RoboHub Eindhoven: Robocup@Work Team Description Paper 2021

Sjriek Alers, Pablo Negrete Rubio, Ronald Scheer,
Ruben Arts, Marian Lepadatu, Bjarne Wildschut, Jules van Horen
Remco Kuijpers, Mike van Lieshout, Jeroen Bongers, Mark Geraets

RoboHub Eindhoven
Fontys University of Applied Sciences
Department of Engineering - Mechatronics
De Rondom 1, 5612AP Eindhoven, The Netherlands
`info@robohub-eindhoven.nl`, http://robohub-eindhoven.nl

Abstract. This paper introduces the RoboHub team submission to the RoboCup@Work World Championship 2021. This papers details the current state of our robot and the team behind the robot. We give an overview of the hardware platform, the software framework and technical challenges in navigation, object recognition and manipulation.

1 Introduction

RoboHub Eindhoven is a team of motivated students, teachers and professionals that are working together to discover new solutions for robotic systems. RoboHub Eindhoven is part of the Engineering department of the Fontys University of Applied Sciences. We are challenging ourselves to find creative ways to bring robotics to the next step. Together we want to share our knowledge with other motivated people, therefore we work with companies that want to support us and our technology. Within the RoboHub Eindhoven we are participating at the RoboCup@Work league with the RoboHub team that focuses on the industrial use of autonomous robots. We have a multi-disciplinary team of students working on the robot, where the core team is complemented with students from our Adaptive Robotics Minor and guided by experienced RoboCup coaches [1]. Furthermore, we created an educational outreach project around our robot. With our educational outreach project we expect to inspire non-technical students to embrace technology and start learning to create and program robots and how to work with robots. The team already started in 2016 and competed multiple times in the RoboCup German Open. At first we participated using a KUKA YouBot[1], but due to the fact that the YouBot was discontinued we started to investigate the usage of another platform. In 2018 we competed with a prototype of the Probotics Packman platform [2] equipped with a UR3 manipulator [3]. From

[1] http://youbot-store.com
[2] https://probotics-agv.eu
[3] https://www.universal-robots.com

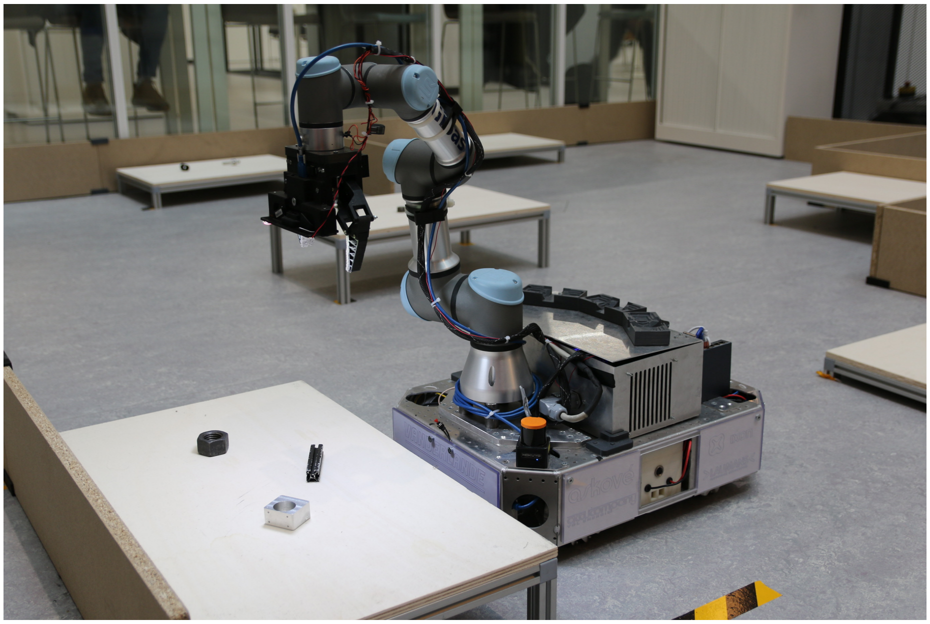

Fig. 1. the *Sui^2* robot in action.

2019 onwards, we compete with our custom designed robot platform *Sui^2*, which is also equipped with an UR3 manipulator.

