APPLIED ARTIFICIAL INTELLIGENCE

Robotics

PROJECT REPORT

Author:

Arthur VANDENHOEKE

Professor :

Denis STECKELMACHER

June 4, 2021

Contents

Chapter 1

Introduction

In this robotics class, we consider the task of allowing a person to hold his/her Raspberry Pi in front of a laptop's webcam. The Raspberry Pi, equipped with its sense hat, allows to display a single red dot on its LED matrix so that the webcam can monitor it. The goal of this project is to ensure that the red dot always remains as close as possible to the webcam's captured image. This must particularly hold if the person shakes, moves, or rotates the Raspberry in front of the webcam. If the Raspberry Pi's LED matrix is placed too far from the webcam's image center, the red dot should be placed as close as possible to the latter.

If we were to solve this problem manually, we would need to have access to the Raspberry Pi's accelerometer and gyroscope to determine the speed and the direction of the motion of the red dot on the LED matrix. This would require us to use a reference frame for which the origin would be the center of the webcam's image. Access to acceleration and gyroscopic data would thereby allow to construct a velocity vector anchored at the red dot and pointing towards the origin of this reference frame. Doing so would enable us to undo the person's motions and therefore stabilize the red dot at the center of the image. Although this approach seems technically feasible, its implementation is not straightforward.

In this course, we intend to use Reinforcement Learning (RL) as a "last-resort" solution to overcome the complexity of the previous approach. Our goal is thus to produce a *policy* that allows to move the red dot on the Raspberry Pi's LED matrix so as to be as close as possible to the center of the image. In order to achieve this goal,
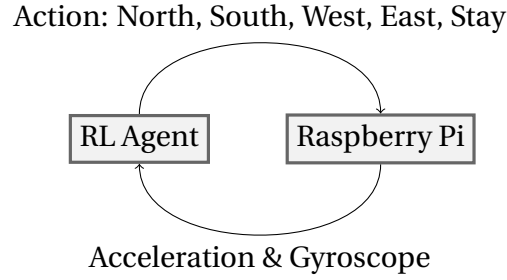
Action: North, South, West, East, Stay

Figure 1.1: Communication between the Raspberry Pi and the RL agent. The Pi sends its acceleration and gyroscopic data over to the agent which decides what action to take in order to re-center the red dot on the Pi.

we will need the following components:

- A Raspberry Pi with a mounted sense hat (comprises the LED matrix);

- A laptop with a front webcam;

- A Reinforcement Learning agent.

In the following sections, we start by explaining how the Raspberry Pi can be set up to communicate with the laptop over the red dot's position on the webcam image. We also describe how the red dot's position can be retrieved from the raw image using image processing techniques. The last section of this chapter investigates the set up of the RL agent and its corresponding environment.

1.1   Configuration of the Raspberry Pi

In order for the RL agent to decide how to move the red dot on the Raspberry Pi 's LED matrix, the agent must have access to acceleration and gyroscopic data of the Pi. In return, the Pi must have access to the decision of the agent in order to re-center the red dot following some perturbation of the Pi. The actions of the agent are (1) *Move North*, (2) *Move South*, (3) *Move West*, (4) *Move East* and (5) *Stay*. Both entities thus communicate according to the scheme of figure 1.1.

To ensure a communication between both entities, we implement a Python socket server on the Pi in order to send sensor readings to the computer on the same wifi

network. In return, the computer sends an action to the Pi for it to re-center the red dot.

1.2 Webcam image processing

1.3 Reinforcement Learning agents