RoboCup Rescue 2019 Team Description Paper Hector Darmstadt

Stefan Kohlbrecher, Marius Schnaubelt, Gabriel Huettenberger, Martin Oehler, Kevin Daun, Stefan Fabian, Karim Barth, Tobias Ullrich, Katrin Becker and Tom Linnemann

Info

Team Name:	Hector Darmstadt
Team Institution:	Technische Universität Darmstadt
Team Leader:	Martin Oehler
Team URL:	http://www.teamhector.de/

RoboCup Rescue 2019 TDP collection:
https://robocup-rescue.github.io/team_description_papers/

***Abstract*—This paper describes the approach used by Team Hector Darmstadt for participation in the 2019 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 2019, the team focuses on highly reliable and accurate 3D Mapping and autonomous rough terrain locomotion. As a contribution to the RoboCup Rescue community, many 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 Jasmine UGV pictured here at the RoboCup 2018 competition.

I. Introduction

Team Hector Darmstadt (Heterogeneous Cooperating Team of Robots) has been established in late 2008. 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 2018 and in the RoboCup world championships from 2012-2015 and in 2018. 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.

Many members of the team participated in the TOTAL ARGOS Challenge as part of Team ARGONAUTS. Here, many software technologies proven in the RoboCup Rescue Robot League were adapted for use in industrial inspection [1], with Team ARGONAUTS ultimately winning the ARGOS Challenge in 2017. Another success was achieved by winning the World Robot Summit Plant Disaster Prevention Challenge in 2018.

Contributing to an initiative within the RoboCup Rescue community to establish an open source framework for USAR robotics [2], 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 [3].

II. System Description

A. Hardware

Shown in Figure 1, the robot platform we use for the RoboCup 2019 competition is based on a robot platform originally designed by Team Stabilize [4] that has been updated to carry a highly advanced autonomy payload. We briefly describe some key hardware components and sensors in the following.

a) Manipulator Arm: The robot is able to perform basic manipulation tasks using the 5-DOF manipulator arm,

Fig. 2: The Jasmine robot performing a manipulation task.

mounted at the back of the robot. It is designed to be compactly foldable. The flexible gripper with two fingers adapts to the geometry of objects. A camera is mounted inside the gripper for precise control. Figure 2 shows Jasmine during a manipulation task at the RoboCup 2018 Finals.

b) Wheel/Track Encoders: To measure the translational and rotational speed of the vehicle, it is equipped with encoders measuring track motion. This odometry data is used for low level speed control.

c) LIDAR: The vehicle is equipped with a Velodyne VLP-16 Lidar attached to a continuously spinning mount. Using this setup, nearly complete coverage of all directions with highly accurate point cloud data is achieved.

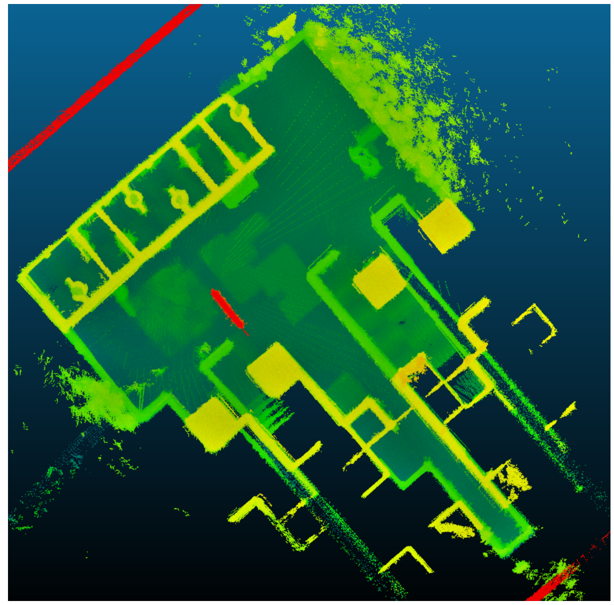
d) Thermal Camera: A Thermaleye thermal camera mounted on a pan/tilt unit is used for victim and heat source detection.

e) RGB-D Camera: An Intel Realsense D435 RGB-D camera is used for object of interest and victim verification. This camera is mounted on the same pan/tilt unit as the thermal camera.

