

Submission for Robothon 2023

Team RoboPig

30 April 2023

Executive Summary

This document is part of the submission for the Robothon 2023 competition, by team RoboPig. We describe the robot platform and all associated software and hardware tools used in the competition, as well as a quick start guide for our setup.

Robot Platform

Our team is using the system shown below, which consists of an off-the-shelf UR5e robot with a Robotiq Hand-E gripper. The robot platform itself has not been modified. For localization we use a static IDS camera mounted above the setup that is surrounded by a ring light. Additionally, image processing during tasks and in the BYOD portion is performed by an Intel Realsense D435i attached to the gripper. To route the probe cable, we designed and 3D-printed a custom hook tool that is placed on the table, allowing the robot to access it whenever it is needed.

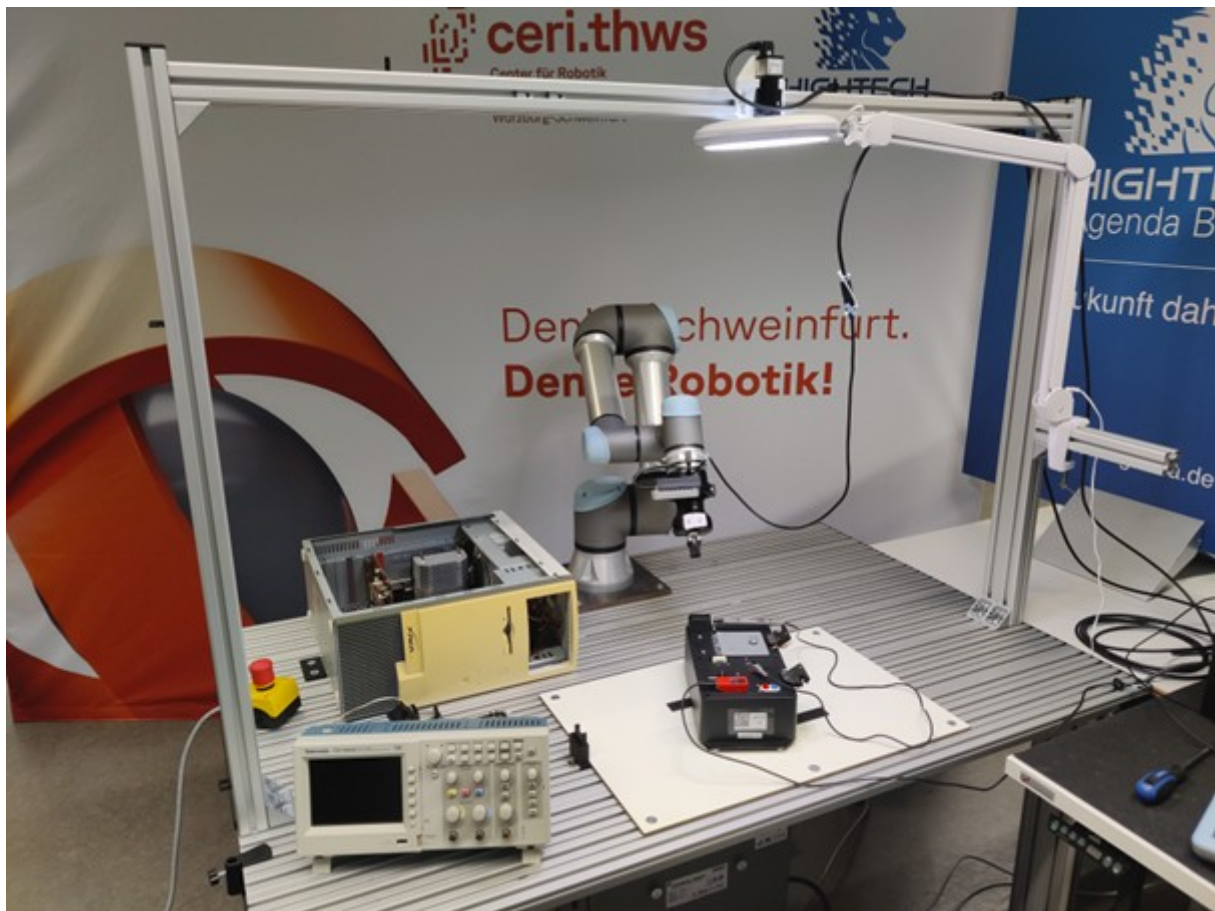








Figure 1: Workspace team Robopig

Used Equipment List



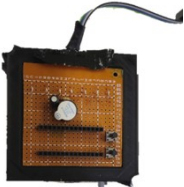

General Equipment

Picture	Equipment	Function
	UR5e Cobot	Utilization of built-in safety functions, motion planning and tactile/force sensing Controlled via The Real-Time Data Exchange (RTDE) interface
	Robotiq Hand E Gripper	Connected to the UR5e Pick all sorts of objects in the tasks
	Intel Realsense D435i	Mounted on the gripper Used for triangle detection and further image processing
	IDS This repository is the submission of team RoboPig of FHWS for the Robothon 2022 competition	Mounted in a static position over the working space of the robot Image used to localize the task board
	Additional lights	Simple ring light around IDS camera

Taskboard Task

	3D-printed cable hook	Custom tool for manipulating flexible cables For storing and accessing: 3D-printed holder mounted to the table
---	-----------------------	---

BYOD Transferability Task

	Tektronix TDS 2002B	Oscilloscope used to measure error codes of the motherboard
	Computer	Main part of BYOD-task The state of the computer is unknown in the beginning, Goal of the task is to detect errors and repair them autonomously
	Custom PCB	PCB interfacing the front panel connector of the motherboard, making the measurements safer and more reliable
	Custom plug for robot	Plug with 3D-printed attachment for picking with robot. Interfaces the standardized front panel connector of the motherboard

Hardware & Software Architecture

Our hardware and software architecture is shown in the figure below.

