

KUKA YouBot Revival and System Development Project

Documentation of progress through project, in requirements of Final Project in DSD

Author: Aditya Agrawal

Advisor: assistant professor Tomasz Kucner

Supervisor: senior university lecturer Salu Ylirisku

ELEC-C0302 Final Project in Digital Systems and Design
Aalto University

May 10, 2025

Contents

1	Project Overview	1
1.1	About the Hardware	1
1.2	Teach & Repeat	1
1.3	Motivation	2
2	Initial state of YouBot	3
2.1	Hardware Overview	3
2.1.1	Robot Base	3
2.1.2	Computer	3
2.1.3	Battery	3
2.1.4	Sensors	3
2.2	Software Overview	3
2.2.1	Operating System	3
2.2.2	ROS	3
2.3	Documentation	3
3	Project Goals	5
3.1	Navigation	5
3.2	Required Hardware	5
3.3	Intended Setup	5
4	System Development	6
4.1	Peripherals	6
4.2	Battery Replacement	6
4.3	Running ROS interface	6
4.4	Software Setup	6
4.5	ROS1 Navigation Stack	6
4.6	ROS1-ROS2 Bridge	6
4.7	Sensors	7
4.7.1	Kinect Camera and libfreenect	7
4.7.2	Hokuyo URG-04LX Laser Rangefinder	7
5	Future Applications	8
6	Reflection	9
6.1	Project Management and Planning	9
6.2	Project Goals	9
A	User Manual for the KUKA YouBot	11
A.1	How to turn on the YouBot	11
A.2	.bashrc recap	11
A.3	How to run original demos	11

1 Project Overview

The following document aims to document the progress of my work on the KUKA YouBot over the course of the past three months within the Robots Lab under Professor Tomasz Kucner. The initial goal was to conduct a comprehensive revival of the robot and enable it to autonomously navigate a static environment using a low-cost camera. However, the goals shifted towards a more free-form exploration of the robot's capabilities and limitations. This document aims to document my progress and learnings, the challenges faced, and the conclusions drawn upon completion.

1.1 About the Hardware

The YouBot is a mobile robotic platform developed by German automation company KUKA. First introduced in the early 2010s [cite], it was primarily designed for research and educational purposes in the field of robotics. To further this purpose, a significant portion of software used on the YouBot is open-source and available on GitHub [cite].

The YouBot typically consists of two main parts: a mobile base and a robotic arm. The mobile base is equipped with four omnidirectional mecanum wheels and motors for movement, alongside an onboard computer for processing and control. This onboard computer uses Ubuntu and ROS1 with conveniently provided drivers and wrappers, allowing for smooth software integration as well. The robotic arm has 5 degrees of freedom (DOF) and a two-finger gripper, allowing it to potentially perform a variety of tasks through the onboard computer. Since our project involves only the mobile base, we will not be discussing the arm in detail.

The YouBot was also equipped with a variety of sensors when I received it, including a Kinect v1 camera and a functioning Hokuyo URG-04LX laser rangefinder. The Kinect camera module contains an infrared depth camera and RGB camera. This combined with its high availability and low cost made it a popular choice for navigation and perception tasks in robotics. The Hokuyo URG-04LX is a 2D laser rangefinder that was commonly used in robotics research as well, due to its compact design and high-resolution distance measurements.

This robot and the attached sensor modules are thus particularly well-suited for research within mobile robotics, particularly those pertaining to navigation and perception, as well as human-robot interaction.

1.2 Teach & Repeat

The initial prospect for this robot was to enable it to autonomously navigate using the above-mentioned sensors. Teach & Repeat (T&R) was considered to be an ideal candidate to allow for the same.

Teach & Repeat (T&R) [cite] is a two-phase robotic navigation method where a robot is first "taught" a path via human guidance or pre-recorded data. The sensor data captured during this phase can then be used to allow the robot to autonomously "repeat" the path later on, even in different environments.

T&R only requires a single camera for a basic implementation. It is a relatively simple method to implement through the use of open-source software and libraries, such as

OpenCV and ROS. Furthermore, it is robust to changes in the environment and can dynamically correct errors through the use of visual odometry. As such, it has been a very popular research topic within the field of robotics, particularly in the context of mobile robots [cite]. Much of the work done on extending T&R aims to improve this robustness and scalability for a variety of sensors and use-cases.

Thus, it seemed ideal for our project considering the age of the YouBot, the sensors available, and my lack of experience within the field of robotics.

1.3 Motivation

The motivation for this project is to document the current state of the YouBot, repair and revive it to a usable state, and explore its current capabilities and limitations, as well as the potential for future applications.

Alongside completing this project, I personally wished to gain a deeper appreciation for the worlds of navigation and perception within to robotics, as well as understand the interfaces used to operate and customize this robot. This includes understanding the ROS ecosystem, open-source software like libfreenect [cite], the various drivers and wrappers used to interface with the robot through Ubuntu, the bash scripting language, the networking protocols used to communicate between the onboard computer, the sensors as well as external computers, and the hardware components of the robot itself.