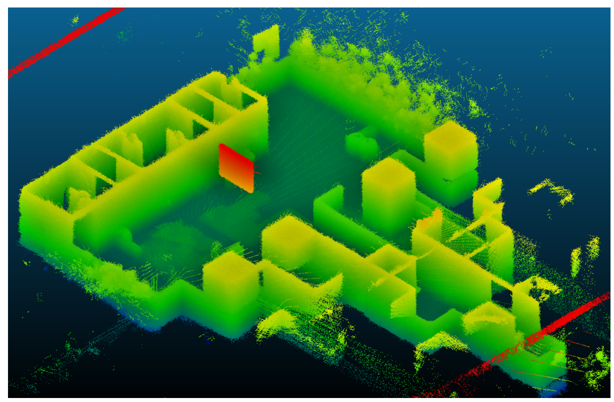
f) 360 Degree Camera: An Insta360 Air camera is mounted on top of the Lidar cage and is used to acquire visual information from all directions. The robot operator uses a virtual pinhole projection to navigate the robot. Additionally, a 360 equi-rectangular projection is used for situational awareness. During detection tasks, the 360-image is used to recognize objects in all directions simultaneously without the need to move a camera.

g) Inertial Measurement Unit: To measure the attitude of the platform, the vehicle is equipped with a 9-DOF inertial sensor which measures accelerations and angular rates an estimates the orientation of 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.



(a)



(b)

Fig. 3: 3D SLAM: (a) Top-down view of 3D map generated during the Best in Class Autonomy Final missions at RoboCup German Open 2018 (b) Perspective view of the same map. Note geometry higher than 3m has been cut from the visualization as otherwise the ceiling geometry of the venue would obstruct the view.

B. Software

1) SLAM: The Simultaneous Localization And Mapping (SLAM) problem is solved in 3D by using a modified variant of Google Cartographer [5] optimized for use with spinning LIDAR data and rough terrain locomotion. Figure 3 shows 3D map output.

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. This package is available and documented as open source software as part of the *hector_slam* stack for ROS, which is widely used within the RoboCup Rescue League and beyond.

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 heterogeneous sensors such as thermal and visual cameras can be found in [6].

a) Thermal- and Depth-Based Victim Detection: In addition to visual victim detection we use a thermal and also an 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. Using the *hector_qrcode_detection* package, QR code detection is by default running for the RGB-D camera and the 360 camera, 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, the 3D spinning Lidar data is used. 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 [7]. Ray-casting 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 an 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 *grid_map_navigation_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.

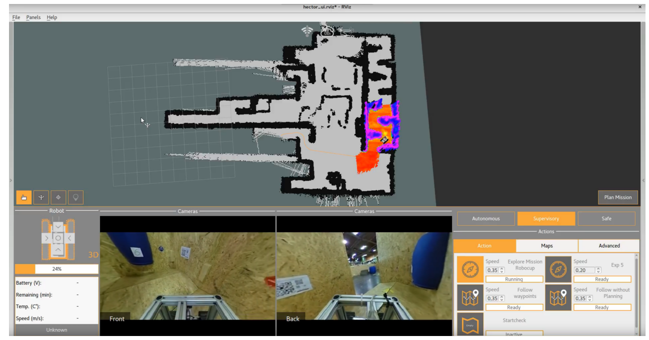


Fig. 4: New User Interface: Scene from the RoboCup German Open 2018 Best in Class Autonomy Final.

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 if infrastructure is degraded due to a disaster is a topic of research [8], but not used within the RoboCup Rescue competition.

D. Human-Robot Interface

To enable seamless sliding autonomy control from pure teleoperation to full autonomy, an advanced user interface on top of existing ROS tools has been developed.

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 [9]. 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: A new user interface on top of *rviz* has been developed that allows for controlling the robot in a unified fashion with all control and visualization options available to the operator in a single user interface window. Figure 4 shows a screen capture of this UI.

c) Teleoperation: In case supervisory control is not sufficient, the robot 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 via the user interface.

