High-level Agent

Classes:

- highLevelLogic.py: used to start the agent, contains the logic
- objectDetection.py: class for object detection
- models_images.py: needed for TensorFlow object detection
- testimages.py: script to run agent with multiple images from a folder

How to use highLevelLogic.py:

Run in console with `python highLevelLogic.py ARG1 ARG2`

ARG1 = image, e.g. *1.jpg* (stored in folder images)

ARG2 = json file for shelf, e.g. *B1* (stored in folder jsons)

➔ `python highLevelLogic.py 1.jpg B1`

Result is stored in folder processedimages

How to use objectDetection.py:

This class is used by the highLevelLogic.py class. Some parts of the code come from the Tensorflow Object detection tutorial (https://www.tensorflow.org/hub/tutorials/tf2_object_detection). Some parts are self-made.

From line 29 some configurations can be made. To use the high-level agent with the TF object detection, activate line 31 and 86 (path to labels and hub model).
To use another model, for example a self-trained model, use lines 30 (insert your model path), 33 (insert path to label map) and 84 (loads model with model path).

How to use testimages.py:

You can use this script to run the agent multiple times on different images. Put the images you want to use in the folder images/testdata. Name the images with ascending numbers and adjust the range in the script according to ne number of images. Result images and logs of the console are saved to images/testdata/results.

Errors:

If the following error occurs :

OSError: SavedModel file does not exist at: C:\Users\Hanna\AppData\Local\Temp\tfhub_modules\3085eb2fbe2ad0b69801d50844c97b7a7a5ecade\{saved_model.pbtxt|saved_model.pb}

Go to the folder and delete the 3085eb2fbe2ad0b69801d50844c97b7a7a5ecade folder

Small Sample Learning

- results: contains all the data that was collected during the experiments with the different SSL approaches
- workspace: contains everything needed for training an own model, based on a TensorFlow model
  followed tutorial: https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/training.html
  - training_demo: contains everything for actually training the model
  - augmentation: contains script to augment images and create corresponding xml files containing bounding boxes for each image
  - preprocessing: contains script to generate tensorflow.record from the xml files