2 Description of the hardware

Our custom designed robot is named *Sui^2*, the name is derived from 'sui iuris', which is Latin for autonomy. The frame of *Sui^2* is a sheet-metal design made of 3mm aluminum that has been bent and welded to form a rigid body. The frame has everywhere rounded corners, which gives the robot a softer appearance. Large holes in the frame will save a lot of weight while maintaining strength, where the small holes reduce the weight further and can be used for attaching electronics or mechanical parts. The corners are opened on the top to allow lidars to be placed there, in addition the platform is equipped with 2D and 3D camera's. A battery pack can easily be exchanged through an opening at the side of the robot. The bottom plate offers room for the motors and a suspension system. A suspension is needed to ensure that all four wheels stay on the ground at all times, and that the platform does not tilt when the center of gravity changes. The drive-train of the robot has four 180-watt brushless DC motors with PID control on the velocity, and the robot can reach speeds up to 1.7m/s. The motors are equipped with encoders and attached to mecanum wheels to ensure an omni-directional drive. On top of the robot platform an UR3 robot arm is mounted to perform the manipulation tasks. The final design of *Sui^2* is shown in Figure 1.

The robot is controlled from an Intel NUC with an i7 processor for the high-level control, and an ARM Cortex M7 for the low-level control. To connect to the refbox, an industrial graded IXrouter [4] is added which includes a cloverleaf antenna for good connectivity. Additional low-level controls are being performed to make the robot more reliable, reduce internal communication and ensure faster computationally.

3 Description of the software

Our software implementation includes the high-level and low-level controls, sensor and actuator processing and monitoring software.

High-level Control: For the High-level robot control we make use of the Robot Operating System (ROS). ROS [2] provides many useful tools, hardware abstraction and a message passing system between nodes. Nodes are self contained modules that run independently and communicate which each other over so called topics using a one-to-many subscriber model and the TCP/IP protocol.

Localization and Navigation: For localization, *Sui[2]* uses two Hokuyo lidars. The raw lidar data is processed within ROS to perform robust localization. The Global planner is unchanged, but the local planner is replaced in our system by the Timed Elastic Band planner [5] which locally optimizes the robot's trajectory with respect to trajectory execution time [4]. Furthermore, low level smoothing of the acceleration profiles is implemented to make the robot more controllable.

Perception and Manipulation: For object detection and recognition the robot is equipped with a realsense RGB-D camera. This camera is located on the robot-arm near the end-effector. The images are processed with YOLO [6], a clever neural network for doing object detection in real-time [3]. The model determines what object the camera is seeing in real-time and gives the coordinates of the object, as seen in Figure 2. When using four objects the model needs around two hours of training on our GPU, this approach is fast enough that we are able train for new object the moment we get to the competition. To pick the required object, we tried to implement MoveIT [7] as motion planner, but we found out that the robot sometimes was making strange movements. We decided to replace MoveIT by URScript [8] which is way more stable. With URScript it's possible to send commands directly to the UR3 itself, in this way the control is done at the input and the motion planner inside the UR3 itself is used.

[4] https://www.ixon.cloud
[5] https://wiki.ros.org/teb_local_planner
[6] https://pjreddie.com/darknet/yolo/
[7] https://moveit.ros.org/
[8] https://github.com/ros-industrial/ur_modern_driver

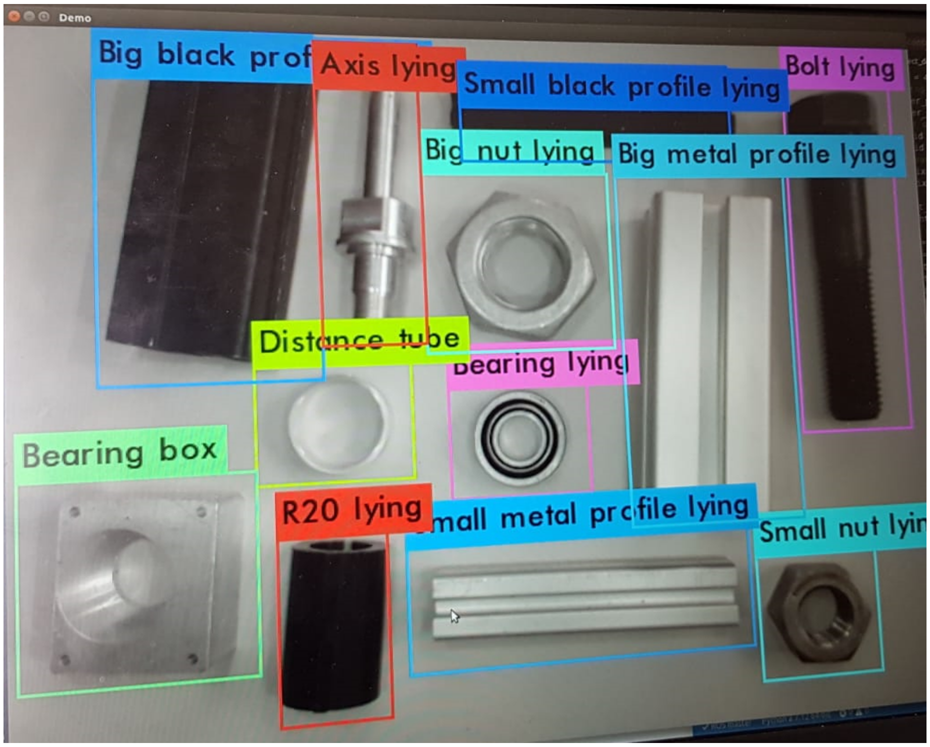

Fig. 2. 2D object detection using Yolo.

