Milestone 1: Teleoperating the Robot with your Keyboard

Lab 2: Implementing teleoperation codes

We will be using the Alphabot2-Pi robot for our lab project. This first milestone will allow you to become more familiar with the robot, and with some useful controls.

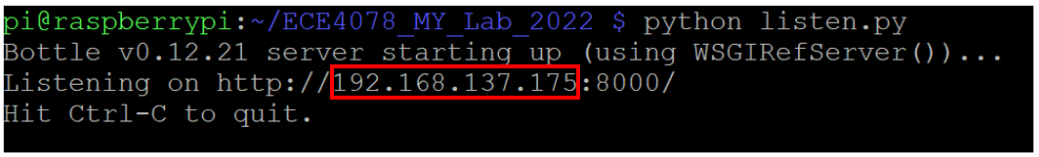
How the codes/file structures work

There will be two sides of codes, the server side (the robot) and the client side (your pc). On the server side, the codes are responsible for low-level control of the robot. The robot will run a script (listen.py) which actively listens for requests from the client to either move the robot (forward, backward, left, right, stop) or get a screenshot from the camera. The scripts on the server side will be provided for you through a public github repo (detailed description below). All robots will run the same script and you are not required to edit the scripts here.

The client side is where all the processing, computation and high-level control will be executed. For every milestone, skeleton code and files will be provided for you in Moodle/Google Drive, and you are required to download and modify the TO-DO sections in the scripts. The scripts will be run on your PC, and you can work on them together via GitHub. Further setting up instructions will be described below.

Objective 1: Setting up the server side

1. Connect the Alphabot2 to your PC using PuTTY.
2. Install some packages by typing the command:
   `sudo pip install opencv-python bottle`
   `sudo pip install -U numpy`
   `sudo apt-get install libatlas-base-dev`
3. Type `ls` in the terminal and press enter. You should see a list of folders/directories, including the private repo that you have cloned in Lab 1.
4. Your private repo will not be needed anymore, as we will be providing you the server code via a public repo. So, delete the repo using the command:
   `rm -rf REPO_NAME`
   You can type `ls` again to check that the repo is indeed removed.
5. Next, get the server files and codes using the command:
   `git clone https://github.com/clow0003/ECE4078_MY_Lab_2022.git`
   Note that you can paste in the PuTTY terminal by pressing the right mouse button.
6. Then, `cd` into the cloned repo, and run the server script using python listen.py
7. Take note of the shown IP address, which will be used later in the client side to make requests to the robot.



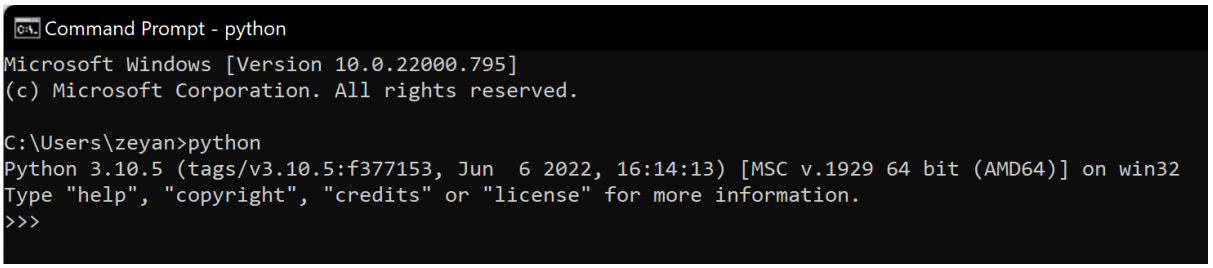
Note: Every robot will run the same server script. You can have a look at the scripts to see how the robot is controlled, however it is unlikely that you are required to modify the codes here. We will ask you to pull any new changes in the event that the server codes are updated in subsequent labs.

