Coupling of Virtual and Real Robotic Arm

Robert Roth

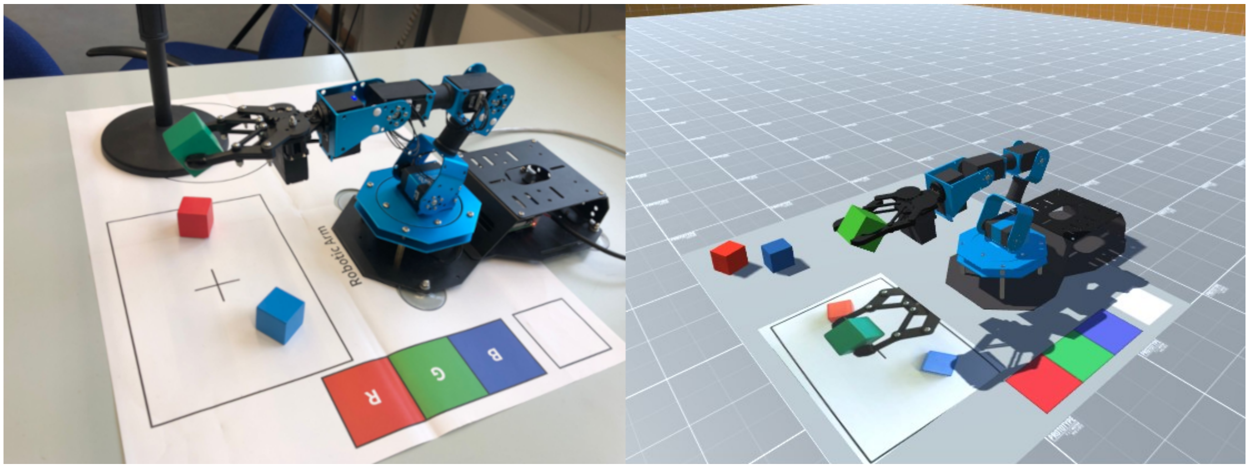

Fig. 1: The real robotic arm and the digital twin

1 Introduction

With the continuous advancement of digitalization, the coupling of the real world with the digital world is gaining importance. A result of this are digital twins. Digital twins are virtual replicas of real objects that simulate their behavior while exchanging data bidirectionally with their real counterpart. With their help, processes can be virtually tested in advance to identify errors and evaluate alternatives. In the context of the original thesis, a digital twin for a robotic arm capable of recognizing, grasping, and relocating colored cubes was developed. Subsequently, communication between the twins was implemented. The twins can exchange status information about the position of the cubes or the orientation of the robot joints over a socket connection. Additionally, a custom API was developed to simplify programming of the robotic arm. Following this, a two-part evaluation was conducted. The first part showed that the cubes are mostly well grasped by the ArmPi, but most errors occur in the corners, and stacking the cubes often fails. The second part of the evaluation consisted of a user study on the API, which showed that the API is easy to understand and complete.

However, the poster or presentation at StuKoI should focus on the functionalities of the digital twin, as this is more illustrative and easier to demonstrate. Thus, the presentation should be adapted to the time frame of the event.

2 Fundamentals and Implementation

The work is based on the *ArmPi*, a robotic arm from *Shenzhen Hiwonder Technology*. It consists of 6 movable joints, also known as servos, controlled by a Raspberry Pi. The robotic arm is delivered with a camera, a floor map, and three colored cubes (red, green, blue).

The digital model is used in *Unity*. Unity is primarily a runtime and development environment for games, but can also be used for the development of interactive 3D applications, as in this case. To turn the digital model into a digital twin, the physical properties of the real robotic arm must first be simulated by the digital model. For this purpose, the servos were first adjusted so that they could be controlled individually. In the next step, the interaction with other objects, especially the digital cubes, was adjusted. The digital robotic arm should be able to grasp and reposition the cubes, just like the real robotic arm.

The last important step for a twin is the bidirectional communication between the real and the virtual robotic arm. For this purpose, both robotic arms were extended with the functionality to send messages to each other and receive them. This was used, for example, to track the real cubes and adopt the position of the virtual cubes. In addition, both robotic arms were adjusted so that they can assume the same servo positions at any time. This change made it possible to create a completely identical initial state from which the simulation can start.

The ultimate goal of the digital twin is to test and debug program code from the real robotic arm with it. A major challenge in this endeavor is that there is no interface through which program code for the robotic arm can be written in a user-friendly manner. For this reason, an API should be written as an interface to facilitate programming of the robotic arm for users. With this API, it is now possible to recognize and reposition objects with relatively little programming experience.

3 Presentation at StuKoI

The presentation on the digital robotic arm at StuKoI should include a brief introduction to digital twins as well as the ArmPi itself. In particular, it should also show what control options are enabled by the API. Following this, a live demo, in which both the real robotic arm and its digital twin are introduced, should be shown. In the live demo, the real robotic arm should recognize a cube and move it to a specific position. The digital twin should imitate this movement.