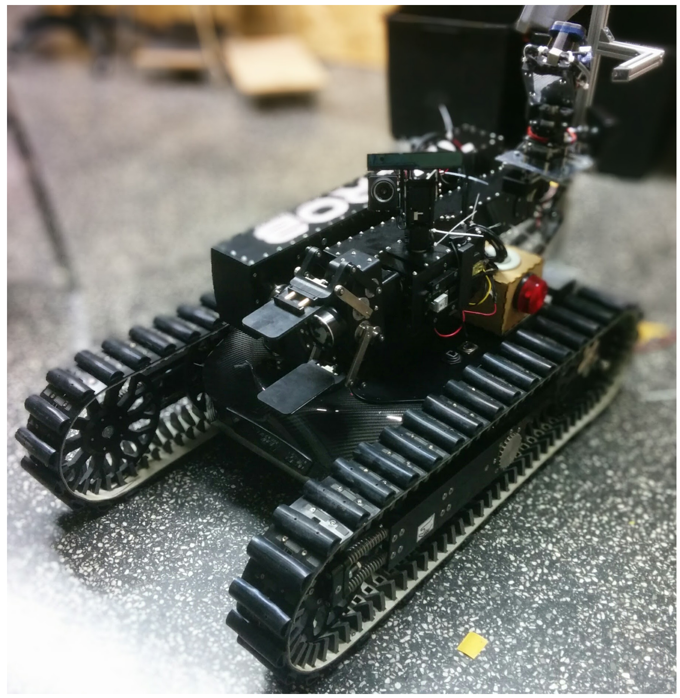


Fig. 1: The Taurob Tracker platform in the "Hector Tracker" configuration with additionals sensors for autonomy added, such as a spinning Hokuyo LIDAR mounted on a slip ring.

I. Introduction

Team Hector Darmstadt (Heterogeneous Cooperating Team of Robots) has been established in late 2008 within the Research Training Group GRK 1362 "Cooperative, Adaptive and Responsive Monitoring in Mixed Mode Environments" (http://www.gkmm.tu-darmstadt.de) funded by the German Research Foundation (DFG). Driven by the goal of using heterogeneous hardware and software in disaster environments, a successful participation in RoboCup Rescue is an important milestone towards truly versatile robots for disaster response.

The team participated in RoboCup Rescue 2009 for the first time. Focusing on autonomy for rescue robot system, the team has a history of highly successful participation in the RoboCup Rescue Robot League competition. The "Best in Class Autonomy Award was awarded to the team in the RoboCup German Open competitions from 2011-2015 and in the RoboCup world championships from 2012-2015. The team also demonstrated that approaches leveraging a high level of autonomy are competitive with teleoperated robots. The team scored the first place in the overall scoring at RoboCup German Open 2011-2014. Most notably, it won the world champion title at the RoboCup 2014 competition in Brazil.

Contributing to a initiative within the RoboCup Rescue community to establish an open source framework for USAR robotics [1], the team has released many of the software modules used to achieve top scores at the RoboCup competition as open source software for ROS to facilitate progress and reduce the need for re-inventing the wheel (for instance every team having to develop their own SLAM system from scratch).

Ground robots most frequently used during prior RoboCup competitions are based on the Kyosho Twin Force R/C model. Higher mobility, full 3D perception and manipulation capabilities are required for many real world applications however, so a highly mobile tracked vehicle based on the Taurob Tracker platform as shown in Figure 1 is now used.

Major additions and changes compared to previous years participation are:

- Improved 3D state estimation and SLAM

Robo Hero is with the Department of Computer Science, Technische Universität Darmstadt, Germany, e-mail: .

- Motion planning and control capabilities for traversing rough terrain autonomously or semi-autonomously.
- Adapting approaches for semi-autonomous control and manipulation developed within the scope of Team ViGIR [2] and participation in the DARPA Robotics Challenge for use with tracked USAR robots.

