Virtual Tag

A game you all know and love ... with a twist

Table of Contents

Installing

Here we will talk you through is needed to be installed on both your PC and Raspberry Pi. These are needed for the game to run correctly.

Installing Python[1]

1. Download Miniconda or Anaconda using the embedded links (make sure you download Python 3).
2. Install conda:
   a. **Windows**: Run the installer. Use default settings unless you have any slight concerns. From here on out, you will be using the installed Anaconda Prompt that was installed to do commands

   b. **Mac/Linux**: *Miniconda (bash):* Use the terminal to
   ```
   $ (sudo) bash [name of file you downloaded]
   ```
   If absolutely necessary, try using sudo in front of this line to install. When you are accepting the license conditions, press 'q' to acknowledge that you want to quit out of the license before typing yes.

   To check if this works, type:
   ```
   $ conda --version
   ```
   and
   ```
   $ conda update conda
   ```

   ```
   $ conda update conda
   ```

   For anything else you want to do, you will also have to run the command
   ```
   $ conda install numpy
   ```

Setting up a Virtual Environment[2]

A virtual environment is not necessary however it will create an easy environment to enter when you want to run the game, with all the right programs. To create this environment simply use the following command line in your terminal:

```
$ conda -V
$ conda update conda
$ conda create -n yourenvname python anaconda
$ conda activate yourenvname
$ conda install -n yourenvname pip
$ conda install -n yourenvname numpy matplotlib pandas
$ conda deactivate
```

---

[1] Pavan Holur, Jeffrey Jiang, "Introduction to SW Development and CV" from Tutorials for 180DA/DB on CCLE, 4-5

[2] Pavan Holur, Jeffrey Jiang, "Introduction to SW Development and CV" from Tutorials for 180DA/DB on CCLE, 5-6

```
Note: yourenvname is the name you want to call the new
environment.
```

Now you can activate this environment when you want to using

```
$ conda activate yourenvname
```

Installing OpenCV[3]

To install (if you are using a virtual environment, enter your virtual environment):
```
$ conda install -c conda-forge opencv
```

To verify installation:
```
$ python
>> import cv2
>> import numpy as np
>> Ctrl + D (to close out of python script)
```

Installing Pygame, MQTT, and PyAudio

In your command line and in your virtual environment (if you're using one) run:
```
$ python3 -m pip install -U pygame --user
$ pip install paho-mqtt [4]
$ pip install pyaudio
```

If this audio installation does not work, then use:
```
$ pip install pipwin
$ pipwin install pyaudio
```

Setting up the Raspberry Pi[5]

Make sure that you have a Raspberry Pi with the most recent operating system.
1. Connect your micro SD card to your laptop. **(Do not reformat the drive. If givent the option!)**
2. Open config.txt. At the end of the document, add the following two lines. These will allow you to implement all the connections that you need.

   ```
   dtoverlay=dwc2
   enable_uart=1
   ```

3. Now open cmdline.txt. You will find that the contents are just a single line, and an empty line underneath. Without changing anything else, add the following at the end of this line.

---

[3] Pavan Holur, Jeffrey Jiang, "Introduction to SW Development and CV" from Tutorials for 180DA/DB on CCLE, 6-7

[4] Pavan Holur, Jeffrey Jiang, "A Sample Communications System" from Tutorials for 180DA/DB on CCLE, 4-5

[5] Pavan Holur, Jeffrey Jiang, "Raspberry Pi Setup" from Tutorials for 180DA/DB on CCLE

```
modules-load=dwc2,g_ether
```

4. Finally create a single empty file called `ssh`. The ssh file will disappear upon loading it back onto your Raspberry Pi.
5. To connect via a wireless connection, we must create one more file called `wpa_supplicant.conf`. In this file you must have:

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=US

network={
ssid="YOURSSID"
psk="YOURPASSWORD"
scan_ssid=1
}
```

With your network and password. **The network cannot be a 5GHz network.**
6. Insert the micro SD back in the Raspberry Pi and allow the processor to read the new files.

7. To find the IP address of your Pi, log into your wifi's admin page and find the list of users. The Raspberry Pi should be listed under this and the IP will be nearby on the same page.

8. Once you have the IP, to get into the Raspberry Pi, you must go into your command line and use the command:
```
$ ssh pi@[insert IP address here]
```

And insert your password or the preset password 'raspberry'.

9. In the Pi command line you must run:
```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install git
```
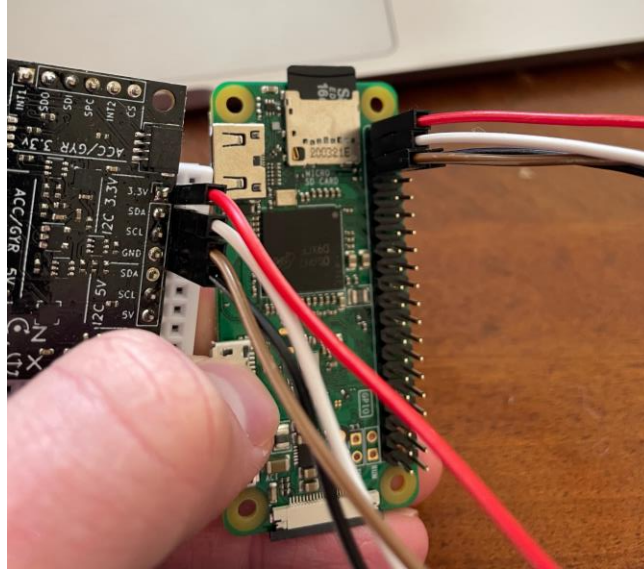
Installing Libraries on your Pi[6]

These are commands you must run to install the necessary libraries on your Pi.
```
$ pip install paho-mqtt
```

[6] Pavan Holur, Jeffrey Jiang, "A Sample Communications System" from Tutorials for 180DA/DB on CCLE, 4-5

Setting up the IMU (BerryIMU)[7]

1. Wire your IMU up as shown in the image above.
2. Within the Raspberry Pi command line and with the IMU attached, run:
```
sudo apt-get update
sudo apt-get upgrade
conda upgrade conda
conda update conda
pip install –upgrade pip
sudo apt-get install git i2c-tools libi2c-dev
conda install -c conda-forge smbus2
sudo apt-get install vim
```

3. Try opening the blacklist file
```
sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

   If the file is empty or does not exist, leave it that way. If it is not, if there is the line blacklist i2c-bcm2708, put a '#' in front to comment this line out.

4. Add this into /etc/modules by using nano/vim
```
i2c-dev
i2c-bcm2708
```

5. Add this into /boot/config.txt
```
dtparam=i2c_arm=on
dtparam=i2c1=on
```

---

[7] Pavan Holur, Jeffrey Jiang, "Basic BerryIMU Usage" from Tutorials for 180DA/DB on CCLE

6. Reboot your Raspberry Pi using
   `sudo reboot -h now`
   and reopen your connection to run
   `sudo i2cdetect -y 1`
   to check if everything installed correctly

   This shows that there are 3 devices found, which may include the USB port and the BerryIMU. As shown in the guide, 0x6a is the gyroscope, 0x1c is the accelerometer and gyroscope, but you can double check this on the datasheet. If everything is a dash, then something did not work (your IMU isn't being read). When this works, this means that the code is able to read your IMU and you can move on to actually interfacing and writing code that uses the IMU.

```
pi@raspberrypi:~ $ sudo /usr/sbin/i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- 1c -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- 6a -- -- -- -- --
70: -- -- -- -- -- -- -- 77
```

Startup

The startup is relatively simple but make sure to follow these steps before running.
1. Download the game files and put them in your preferred location on your computer.
2. Copy this file over to your Raspberry Pi Zero as well.
3. In the command line, navigate to the directory that you have your game files in for both your computer and your Pi.
4. When everyone is ready, you can run the command python3 main.py on both the computer and the Raspberry Pi.
5. One player will choose primary, DO NOT HAVE MULTIPLE PRIMARIES.
6. The primary player will then choose the settings in the settings menus. Go ahead and play with the settings to change up the fun!!

Gameplay

This game is limitless fun, expanding beyond the 2-D playscape and introducing new dimensions to the game of tag.

Settings

Number of Players: This game is set for 1-10 players and
Size of Play space: You can choose a play space from 6x6 to 20x20.
Game Type: Standard game type will play until the last person is tagged; Infinite will play indefinitely until the users quit the game.
Number of Obstacles: Choose the number of obstacles you want in the space.
Number of Powerups: Choose the number of powerups you want in the space.

Rules

The rules are simple! Tag your friends while they use power ups and axis swaps to avoid you. The last person to be tagged wins! Of course, if you don't want the fun to end you can simply use the sliders in the settings menu to play infinitely.

Knowing where and who you are:
- First, the playspace pops up and you will see YOUR PLAYER blink before anyone can move.
- The other indicator of your player color is around the frame displaying your face

Knowing where you are:
- Look for your blinking player when the display pops up

Knowing if you are "IT":
- Underneath your camera feed, there is text that tells you if you are it, or if you should run.
- Around your camera feed, there is a lighter ring of color surrounding the player color.
- Around your player there is a lighter ring of color surrounding the player.

Powerups

To activate your powerup, simply tell the computer "powerup" and it will use the powerup in your possession

1. Freeze: Freezes all other players.
2. Speed Boost: Boosts your speed.
3. Teleport: Switches your place with another player.

To drop the powerup, say "drop".

Controls

To control the player, you need to have your control arrow. When the game space is created, there is a feed from your video camera with a small constraint box in the frame. This is where you want to use your arrow. By pointing it in the direction you want your player to go, the pose recognition will pick up on the direction and move. For most accurate reading, make sure you have a good light source shining towards you and that you're relatively close to the camera!

Print out this page and fold or cut out along dashed line to use the arrow on the right as a controller.