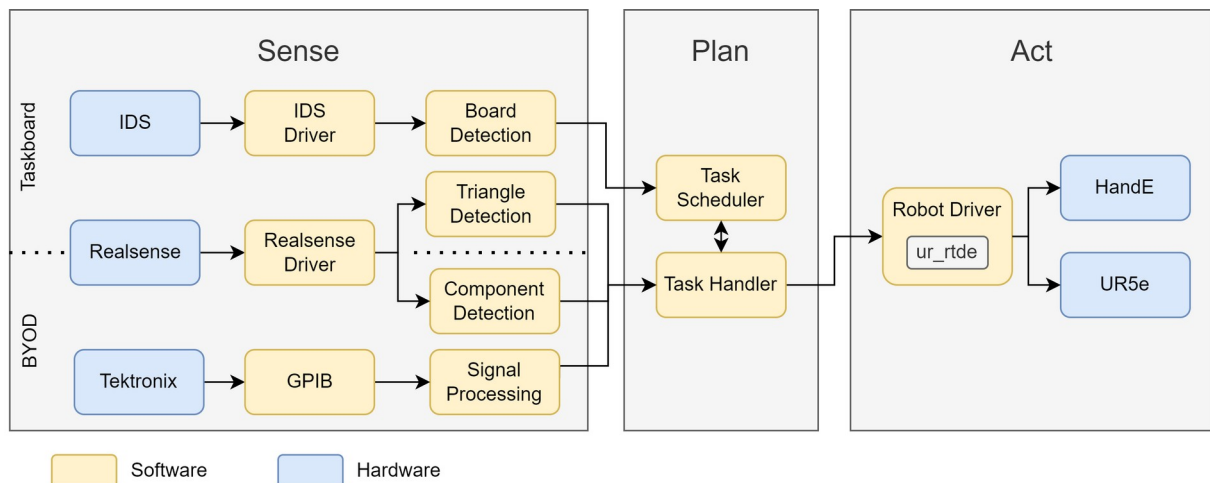


Figure 2: System architecture

Each software component is further described below (left to right):

- Sense:
 - o IDS Driver:
 - No official ROS driver available → Developed it ourselves
 - Reads out the camera, applies color correction and undistorts image
 - o Realsense Driver:
 - official Intel Realsense ROS wrapper called *realsense2_camera*
 - <https://github.com/IntelRealSense/realsense-ros>
 - o GPIB
 - The GPIB/General Purpose Interface Bus: digital interface for programmable I control.
 - Available in most Oscilloscopes → General solution for most oscilloscope types
 - o Board Detection:
 - Uses images from IDS camera
 - localizes the task board by recognizing dominant features and applying a Perspective-n-Point algorithm.
 - o Triangle Detection:
 - Detects and calculates the distance of the triangles of the second taskboard task.
 - Uses images from the Realsense camera, as the robot covers the display in many positions
 - Converts image to CIELAB color space for optimal color thresholds
 - o Component Detection:
 - Implements a Convolutional Neural Network based on YOLOv8 structure

- Trained to detect GPU and RAM
 - Used to check if components are missing inside the computer
- o Signal Processing:
 - Pulse width and frequency detection using Fast Fourier Transform
 - Estimation of tone length by using a variable sliding window algorithm
- Plan:
 - o Task Scheduler:
 - Determines which tasks and sequences should be done
 - Can be interfaced with a GUI to change task order
 - Is used for the taskboard and the BYOD-task
 - Uses generic task functions of the task handler
 - o Task Handler:
 - Contains generic functions for solving the individual steps of tasks
 - Compiles Trajectories and executes them with blending for smooth movements
 - Is used for the taskboard and the BYOD-task
- Act:
 - o Robot Driver:
 - Self-developed robot driver for UR robot.
 - Based on ur_rtde library from SDU Robotics.
 - Provides different possibilities to move the robot and gripper.
 - Transmits important telemetry data of the hardware.
 - Planned to continue development and make open source soon

Software dependencies

Component	Version	Description	Website
ROS	noetic	Communication between Software components	wiki.ros.org
realsense-ros	ros1-legacy	Reading images	github.com/IntelRealSense/realsense-ros
Eigen	-	Matrices and linear algebra	eigen.tuxfamily.org
ur_rtde	v.1.5.5	Controlling robot	gitlab.com/sdurobotics/ur_rtde
IDS-peak	-	Reading images	ids-imaging.com/ids-software-suite.html
open_cv	-	Machine vision	opencv.org
YOLO	v8	Object detection	ultralytics.com/yolov8
usb_tmc	-	Reading oscilloscope values	github.com/python-ivi/python-usbtmc

Quick Start Guide

1. Launch core driver depending on task (eg. taskboard_core.launch for taskboard)
 - a. This will:
 - i. Start cameras and begin publishing images
 - ii. Start the robot driver and begin publishing robot positions
 - iii. Start a pose management node and publish all relevant poses
 - b. System is now ready and waits for task scheduler
2. Run task scheduler and input order of tasks
 - a. This will:
 - i. Generate the desired task sequence
 - ii. Run the image processing pipeline depending on what task is chosen
 - iii. Update all positions depending on the camera detection