Mini Project

Self-driving Jet Racer

Introduction:

Artificial intelligence is central to modern transport and has a vital role in the development of self-driving cars. The goal of this project is to use the AI techniques learned in this course to develop and apply an AI based model to enable the Jetson-Racer car [1] to drive autonomously. Before you start with this assignment, you can familiarize yourself with the Jetson-Racer by reading the information given on the links [1] and [2].

After reading this document, also go through the Jupyter Notebook files uploaded on Brightspace for this project.

In this assignment you will use the live images taken from the Jetson-Racer camera as input to your AI model. Based on these live images the car must decide autonomously how to stay on the track and whether it should turn left or right. It should follow the track in all situations. For example, if the track takes a left turn then the car should follow this track by steering to the left.

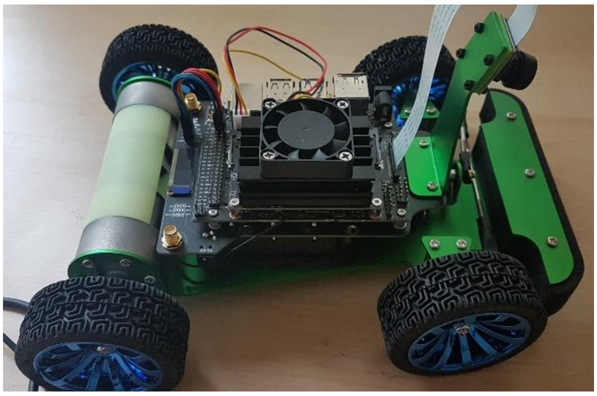


Figure 1: Jetson Racer

The idea is to make an AI model based on neural networks (NN or CNN). The model can generate the output which helps the Jet-Racer to make decisions such as: turn left, turn right, and go straight (see figure 2). In order to make such decisions you need to collect images from the Jet-Racer camera to train your AI model.

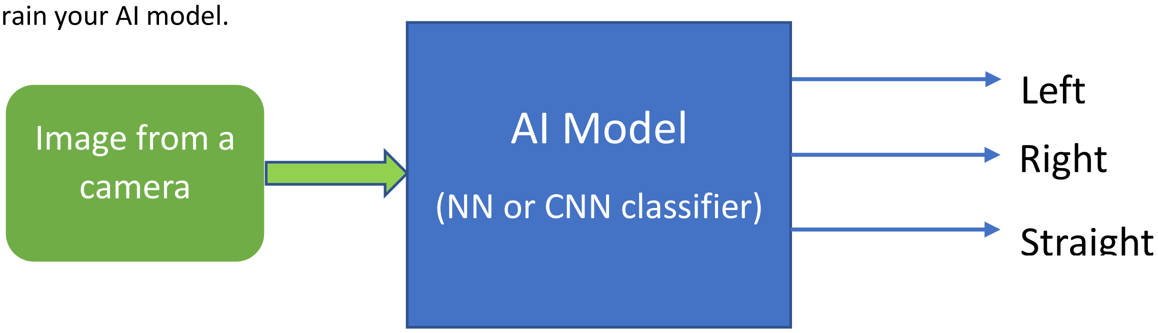


Figure 2 : Possible AI model structure

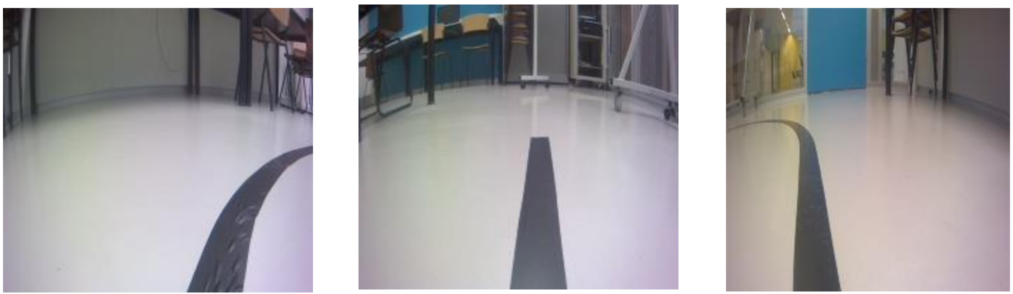
[1] https://www.waveshare.com/jetracer-ai-kit.htm

[2] https://www.waveshare.com/wiki/JetRacer_AI_Kit

Design Steps

The following steps will help you in this project:

1. **Connecting to Jet-Racer and starting JupyterLab**
    a. You need to connect to the Jet-Racer car through your PC using the WiFi router located in D1.052. The network name is *hhsjetracer* and the password is *EMES2022*.
    b. After you have connected to the network, you can turn ON your Jet-Racer. Wait until you see an IP address displayed on the Jet-Racer LCD screen. Use this IP to connect with Jet-Racer. For example if the Jet-Racer IP is 168.192.201.13 then you need to type the following in your web-browser (preferably Chrome or Firefox) 168.192.201.13:8888 and press enter. You will be asked for the password. The default password is *jetson* . After entering the password you will be connected with your Jet-Racer via JupyterLab. Through this interface you can then create notebooks and execute code on your Jet-Racer.
2. **Creating a workspace**
    a. Once you are connected to the Jet-Racer via JupyterLab, create a new directory for your group under /home/jetson/jetracer/student_work/. Please try to keep all your work under this directory.
    b. Copy the example Jupyter notebooks and python files from Brightspace into this directory.
3. **Testing your Jet-Racer**
    a. You can open the *basic_motion.ipynb* Jupyter Notebook and go through the code to test the steering and throttle control of your jet-racer.
4. **Collecting images**
    a. To collect images for your dataset open the *mini_proj_image_collect.ipynb* file and examine the code. By running this file you will have a drop down menu to select a situation (i.e, left, right , straight) in which the jet-racer is placed. Once the situation is selected, you can take a snapshot by clicking on the image. The image will be saved in the respective folder, left, right or straight. Similarly take at least 50 images per situation. So in total you will have at least 50x3 = 150 images. Be careful in placing the Jet-Racer per situation, make sure that you capture all the situations that Jet-Racer will face in the real test.



Right Straight Left

Figure 3 : Possible situations in placing the Jet-Racer

    b. The collected images will be saved in the folder AAI_ES_A including subfolders *right*, *left*, and *straight*. To download this dataset folder AAI_ES_A in your laptop, you can open the terminal in the Jupyter Notebook, *cd* to your local working directory and make a compressed tar file by executing the following command.

tar -czvf AAI_ES_A.tar.gz AAI_ES_A
Now you will see a compressed file ***AAI_ES_A.tar.gz*** under files in the Jet-Racer Jupyter Notebook. You can right click on that file and click download to save this in your laptop. Now you have successfully collected and saved the images dataset in your own laptop.

5. **Building and Training your AI model**
    a. You will build and train your model on your own PC and not on the JetRacer itself.
    b. It is recommended that you create a new conda virtual environment for your project that uses Python version 3.7.
        i. You can follow the instructions that you used in the first lab, but just use: *conda create -n aai_proj python=3.7*
        ii. You may also want to install opencv for image preprocessing tasks:
            1. To install, type the following in the anaconda prompt after activate the correct environment: conda install opencv
            2. To import in python, type: import cv2
    c. You can use the *mini_proj_train_model.ipynb* file to load and prepare the images in your dataset. The code will add labels to the dataset based on the subfolder names. If necessary perform some image cropping and other preprocessing steps. You can extend this file to build your AI model using the Tensorflow Keras API. Train, validate, and test your model until you are satisfied with the results. Once you are satisfied with the performance of your AI model you can convert and save this model in Tensorflow Lite format.

6. **Deploying your model**
    a. When your model is ready you can copy the tensorflow lite model file to the Jet-Racer by dragging and dropping from your laptop**.**
    b. Open *mini_project_run_model.ipynb* and read through the code to see how you can control the Jet-Racer. Add your final code to control the Jet-Racer using the output from your AI model.

7. **Testing your performance**
Test your Jet-Racer at different speeds (i.e, throttle value) . Start with low speed and see if the car can stay on the track. Slowly increase the speed to test the limits of your Jet-Racer.

Deliverables

The following two deliverables are required:

1. You are required to give a live demonstration in week 8 or week 10 (resit) of your project.
2. You must submit the complete code for your design (written in Jupyter Notebooks) in a .zip file written to Brightspace. The code should be well documented and easy to read.

Evaluation Criteria

You will be evaluated on the following 9 criteria:

Criterion	Quick Explanation
Data collection, proper splits	Did the students collect proper training data, label it and split it into training/test/validation sets
Data understanding and preparation	Did the students investigate the data and perform sanity checks? Did they do any transformations like cropping, changing color scales, etc?
Modelling	Did they try different models, with different hyperparameters? Did they draw proper conclusions and select the best model?
Evaluation	Did they do a proper evaluation of different models with an appropriate performance measure?
Deployment	Does the car run autonomously?
Speed	How slow/fast does it go?
Both directions	Does the car run counter-and clockwise?
Changing light conditions (under table)	Is it able to deal with changing light conditions, e.g. able to drive under the table?
Code well documented with proper conclusions	Is the code well documented, understandable, and contains proper conclusions of the analysis?