There were significant learnings made in terms of project management and planning as well, as this project was a significant departure from any other course I have taken in the past. The open-ended nature of the project and the lack of clear goals made it difficult to stay motivated at times, but it also allowed for a more free-form exploration of the robot's capabilities and limitations.

2 Initial state of YouBot

This section aims to document the KUKA YouBot's state as I found it during the start of my project.

The KUKA YouBot in question is a decade-old robot. In context of the specific one in the lab, it was last used in 2019 for a project in the AEE programme and has been unused since then. Even in 2019, the documentation was sparse and out of date and the provided lead-acid batteries for the robot were completely unusable due to sulphation.

2.1 Hardware Overview

2.1.1 Robot Base

The robot itself has 24V IN through a 3-pin XLR connector. There's two 24V OUTs on the top as well. There's an ethernet port and two etherCAT ports as well. There's many USB ports. You can use a visual display for the onboard computer as well using an ancient VGA port. [1]

There's four omnidirectional wheels, a 24V lead-acid battery (originally!). There's this metal plate on top that was installed when the robot itself was procured.

2.1.2 Computer

The onboard computer is an [] with an [].

2.1.3 Battery

lead-acid, completely dead, practically unusable for the purposes of this project.

2.1.4 Sensors

in the box there were included some sensors: two Hokuyo nodes (one of them had a completely broken Mini-USB port), as well as a KINECTv1 camera.

2.2 Software Overview

The computer runs [this software] and ROS version []. I wasn't able to immediately boot up any tests to ensure if the robot itself was working at any stage in time.

2.2.1 Operating System

2.2.2 ROS

2.3 Documentation

The documentation for the KUKA YouBot is sparse and unavailable for the most part. KUKA has removed most references to the product from their website, and the original youbot-store.com website is no longer available. Furthermore, the only available user manual for the YouBot online is a decade-old PDF file that was uploaded by a third-party user, and is marked as a draft version from 2013.

Thankfully, the YouBot was marketed as an open-source platform, and the drivers as well as various parts of software are available on GitHub. Furthermore, we were able to find some more information about the YouBot through the Wayback Machine, which had archived parts of the original website. While this archive is not complete by any means and is very tedious to comb through, it provides us with a powerful tool to piece together data about the YouBot and its use cases during its hayday.

We've also taken the liberty of citing some YouBot related projects here.

3 Project Goals

The initial goal was to "revive the robot to a usable state and enable it to autonomously navigate a static environment using Visual Teach & Repeat [VT&R]. Add in more details from the initial project plan here.

3.1 Navigation

What is VT&R?

3.2 Required Hardware

So for this we were expecting the robot base to work, the battery to work, and the kinect camera to work. And also I used another ubuntu machine to ensure the computing wasnt handled on the ancient piece of hardware.

3.3 Intended Setup

Our intended setup was to use the YouBot as a mobile base with the Kinect camera mounted on top of it. The onboard computer of the YouBot would only be used to control the motors and communicate with the sensors. The actual bulk of the processing would be done on a seperate computer that runs ROS2 and enables the necessary navigation and perception stacks.

4 System Development

4.1 Peripherals

4.2 Battery Replacement

The lead-acid batteries, as previously mentioned, were completely unusable. I contacted our lab engineer (Vesa Korhonen) and he was more than willing to assist me with replacing this battery, as well as testing the abilities of the others. replaced batteries, obtained laptop and peripherals

Some description of how lead batteries work maybe?

The old lead-acid batteries had become completely unusable for the purposes of this project. Lead-acid batteries need to be used regularly (i.e. once every 6 months) to be kept in a usable state, and even then these batteries only last for a handful of years. The batteries otherwise suffer from sulphation, which is a process where lead sulphate crystals form on the plates of the battery. This process is irreversible and leads to a significant decrease in the capacity of the battery.

There were three original batteries in the box meant for the YouBot that were more than a decade old and a makeshift one made by the lab engineer in 2019. While we were able to receive a voltage on the makeshift battery, it was far too less and the voltage quickly dropped to 0 on attempting to charge overnight. As a result, me and Vesa decided to make a quick replacement battery.

This involved combining two lead-acid batteries that we bought from Vantaa using sticky tape and then attaching them to our XLR 4-pin connector. While the battery was quite quick and dirty, it works well for our purposes. We didn't bother to even hermetically seal it since I'm probably the only use going to be using the thing anyway.

4.3 Running ROS interface

4.4 Software Setup

Discuss issues in bashrc Discuss history Discuss running original demos

4.5 ROS1 Navigation Stack

Finding the packages running these packages

4.6 ROS1-ROS2 Bridge

One of the first tasks I decided to embark on after getting the robot up and running was to set up the ROS1-ROS2 bridge. This was the most important link within the system, as it would allow the robot to seamless access the ROS2 navigation stack and enable it to access the computational resources of a more recent computer. However, the version of ROS1 on the robot is Hydro, which is more than a decade old and predates ROS2 development, i.e. it is not compatible with a ROS1-ROS2 bridge.

There are thus a handful of options to consider: One would be to upgrade the ROS1 version of the robot to a version that supports the bridge, such as Melodic or Noetic.