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

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 2019 competition. We focus on highly reliable 3D mapping, autonomous rough terrain negotiation and automated perception of objects of interest. These capabilities have been already demonstrated successfully at the RoboCup German Open 2018 and RoboCup 2018 competitions and will be further improved for participation in RoboCup 2019.

TABLE I: Manipulation System

Attribute	Value
Name	Jasmine UGV
Locomotion	tracked
System Weight	25kg
Weight including transportation case	30kg
Transportation size	0.7 x 0.5 x 0.7 m
Typical operation size	0.6 x 0.42 x 0.6 m
Unpack and assembly time	30 min
Startup time (off to full operation)	1 min
Power consumption (idle/ typical/ max)	30 / 100 / 300 W
Battery endurance (idle/ normal/ heavy load)	90 / 60 / 40 min
Maximum speed (flat/ outdoor/ rubble pile)	0.6 / 0.4 / 0.2 m/s
Payload (typical, maximum)	10/ 20 kg
Arm: maximum operation height	0.7m
Arm: payload at full extend	0.3 kg
Support: set of bat. chargers total weight	1.0kg
Support: set of bat. chargers power	1,200W (100-240V AC)
Support: Charge time batteries (80%/ 100%)	30 / 40 min
Support: Additional set of batteries weight	2kg
Cost	4000 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

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:

- Karim Barth — User Interface
- Katrin Becker — Motion Control
- Kevin Daun — 3D SLAM
- Stefan Fabian — Perception, User Interface
- Gabriel Hüttenberger — Behavior Control
- Stefan Kohlbrecher — System Design, SLAM, Navigation
- Tom Linnemann — Mechanical Design
- Martin Oehler — Perception, Calibration, Motion Planning
- Marius Schnaubelt — Mechanical Design, 3D Perception
- Tobias Ullrich — Mechanical Design, Sensor Calibration

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.
Jasmine robot chassis	Team Stabilize	4000 USD	1
IMU	Microstrain 3DM-GX3-45	3000 USD	1
RGB-D Camera	Intel Realsense D435	180 USD	1
Thermal Camera	ThermalEye	3000 USD	1
LRF	Velodyne VLP-16	8000 USD	1
Rugged Operator Laptop	Schenker W503	2000 USD	1

TABLE IV: Software List

Name	Version	License	Usage
Ubuntu	16.04	open	OS
ROS	kinetic	BSD	Middleware
PCL [10]	1.7	BSD	ICP
OpenCV [11], [12]	3.3	BSD	Victim, QR, barrel detection

REFERENCES

[1] S. Kohlbrecher and O. Von Stryk, "From robocup rescue to supervised autonomous mobile robots for remote inspection of industrial plants," *KI-Künstliche Intelligenz*, vol. 30, no. 3-4, pp. 311–314, 2016.

[2] 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.

[3] S. Kohlbrecher, J. Meyer, T. Graber, K. Petersen, O. von Stryk, and U. Klingauf, "Hector open source modules for autonomous mapping and navigation with rescue robots," in *RoboCup 2013: Robot World Cup XVII*, ser. Lecture Notes in Computer Science, vol. 8371. Springer, 2014, pp. 624–631.

[4] W. Chaikanta, W. Chaiwong, R. Saechin, P. Kamnuanek, W. Pinit, N. Jansa, C. Jearanaibanyong, C. Jearanaibanyong, J. Chimphimol, N. Tumbandalnsuk, K. Pueknoun, T. Wattakeekamthorn, K. Wattakeekamthorn, and C. Wongkhorsub, "Robocuprescue 2013-robot league team stabilize (thailand)," 2013.

[5] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271–1278.

[6] 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.

[7] 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

[8] L. Baumgrtner, S. Kohlbrecher, J. Euler, T. Ritter, M. Schmittner, C. Meurisch, M. Mhlhuser, M. Hollick, O. von Stryk, and B. Freisleben, "Emergency communication in challenged environments via unmanned ground and aerial vehicles," in *Proc. IEEE Global Humanitarian Technology Conference (GHTC)*, San Jose, CA, USA, 19-22 Oct. 2017.

[9] 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.

[10] 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.

[11] 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.

[12] 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.