

Where do I park

This project aims to create an easy to use solution for detection of free parking spaces using a low-end camera and Raspberry Pi

First, we needed to decide which technique to use to detect parking spaces. One available is to create a training set with marked parking spaces. However, marking dozens of images can be cumbersome and rather time-consuming. Moreover, the goal of this project is to be accessible to everyone regarding their CS skills. Therefore, we went for a solution that does not require additional setup.

We detect a car using an object detection algorithm. Then we create a box around each car and calculate intersection over the union. Finally, based on a calculated proportion we decide whether the car is present or not.

Another issue we faced was to choose the right algorithm for object detection. Few available are:

- Histogram of Oriented Gradients
- Convolutional Neural Networks
- Masked Region-based Convolutional Neural Networks

We decided to use Masked R-CNN as it promises the accuracy of CNN along with other features to efficiently speed up the detection process. There are few open-source implementations of Masked R-CNN algorithm. Firstly, we considered implementation from Matterport. Nonetheless, in the end, we decided for implementation from Facebook Research called `detectron2` as it's still actively maintained.

Results we achieved from car detection using the aforementioned strategy can be seen in figures section.

During the second semester we took car detection one step further, to free space detection. We accomplished it using the agreed upon algorithm from the first semester.

Next, to make it easier to end user to use & deploy, we intergated the parking space detection with well known messaging application that supports bot users. We've chosen `telegram` for its nice & straight-forward bot API.

We made a python package that is installable by a single command:

```
pip install \
  -f https://download.pytorch.org/whl/torch_stable.html \
  -f https://dl.fbaipublicfiles.com/detectron2/wheels/cpu/torch1.5/index.html \
  git+https://gitlab.com/matusf/where-do-i-park.git
```

The package installs a CLI programme that user can use to run the bot. The programme can be configured using command line arguments or environment

variables. Here is result of `help` command

```
$ run-park-bot -h
usage: run-park-bot [-h] [-t TELEGRAM_TOKEN] -i VIDEO_SOURCE
```

Available telegram commands: `/help` `/park`

optional arguments:

```
-h, --help          show this help message and exit
-t TELEGRAM_TOKEN  telegram bot token (env: TELEGRAM_TOKEN)
-i VIDEO_SOURCE    video source input (env: VIDEO_SOURCE)
```

When programme is running, user can communicate with the bot using chat application. Available chat commands are `/help` and `/park`.

Full conversation flow with bot can be seen in figures.

Student: Matúš Ferech

Project lead: Marek Šuppa

Figures



Figure 1: original image



Figure 2: masked image

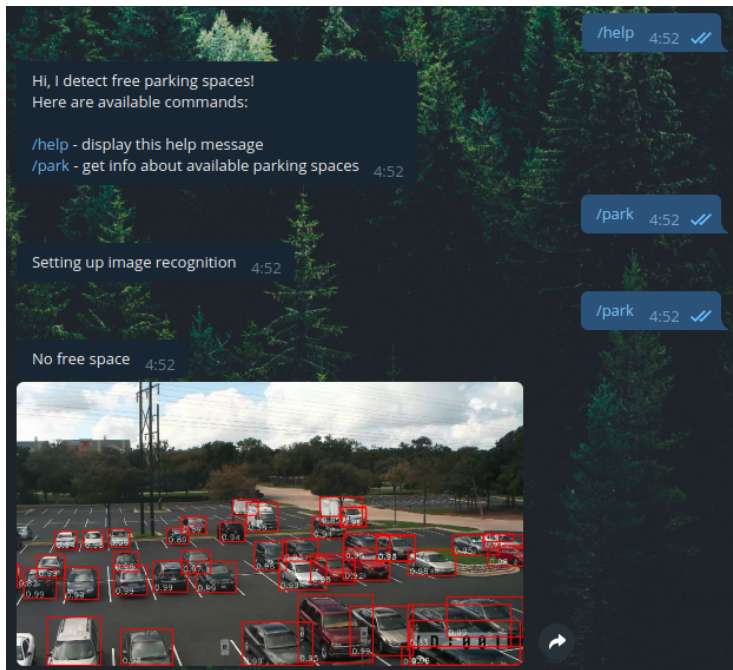


Figure 3: Unavailable

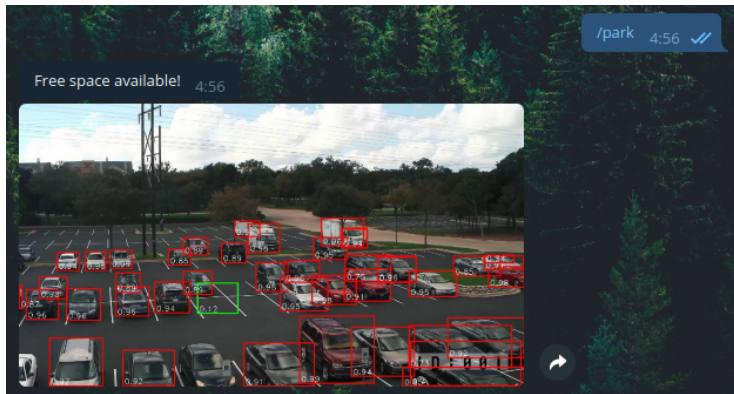


Figure 4: Free space detected