However, this would require a significant amount of work and may cause compatibility issues with the outdated software on the robot. Another option would be to use a different computer with a more recent version of ROS1 and use it as a middleware between the robot and the ROS2 navigation stack. This would allow for a more seamless integration of the two systems, but would require additional hardware and software setup and create an unnecessarily complex system.

4.7 Sensors

4.7.1 Kinect Camera and libfreenect

The Kinectv1 camera is a depth and RGB camera that was originally designed and sold in tandem with the XBOX 360 to enable motion tracking and gaming. Due to its low cost, high availability and ease of use, it has indirectly become a popular choice for robotics research as well.

We can use open-source software such as libfreenect to access the data from the various sensors on the module.

4.7.2 Hokuyo URG-04LX Laser Rangefinder

The Hokuyo URG-04LX is a 2D laser rangefinder that is commonly used in robotics research. It is a compact and lightweight sensor that provides high-resolution distance measurements in a 240-degree field of view. The URG-04LX is capable of measuring distances up to 4 meters with an accuracy of ± 10 mm. It communicates with the onboard computer using a serial interface, making it easy to integrate into existing systems.

5 Future Applications

In this section we will discuss a potential future application of the KUKA YouBot, in the context of the T&R method of navigation. The T&R method is a form of visual SLAM, where the robot can be "taught" a path through a static environment using a camera. The robot then uses this information to enable it to repeat the path movement autonomously through key frames it has saved from the environment. This method is simple yet effective in its use cases and can be used in a variety of applications, such as industrial automation, home robotics, service, etc.

t&r is not the best implementation of the youbot

platform and real world application

why im a bit skeptical about trying to implement t&r in the context of assistive platform trying to present as an application as a platform that can later on be used to execute repetitive task based on the experience of a platform with

speculative : presenting it in terms of real-world applications

flexible intra-logistics human awareness and flexibility seamless human-robot existence

While the YouBot holds the potential capabilities for undertaking Teach and Repeat, its abilities as a mobile industrial robot base would be better suited for real-world applications within closed environments as an assistive platform.

This robot base would be most useful in an assistive sense, where it assists humans in completing rather basic and repetitive tasks, like fetching beer.

6 Reflection

6.1 Project Management and Planning

This project was a significantly different experience from any other course I have taken in the past, especially considering its scope of 10 credits and open-ended grading. Finalizing the project plan under my advisor during mid-February gave me initial hope in realising this project early on, but personal issues as well as fear of failure led to an initial lack of motivation. It was only with meetings with the advisor to clear out communication and establish clear goals for the short-term that I felt I was able to make some progress. The project also evolved considerably over time, as I eventually realised that my initial goals were too ambitious for my calibre. We thus pivoted towards a more free-form exploration of the capabilities of the robot, and establish a deeper understanding of the theory through the same. I'd like to thank my advisor for his steadfast support and patience throughout the short yet eventful project.

This leads me to touch briefly on AGILE vs WATERFALL project management methodologies. I initially approached this project with a waterfall-based mindset, with the understanding that my project would be extremely linear and straightforward in execution. This method was promising for the first week or so, when I made substantial progress in setting up the kinect camera, the robot itself, and the ROS interface. However, because of a small misstep I made in my planning by jumping into the ROS1-ROS2 bridge, reaching a dead end in that topic made me lose all my momentum overall.

This led to a lack of motivation, procrastination, and disappointment in my own abilities and growth during the initial stages, despite the quick learning I did make. On discussing this with my advisor, he advised to approach this with a much more short-term agile mindset, acknowledging that the project would not fulfill its initial goals, but that it would provide a more reasonable approach to the project and allow for solving projects in a much more methodical manner. While adjusting to this mindset was initially difficult, by the end of the project it felt much more rewarding to approach it in this manner.

The AGILE mindset also allowed me to explore the robot in a much more free-form manner and keep my work interesting whilst making progress.

Overall, despite the lackluster results, I am happy with my progress and gained knowledge. I feel quite comfortable with the ROS ecosystem, Bash scripting, and Linux in general. I also gained a deeper appreciation for the mechanical and electrical components of the robot itself, and how it all comes together to create a functional system. I'd like to thank my advisor for his provision of a different perspective on the project, explanations of everything and constant patience.

6.2 Project Goals

References

- [1] “Youbot - Overview — github.com”, <https://github.com/youbot>, [Accessed 23-04-2025] (page 3).
- [2] I. Tsogias, “Kinematic Analysis of a KUKA YouBot”, 2016. DOI: 10.26240/heal.ntua.12581. [Online]. Available: <https://dspace.lib.ntua.gr/xmlui/handle/123456789/43358>.
- [3] “YouBot Store — web.archive.org”, [Accessed 23-04-2025]. [Online]. Available: %5Curl%7Bhttps://web.archive.org/web/20170318163736/http://www.youbot-store.com/developers/%7D.
- [4] “YouBot Driver — janpaulus.github.io”, <https://janpaulus.github.io/>, [Accessed 23-04-2025].

A User Manual for the KUKA YouBot

A.1 How to turn on the YouBot

A.2 .bashrc recap

A.3 How to run original demos