Coding Assignment 4

EC 400

Due on 11/10/2021

Setup. In this homework, we will design a simple low-level controller that acts as an auto-pilot to drive in supertuxkart. In a later homework, we will use this auto-pilot to train a vision based driving system. To get started, first download and install SuperTuxKart on your machine. On a linux system, you can do this by writing

```
pip install -U PySuperTuxKart
```

This command may fail because your system lacks other packages. You should then install those other packages. Once you have installed this, you should post about your experience on Piazza.

Assignment. Fill in the function control in the file control.py. The input of this function is an aim-point towards which the car should move, and the current velocity. The aim point uses screen coordinates: $[-1, -1]$ is the top-left of the screen, $[1, 1]$ is the bottom right.

What it should return is an action. You need to specify:

- action.steer the steering angle of the kart normalized to $[-1, 1]$.

- action.acceleration the acceleration of the kart normalized to $[0, 1]$

- action.brake boolean indicator for braking

- action.drift a special action that makes the kart drift, useful for tight turns

- action.nitro burns nitro for fast acceleration

Here are some hints. Note that this is one possible way to solve this, and you may do something totally different that works.

- Play around with setting a target velocity, and setting your acceleration to achieve that velocity.

- If the first entry of your aim point is negative, that means you want to turn left, since the target point is to the left of the center of the screen; and if the first entry of your aim point is positive, this means that you want to turn right.

- If your aim point is too far left or too far right, this means you are turning hard and should set the drift to True.

Once you are finished writing controller.py, you should be able to test your controller by writing

```
python3 -m controller [TRACK_NAME] -v
```

For track name, plug in "zengarden", "lighthouse," "hacienda", "snowtuxpeak", "cornfield_crossing", "scotland". You should complete the first two under 50 seconds, the next two under 60 seconds, and the final two under 70 seconds. Note that the output will report the number of frames rather than seconds, and there are ten frames per second.

Visualize the results. In Anaconda Spyder, you can press F10 and it will create a nice video of your results. But you need to do something else for the next coding assignment. First, you will need to install EGL using anaconda prompt

```
conda install -c anaconda mesa-libegl-cos6-x86_64
```

After that, run

```
python3 -m utils.py zengarden lighthouse hacienda snowtuxpeak cornfield_crossing scotland
```

This should create a directory called "drive data" where images of your controller will be created. **You will need these images for the next coding assignment.**