Objective 2: Setting up the client side

Part A: Setting up Python environment in your PC

First, we need to download and set up Python on your PC. Note that this guide is for Windows operating system only. Also note that if you have prior experience with Python and you already have Python installed in your computer, you can skip this part.

1. Download python from https://www.python.org/downloads/
2. Launch the installer, check the option for "Add Python to PATH", and click install now. You can also go for a customized installation if you want. After installation, ignore "disable path length limit" and click close.
3. Open up a window's command prompt and type "python". If the symbol ">>>" appears, then the installation is successful.



4. Press Ctrl+Z then Enter to exit the python environment. Then, install some packages by typing the command:
   `pip install numpy opencv-python pygame requests`
5. Close the terminal when done.
6. To edit python scripts, there are many kinds of editor/IDEs available on the internet. We recommend using the simple Notepad++, but feel free to use other IDEs.

Part B: Running the client scripts

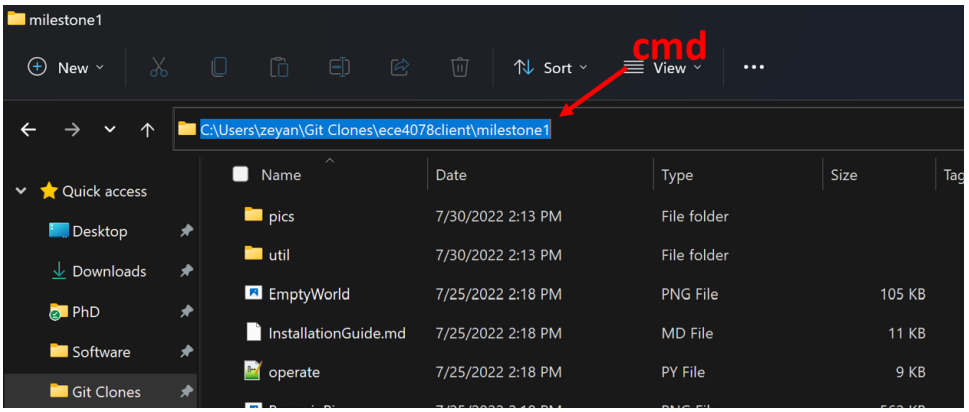
1. Download the skeleton codes (milestone1.zip) from Moodle.
2. Unzip it and move the content to the local repository folder that you cloned in Lab 1. The files structure in your PC should look like this:

```
YOUR_REPO_NAME
└ milestone1
      └ <files in milestone1>
└ milestone2 (next time)
```

In your GitHub Desktop, you can push these new changes so that your teammates can pull these files and work on the code together. However, only 1 PC is needed to run the script later.

3. Open a window command prompt and `cd` to the milestone1 folder. Alternatively, you can open the milestone1 folder, click the location bar at the top of the File Explorer,

type “cmd” without the quotes, and press Enter.



4. In the terminal, run the client script by typing the command:
   `python operate.py --ip IPADDRESS --port 8000`
   Replace `IPADDRESS` with the address found from Objective 1(PuTTY terminal).
5. A Pygame window should pop up as follows.

   Press any key, and you should see the window changes to the main interface which shows the images captured by the bot. If you can see it, all the set up have been done successfully. Press Ctrl+C in the terminals to stop the program.
6. Your task: Implement keyboard teleoperation by editing line 137 - 150 of `operate.py`. Study the codes in `operate.py` and `util/pibot.py` to determine what lines you have to write.
7. Test your implementation by running step 4 again. Make sure that the server script is running before you run the client script.

You don't have to use the provided scripts. Feel free to be creative and write your own scripts for teleoperating the robot with a keyboard.

Marking Guidelines

Basic implementation submitted on Moodle (80pt):

- Drive forward +20pt
- Drive backward +20pt
- Turn left +20pt
- Turn right +20pt

Demonstration (20pt): demonstrate your teleoperation in the lab to the demonstrators during Week 3 lab session.