A comprehensive overview of recent research can be found in [3]. An overview of released open source packages for mapping and navigation is available in [4]. A video of the operator station screen at the RoboCup German Open 2014 Rescue Robot League final mission by Team Hector is available online [5]. It shows how the components described throughout this paper enable Best in Class autonomous behavior.

In the following sections, ROS package or stack names written in *italics* like *hector_slam* are available as open source software and can be found on the ROS wiki, e.g. www.ros.org/wiki/hector_slam.

II. System Description

A. Hardware

We briefly describe some key hardware components and sensors in the following.

a) Wheel/Track Encoders: To measure the translational and rotational speed of vehicles, all used vehicles are equipped with encoders measuring wheel or track motion. This odometry data is used for low level speed control. Due to frequent slippage in the considered scenarios, it is not used in the SLAM system.

b) Laser Scanner: The vehicle is equipped with a Hokuyo UTM30-LX LIDAR. It can be mounted on a roll/tilt unit at the front of the autonomy box and is mainly used for 2D mapping. The LIDAR system can be stabilized to stay close to the intended scan plane regardless of vehicle attitude. For better 3D mapping capability, the sensor can be mounted on a slip ring and spin around the vertical axis, allowing the acquisition of full 3D point clouds of the environment.

c) RGB-D Camera: A RGB-D camera is used for environment perception tasks like traversable terrain detection, 3D mapping and also for victim verification. This camera is mounted on the pan/tilt unit that is also used for the camera. We currently use the ASUS Xiton Pro Live sensor, but might exchange this for a smaller solution like the Intel Realsense R200 sensor.

d) Ultrasonic Range Finders: Additionally to the LIDAR, a set of ultrasound range finders mounted at the back of the vehicle enables autonomous reactive collision avoidance when moving backwards, as the LIDAR only covers a 270 degrees field of view.

e) Inertial Measurement Unit: To measure the attitude of the platform, the vehicles are equipped with a 9DOF inertial sensor UM7 by CH Robotics which measures accelerations and angular rates. It also computes a orientation quaternion onboard the sensor.

1) GPS receiver: As the vehicle can optionally be used outdoors too, it can be equipped with a GPS receiver. The position feedback provided by the SLAM system to the map is fused with information from GNSS in this case.

B. Software

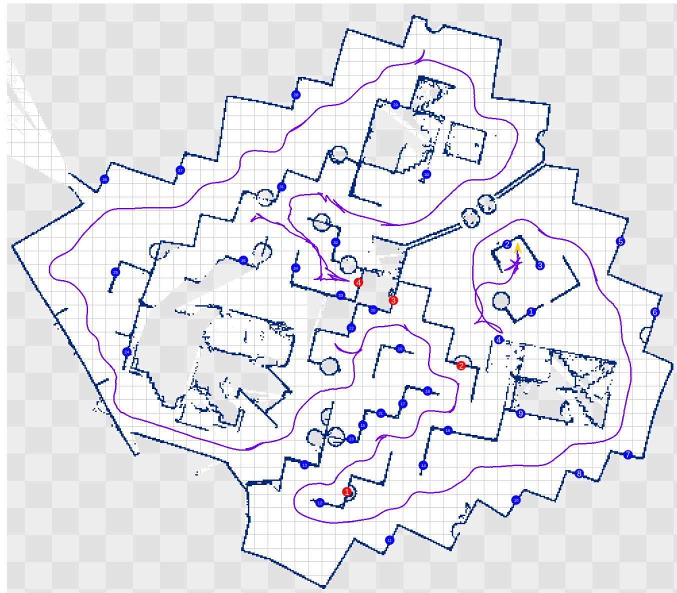


Fig. 2: Map learned using the *hector_slam* system at the final mission of RoboCup 2012. The robot started at the right middle position and autonomously explored the majority of the arena, finding 3 victims (red markers). The fourth victim was found using tele-operation on the last 4 meters of the travelled path. Blue markers indicate the positions of 35 QR codes that were detected autonomously by the robot.

