

# Machine Learning for Agricultural Applications

## Assignment 8

Prof. Dr. Niels Landwehr  
Dr. Julian Adolphs

Summer Term 2020

Release: 25.06.2020  
Discussion: 02.07.2020

### Object detection with YOLOv3

In this assignment you will work with a pre-trained model of YOLOv3. The original code of YOLO<sup>1</sup> (<https://pjreddie.com/darknet/yolo/>) is written in C, hence we use a conversion to TensorFlow/Keras from the website: <https://machinelearningmastery.com/how-to-perform-object-detection-with-yolov3-in-keras/>. Read the tutorial and follow the instructions starting below **Create and Save Model**.

You will load a model, that is already trained. That means, for single image processing you don't need a GPU, because it takes only a few seconds on a normal CPU. Therefore, you can run it either on your local installation or on Colab.

If you work on Colab, you need to load the YOLOv3.weights data set (237 MB) to Google Drive and mount the Google Drive in Colab:

```
from google.colab import drive
drive.mount("/content/gdrive", force_remount=True)
```

For reading the weights, you need to include a path for instance like  
'/content/gdrive/My Drive/Yolo/yolov3.weights'

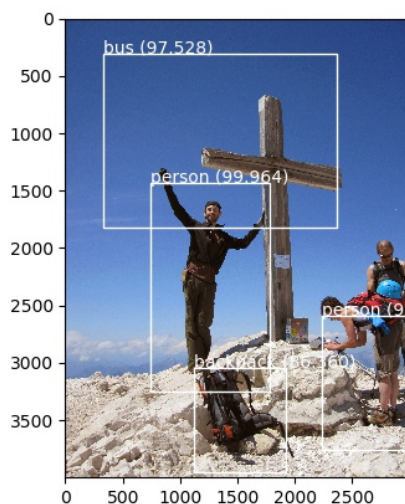


Figure 1: Object detection with Yolo. Do you see the bus? ;-)

---

<sup>1</sup>You Only Look Once: Unified, Real-Time Object Detection Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, CVPR 2016

### Task 1 – Create Model

[20 points]

Download the YOLOv3.weights to your computer or to Google Drive.

Use the `make_yolov3_model()` function to define you model. Read and load the weights, stored in the file `yolov3.weights`. Save the model in a `model.h5`-file.

How many convolutions are included in the model?

How many parameters are used in the model?

### Task 2 – Load Model and Detect

[15 points]

Now make a new python script that loads the model and applies it to the set of images linked in moodle below the exercise sheet. Save the resulting images.

Print out the coordinates of the bounding boxes  $(x_{\min}, y_{\min}, x_{\max}, y_{\max})$ .

Why are some objects not identified or miss-classified?

### Task 3 – Intersection over Union (IoU)

[15 points]

Write a python script, that calculates the Intersection over Union (IoU) of the ground throuth bounding box (yellow area in Figure 2) with coordinates  $(x_{\min}, y_{\min}, x_{\max}, y_{\max}) = (360, 237, 2262, 1752)$  and the predicted bounding box (white frame in Figure 2). Use the coordinates of the predicted frame for `img_7.jpg` from Task 2.

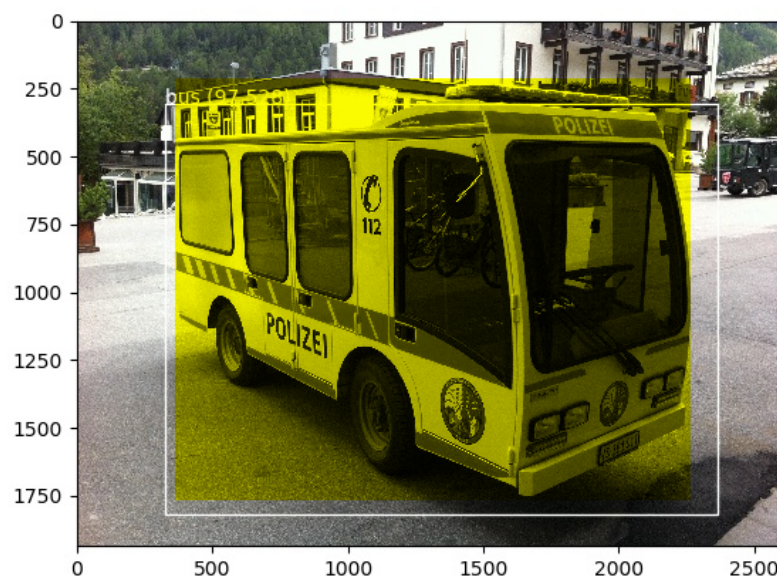


Figure 2: Ground truth bounding box (yellow area) and prediction (white frame).