Assignment 2: Pedestrian Detection
Deadline: 05/05/2024

The goal of the assignment is to train your own HOG-based pedestrian detector.
You will use *openCV* to extract the HOG features, and *scikit-learn* to train a SVM.

Your pedestrian detector must be trained, validated, and tested with the images included in the dataset:

WiderPerson (http://www.cbsr.ia.ac.cn/users/sfzhang/WiderPerson/).

Warning: You will use the txt files in the split_positive folder on OneDrive (see following link) to split the data in 3 sets (do not use the txt files available with the dataset otherwise you'll not be able to assess the method).

You will only use the bounding boxes corresponding to the class "*pedestrian*" (class 1) to build the set of positive samples.
To build the set of negative samples ("not a pedestrian"), download the file from

https://unipa-my.sharepoint.com/:f:/g/personal/liliana_lopresti_unipa_it/Eq8X08V-RW1PnGo4ykyZNioBDxvRYl5U3_wuOld-IBO8Uw?e=oFItiz

and use a suitable number of random windows.

You can use *cv2.HOGDescriptor()* to only extract the HOG features with parameters *winSize = (w,h)*, with w = 64 and h = 128. Use default values for all the other parameters.

Here the expected pipeline:
1. implement a procedure for data preparation (crop images based on annotated bounding boxes and resize them to the proper size in case of positive samples; random windows for negative samples);
2. extract the HOG descriptor of each sample (use *cv2.HOGDescriptor()* on the crops);
3. experiment with appropriate pre-processing procedures;
4. by using scikit-learn, train a Linear SVM (svm.LinearSVC)
5. implement a suitable *multi-scale sliding window procedure*, use your classifier to detect pedestrians in the test images at 3 different scales. Include a stride parameters to speed up

computation. You can have an idea about the scales to use by analyzing the bounding boxes in the training set.
6. implement a suitable non-maxima suppression procedure, depending on a threshold T on the IoU score. The procedure must order the candidate detection based on the classifier confidence. This value is obtained by using the method *decision_function() in scikit_learn.*

You can develop the assignment in group. A possible partition of tasks is as follows:
- student A develops 1, 2 and 3
- student B develops 5
- student C develops 6
- All: develop 4, perform and analyze the experiments

The experiments aim at selecting the best model (by using the validation set) based on the achieved **F1-score**. You must choose:
- if using or not a pre-processing procedure on the extracted HOG features;
- the best value of the hyper-parameter C in the linear SVM;
- the best value of the threshold T in the non-maxima suppression procedure.

Finally, you will use the test set to report the results of your model. You will compare your model with the default people detector in openCV

```
hog = cv2.HOGDescriptor()
hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector())
```

You will have to **report TP, FP and FN.** In computing these metrics, consider that you have a correct detection (TP) if the detection and an annotated bounding box have IoU>0.5. Otherwise, you have a wrong detection (FP). All annotated bounding boxes to which no detection corresponds are counted as missing detection (FN). In object detection, we generally do not count TN (since there are many bounding boxes that are not pedestrians). You can additionally present the precision and the recall of your method.
The assignment must be developed in a group and sent as a zip file (do not include the dataset in the zip) via email by May 05, 2024.

The zip file must include a short report (i.e., half page) describing how the work was developed and how the students in the group divided the work and collaborated. Also, include example images from the test set with the drawing of the bounding boxes found by your detector (blue), by the openCV detector (in red) and in the annotation (in green). Include a couple of images to demonstrate the success of your model and a couple of images showing failures of your model.
The report must describe the results of the experiments (use tables to summarize the results) in test and in validation (what hyper-parameters did you select? why?).