1) SLAM: The Simultaneous Localization And Mapping (SLAM) problem is solved in 2D by using a 2D grid map representation that gets updated using a scan matching approach [6]. The approach has low runtime requirements and can run with an update rate of 40Hz while requiring less than 15% CPU time on a Core 2 Duo setup, freeing resources for other computation. The system does not require odometry data, as the scanmatching approach is very robust. The input used for solving the SLAM problem are laser scans and the robot state as estimated by the navigation filter. Data provided by the navigation filter is used for transformation of laser scans to take into account the attitude of the laser scanner and vehicle during acquisition of scans. Figure 2 shows a map learned using the *hector_slam* system. A video is available online [7].

The map can be manually or automatically annotated with information about victims and other objects of interest. It can be saved in the GeoTIFF format using the *hector_geotiff* package. The described software is available and documented as open source software in the *hector_slam* stack for ROS, which is widely used within the RoboCup Rescue League and beyond.

To enable autonomous cooperative deployment of multiple robots on missions, a feature based map merging system has been developed. Each robot detects ORB features [8] for the estimated map and these are exchanged among teammate robots. A registration approach is then used to estimate a common coordinate frame for all robots. The map stitching approach is available open source via the *hector_mapstitch*

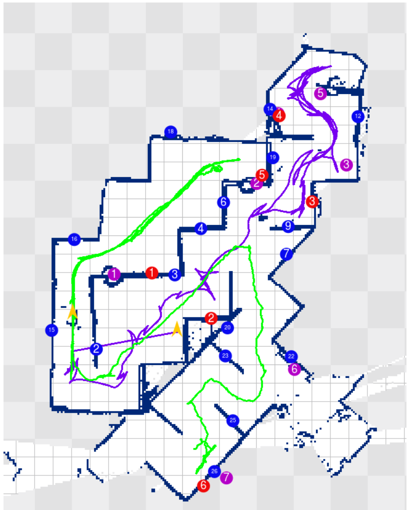


Fig. 3: Fused map from two robots generated during the RoboCup 2015 preliminary round. The map was merged automatically on the operator station computer in real-time. Note that objects detected by both robots are correctly merged as well.

package. Figure 3 shows a merged map as generated during the RoboCup 2015 competition.

In an effort to achieve full 3D state estimation capability, multiple approaches are evaluated. Promising results have been achieved using the *ethzasl_icp_mapping* [9] ROS package with full 3D clouds from a spinning LIDAR sensor. Irregular motion when traversing rough terrain remains a significant challenge and the detection and compensation of slippage using fusion of odometry, visual odometry and IMU data is investigated.

2) Victim Detection: Finding human victims under difficult conditions of unstructured post-disaster environments is one of the main goals of RoboCup Rescue. Significant progress in visual object recognition and scene understanding allows us to apply state of the art computer vision methods. To tackle this problem we use a multi-cue victim detection system supporting optical image cues like RGB, thermal and depth images. This complementary information can be used to increase reliability.

Once the detector has recognized a victim or other object of interest this detection is forwarded to the *hector_object_tracker* which keeps track of known objects and updates this model based on positive and negative evidence. The separation of object detection and modeling enables the flexible integration of different sensory sources for various classes of objects. The position and pose of each object is tracked using a Kalman Filter. The *hector_object_tracker* is the only interface between perception and control, e.g. for the creation or modification of tasks or the manipulation of model state due to operator interaction.

A comprehensive overview of our approach to semantic mapping using heterogenous sensors such as thermal and visual cameras can be found in [10].

a) Thermal- and Depth-Based Victim Detection: In addition to visual victim detection we use a thermal and also a RGB-D camera to verify vision-based hypotheses.

In most cases images provided by the thermal camera are very helpful for identifying possible victim locations. As a drawback of a thermal camera the thermal images often contain not only victims but also other warm objects, such as radiators or fire, so that thermal and visual recognition systems will deliver complementary information.

To further reduce false-positives we use point clouds from the RGB-D camera to evaluate the environment of the victim hypotheses. False-positive victim hypotheses can be identified by the shape of the environment or by missing depth measurements at the victim location.