Web-Interface: We designed a custom web-interface to set-up, monitor and control the robot in the non-competition phase. This is a web-based interface runs on the robot and is connected to ROS via ros-bridge [9]. With this interface we provide controls to drive the robot, similar as using a game-controller. In addition the map of the robot can be exported and we can use this interface to mark/log specific positions and tweak some parameters of the robot during the set-up phase of the competition.

4 Focus and Relevance

There are a wide range of industrial applications for autonomous mobile manipulation, we focus our education and research mainly on the manufacturing and logistic domains.

Industry: We contribute to the ambitions of the Dutch smart industry [10] agenda. We collaborate with several companies where we use our platform as

[9] https://wiki.ros.org/rosbridge
[10] https://www.smartindustry.nl/english/

a showcase to explain how logistic and manufacturing companies can benefit from mobile robots in their warehouses and factory floors. We are also involved in research project that are funded by the dutch government and are in close collaboration with industry partners. Such a project is the 'Fieldlab Flexible Manufacturing' where mobile manipulating robots contribute to a flexible manufacturing line by delivering parts just on time at the specific (automated) assembly stations, where all these tasks are strongly related to Industry 4.0 [5].

Research: Our current and future research aims at multiple directions, we aim to have adaptive multi-robot systems that are able to autonomously operate in these complex and diverse environments. Think of industrial tasks where multiple robots navigate in a single warehouse and collaboratively transport the required parts in an optimal way, where robots exchange products during transport. We also are focusing on smart/dynamic path-planning, based on the robots experience. Here high and low level traffic rules can not only be pre-programmed, but also should arise on given knowledge of the environment. Furthermore, we work on a set of robot-safety related issues, as on the natural interaction between humans and robots.

Education: First of all we use this competition to motivate and challenge our engineering students to achieve a higher and more professional level in their engineering education. By performing such a project the students get highly motivated, apply their knowledge and push their boundaries to acquire new knowledge to solve the given problems. In addition to the technical research challenges we as a team also focus on getting younger people involved into robotics. Our goal is to show the impact that technology can have in our daily lives. During the last year we visited several events where we give young children (and their parents) the opportunity to control our robot. Furthermore we visited several (primary) schools to give demonstrations to inspire the children.

5 Future work

For this years competition we will focus on making the switch from ROS to ROS2 [11]. In addition to switching to ROS2, we will also be working on implementing some newly created modules into the robot.

- Object localization using 3D PointClouds.
- Barrier tape collision avoidance.
- New end effector capable of sensing failed picks.

[11] http://ros2.org

Online Material: Video's of our platform in action will be available at our YouTube Channel [12]. Additional team information can be found at our RoboHub website [13]. Furthermore, all software will be published on the teams GitHub [14] page.

References

1. Alers, Sjriek, et al. "How to win RoboCup@Work? The Swarmlab@Work approach revealed" Robot Soccer World Cup. Springer, Berlin, Heidelberg, 2013.
2. Quigley, Morgan, et al. "ROS: an open-source Robot Operating System." ICRA workshop on open source software. Vol. 3. No. 3.2. 2009.
3. Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
4. Rösmann, Christoph, Frank Hoffmann, and Torsten Bertram. "Planning of multiple robot trajectories in distinctive topologies." Mobile Robots (ECMR), 2015 European Conference on. IEEE, 2015.
5. Najafi, E., T. Laugs, P. Negrete Rubio, and S. Alers. "An implementation of AutomationML for an Industry 4.0 case study: RoboCup@ Work." In 2019 20th International Conference on Research and Education in Mechatronics (REM), pp. 1-6. IEEE, 2019.
6. Grieves, Michael. "Digital twin: manufacturing excellence through virtual factory replication." White paper 1 (2014): 1-7.

[12] https://www.youtube.com/channel/UCQkpwno0b1QEp96Wy66yLPQ
[13] http://robohub-eindhoven.nl
[14] https://github.com/robohubeindhoven