Project Course

Goal: Learn to build an object detector from the scratch by TensorFlow object detection API or TensorFlow Model Garden. Learn to how to use one of the two APIs for a new dataset, evaluate it and finally make a prediction on test images.

This project should be done as a team work (2 persons) or you can do it alone.

Deadline: 06.03.2025 at 24:00.

Tool: For labeling the images (labelImg) and for object detection (Tensorflow Object Detection API)

Data set:

1- Create a dataset with your choice: animals, plant, persons, food, activity, tools, vehicles, pedestrians, staffs, etc,…
   To create the dataset, you have 2 options as follow:
   - Images can be collected are from Google. Then, you can label them by using labelImg tools.
   - Download images and their annotations from ImagNet or Open Image Dataset.

   Note1: You can consider only single object or multiple objects for detection.
   Note2: In total, you should have at **leas**t 200 images (150 are used for training, 20 for evaluation and 30 for test). Of course, more images for training could be more effective to learn a good model but annotating them is more time consuming.
   Note3: You get **5 extra credits** for creating and using this dataset.

2- Use the defined satellite image dataset (available in Moodle)

   The data is public and it is hand annotated it from Google maps in March 2021.Source: *https://www.kaggle.com/ancaco12/aerial-satellite-images/metadata*

   The data consists of 3 object detection classes:

   – Piscina = swimming pool
   – Rotonda = traffic circle/ roundabouts
   – Parking = parking

   It consists of 180 satellite images with resolution 640x640 for training, 20 images for validation and 40 for testing.

Note4: Make sure all your images are **.jpg** images. The usage of images with other extensions may lead to problems during training.

Note5: Separate version to be used in the two options below are given.

Option 1: by using Tensorflow object detection API:

1- Train a **Faster R-CNN** detector with **ResNet101** as a backbone on your data. You first must do the following sub-tasks before training:

- You first need to download the pre-trained model faster R-CNN_ResNet101 model on COCO dataset from here.
- Then you need Modify the configuration file (faster_rcnn_resnet101_coco.config) in models/research/object_detection/samples/configs

2- Export the model

3- Test the model by modifying the notebook (object_detection_API_tutorial.ipynb)

Note: I highly recommended following the instructions in Object Detection API slides 2 for doing this project.

Option 2: by using Tensorflow Model Garden API:

1- Create a Google colab account. Upload the Dataset and Jupyter notebook (object_detection_TFM_tutorial.ipynb) to google colab.

2- Train and Time a **RetinaNet** detector with **ResNet50** as a backbone on your data.

3- Save the model to google drive and test the model using the notebook.

4- Do step 2 and 3 for **fasterrcnn** detector with **ResNet50.**

5- Compare the results of the two models and explain your findings.

Instruction for Submissions: Return a project presentation and a jupyter notebook to Moodle as **two PDF** files.

1. **Project presentation:** submit a PowerPoint project presentation (maximum 5 slides). The presentation should consist of the following parts:

- Names
- Brief description of the dataset, implementation.
- Evaluation the model on your data
- Project conclusions

Note: I will share this file to all students in the course Moodle after the project deadline.

2. **Jupyter notebook:** submit the inference notebook (object_detection_tutorial.ipynb)

Note: convert the PowerPoint and notebook to pdf files and save them with name of your dataset. For example, if you design a detector for car detection, then the name of files are “carDetection_presenation.pdf” and “carDetection_notebook.pdf”.

Assignment Evaluation: The report and notebook are evaluated on 0-20 points scale (0 means that the work is rejected).