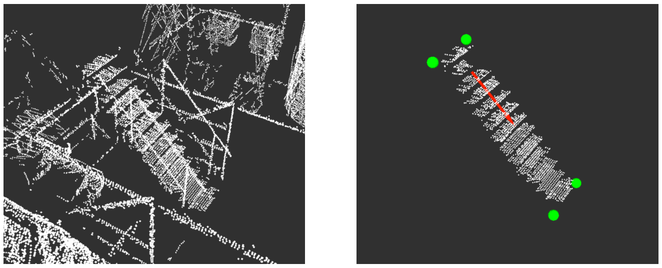
b) QR Code Detection: As a step towards more exhaustive sensor coverage of the environment and future detection of additional objects of interest, QR codes are placed in the RoboCup Rescue arena, with points being awarded for their successful detection and localization in the map. This task is supported by two PS Eye cameras pointing out of the left rear and right rear of the autonomy box. Using the *hector_qrcode_detection* package, QR code detection is by default running for the Kinect RGB-D camera and both PS Eye cameras, for maximum coverage of the robots surrounding. Detections are used as input for the *hector_object_tracker* as described above for the victim detection case.

3) Motion Planning: To better negotiate the increasingly difficult terrain in the rescue arena, a RGB-D camera is mounted on the robot. This allows to acquire point clouds, build a 2.5D height map and classify the terrain into traversable and non-traversable grid cells. A 3D map of the environment is generated using a modified version of the *octomap* mapping package [11]. Raycasting into the 3D map is used to determine distances to objects of interest detected with imaging sensors. The 3D map also serves as the basis for a active gaze control approach that keeps track of observed parts of the environment and can control the gaze (pan/tilt motion) of camera sensors accordingly.

The well established *hector_exploration_planner* is used for risk-aware frontier-based exploration. This exploration approach is highly reliable, but only supports the distinction between traversable and non-traversable terrain. It thus does not allow for the traversal of stairs and other difficult to cross obstacles. To support the autonomous traversal of such obstacles, an approach that additionally leverages 3D sensor data and automated segmentation of the environment has been developed. It can be used to cross difficult obstacles such as stairs autonomously.

C. Communication

A COTS wireless network system is used for high-bandwidth data like video images or map information. Both 2.4 GHz 802.11g/n or 5 GHz 802.11a/n operation are possible. The operator station is connected to a wireless access point to which the robot(s) connect. Integration of robot systems into a mesh network, that can be established even infrastructure is degraded due to a disaster, is a topic of research, but not used within the RoboCup Rescue competition.



(a) (b)

Fig. 4: Stair segmentation: (a) Pointcloud of an environment containing stairs (b) Resulting segmentation with corner points marked green and slope indicated by a red arrow.

D. Human-Robot Interface

Our research focus is on autonomy and human-supported autonomous systems, therefore we do not focus on operator interfaces for teleoperation, but instead concentrate on methods for better cooperation between human operators and robots at a higher level of abstraction.

a) Mission Definition and Control: For defining autonomous or semi-autonomous behaviors control, the FlexBE (Flexible Behavior Engine) approach developed within the scope of the DRC is used [12]. Using FlexBE, basic robot capabilites can be modeled via FlexBE states and complex behaviors can be composed via the FlexBE GUI by drag and drop. FlexBE supports selecting the desired autonomy level of the robot at runtime, making it very well suited for flexible control with an adjustable level of autonomy.

b) Monitoring and Human Supervision: Using ROS as middleware, the *rviz* visualization tool can be used for visualizing maps and other arbitrary data as well as for sending commands to the robots. As a second important tool *rqt* is used, which provides graphical dialogs for publishing and receiving messages, calling services, and visualizing the interactions between all active nodes. Additional plugins can be written in Python or C++ and can be loaded by *rqt*, thus providing an integrated GUI for mission control as well as for debugging. During autonomous and supervised autonomous operation, the FlexBE GUI is used to monitor behavior execution, setting the desired level of autonomy and potentially adjusting the behavior specification on the fly.

c) Teleoperation: In case supervisory control is not sufficient (especially in difficult terrain in the orange and red arena), all vehicles can be fully teleoperated using a gamepad, joystick or the keyboard. In this case the operator uses the aggregated worldmodel generated from sensors onboard the robot and video-streams to obtain situation awareness.

III. Application

A. Set-up and Break-Down

The system consists of one or more robots capable of autonomous or tele-operation via a laptop computer. All of the control equipment easily fits into a standard backpack and depending on the robots used, robots can be carried by hand (wheeled Hector UGV) or should be carried by two persons (tracked vehicles). To start a mission, the robots and the laptop have to be switched on, and the operator can connect to the robots via WiFi.

B. Mission Strategy

As a focus of our research is reducing workload for operators and leveraging synergies between intelligent onboard systems and operators, pure teleoperation is only employed in case of failure of autonomous components. As during previous competition participation, autonomous operation is the desired control modality, possibly switching to a supervised autonomy mode for complex manipulation tasks that benefit from an human operator's superior cognitive and sense-making abilities.

C. Experiments

Robot systems are tested against subsets of standard NIST ASTM standard test methods that are reproduced in our lab. This include a random maze, stairs, simulated victims and some of the proposed manipulation tasks. Importantly, testing of all system software components in simulation is a first class concept that is used to full extent by the team. Using the *gazebo* simulator, robots can be simulated within arbitrary disaster scenarios, allowing to evaluate performance and identify issues before costly and involved tests with the real system are performed. The *hector_nist_arenas_gazebo* ROS package allows the fast and user friendly creation of simulated disaster scenarios using elements of the NIST standard test arenas for response robots.

D. Application in the Field

The Taurob Tracker platform is used by multiple first responder organizations, for instance the fire department of Vienna. The applicability of the platform in teleoperation mode has thus been already demonstrated in the field. Using the software components described above, we will demonstrate improved autonomous capabilities in the future that allow for mapping the environment and reducing the workload of operators.

IV. Conclusion

In this team description paper we provide an outlook towards the RoboCup 2017 competition. Shifting focus from autonomous approaches for exploration and observation towards complex interaction with the environment through manipulation and planning difficult terrain traversal, this year's development are a significant expansion of prior efforts. We plan on revising the contents of this document once the new robot platform equipped with a manipulator arm has been received and experiments performed with it.

Appendix A
Team members and Their Contributions

Many students and researchers at TU Darmstadt contribute to the team. The following list is in alphabetical order:

- Kevin Daun Motion Planning

TABLE I: Manipulation System

Attribute	Value
Name	Tracker
Locomotion	tracked
System Weight	50kg
Weight including transportation case	60kg
Transportation size	1.1 x 0.7 x 0.5 m
Typical operation size	1.0 x 0.6 x 0.4 m
Unpack and assembly time	30 min
Startup time (off to full operation)	5 min
Power consumption (idle/ typical/ max)	60 / 200 / 800 W
Battery endurance (idle/ normal/ heavy load)	300 / 160 / 80 min
Maximum speed (flat/ outdoor/ rubble pile)	2 / 1 / 0.2 m/s
Payload (typical, maximum)	20/ 30 kg
Arm: maximum operation height	140 cm
Arm: payload at full extend	2kg
Support: set of bat. chargers total weight	2.5kg
Support: set of bat. chargers power	1,200W (100-240V AC)
Support: Charge time batteries (80%/ 100%)	90 / 120 min
Support: Additional set of batteries weight	2kg
Cost	50000 USD

TABLE II: Operator Station

Attribute	Value
Name	Opstation
System Weight	3.2kg
Weight including transportation case	4.5kg
Transportation size	0.4 x 0.4 x 0.2 m
Typical operation size	0.4 x 0.4 x 0.4 m
Unpack and assembly time	1 min
Startup time (off to full operation)	1 min
Power consumption (idle/ typical/ max)	60 / 80 / 90 W
Battery endurance (idle/ normal/ heavy load)	10 / 5 / 4 h
Cost	2000 USD

- Gabriel Hüttenberger — Behavior Controls
- Dorothea Koert — Active Gaze Control, Map Merging
- Stefan Kohlbrecher — Team Lead, SLAM
- Paul Manns — Motion Control
- Christian Rose — Mechatronics
- Elisa Strauch — Behavior Control
- Alexander Stumpf — Planning

Appendix B
Lists

An overview of the used hard- and software is provided in the Tables I, II, III and IV.

TABLE III: Hardware Components List

Part	Brand & Model	Unit Price	Num.
Tracker robot	Taurob Gmbh	50000 USD	1
IMU	CH Robotics UM7	150 USD	1
RGB-D Camera	Asus Xtion Pro Live	150 USD	1
Thermal Camera	ThermalEye	3000 USD	1
LRF	Hokuyo UTM-30LX	4000 USD	1
Rugged Operator Laptop	Schenker W503	2000 USD	1

TABLE IV: Software List

Name	Version	License	Usage
Ubuntu	14.04	open	OS
ROS	jade	BSD	Middleware
PCL [13]	1.7	BSD	ICP
OpenCV [14], [15]	2.4.8	BSD	Victim, QR, barrel detection
Hector SLAM [16]	0.3.4	BSD	2D SLAM

References

[1] S. Kohlbrecher, K. Petersen, G. Steinbauer, J. Maurer, P. Lepej, S. Uran, R. Ventura, C. Dornhege, A. Hertle, R. Sheh, , and J. Pellenz, "Community-driven development of standard software modules for search and rescue robots," in *Proceedings of the 10th IEEE International Symposium on Safety Security and Rescue Robotics (SSRR 2012)*, 2012.

[2] S. Kohlbrecher, D. C. Conner, A. Romay, F. Bacim, D. A. Bowman, and O. von Stryk, "Overview of team vigir's approach to the virtual robotics challenge," in *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 1–2.

[3] S. Kohlbrecher, F. Kunz, D. Koert, C. Rose, P. Manns, K. Daun, J. Schubert, A. Stumpf, and O. von Stryk, "Towards highly reliable autonomy for urban search and rescue robots," in *Proc. RoboCup Symposium 2014*, ser. Lecture Notes in Artificial Intelligence (LNAI). Springer, 2014.

[4] S. Kohlbrecher, J. Meyer, T. Graber, K. Petersen, O. von Stryk, U. Klingauf, K. Petersen, A. Kleiner, O. von Stryk, S. Kohlbrecher *et al.*, "Hector open source modules for autonomous mapping and navigation with rescue robots," *Proc. RoboCup Symposium 2013*, pp. 251–260, 2013.

[5] (2014) Operator station recording of final mission at RoboCup German Open 2014 RRL final mission on YouTube. https://www.youtube.com/watch?v=Q_ChNVIsmeE.

[6] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," in *International Symposium on Safety, Security, and Rescue Robotics*. IEEE, November 2011.

[7] (2010) Video of Hector UGV during Best in Class Autonomy Challenge on YouTube. http://www.youtube.com/watch?v=nI1DWboC73w.

[8] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: an efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2564–2571.

[9] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing icp variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, April 2013.

[10] J. Meyer, P. Schnitzspan, S. Kohlbrecher, K. Petersen, O. Schwahn, M. Andriluka, U. Klingauf, S. Roth, B. Schiele, and O. von Stryk, "A Semantic World Model for Urban Search and Rescue Based on Heterogeneous Sensors," in *RoboCup 2010: Robot Soccer World Cup XIV*, ser. Lecture Notes in Computer Science, 2011, pp. 180–193.

[11] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013, software available at http://octomap.github.com. [Online]. Available: http://octomap.github.com

[12] P. Schillinger, S. Kohlbrecher, and O. von Stryk, "Human-robot collaborative high-level control with application to rescue robotics," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016, p. to appear.

[13] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[14] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, 2001, pp. I–511–I–518 vol.1.

[15] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 1, 2002, pp. I–900–I–903 vol.1.

[16] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," in *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE, November 2011.