Exercise 5: MOT algorithm improvements

Objective: Extend IoU-based MOT (from Exercise 4)

5.1 Add Visual Information (Obligatory)

Instead of relying solely on geometric information (IoU) consider the visual similarity between the predicted object state and the appearance of the detected object. Incorporate additional features beyond bounding boxes, such as color histograms, texture features, or deep features into the matching process. Based on extracted characteristics, choose an appropriate similarity metric and combine it with IoU.

1. Extract visual information from image patches delimited from bounding boxes (from det.txt)
    - Integrate deep learning-based appearance embeddings, such as those obtained from a pre-trained convolutional neural network (CNN) like ResNet, MobileNet or OSNet. These embeddings can capture rich semantic information about object appearances.
    - Optionally: Incorporate also color histograms as complementary appearance features. This can enhance the tracker's ability to distinguish between objects with similar shapes but different colors.
2. Update the cost matrix based on IoU by integrating visual information with adapted similarity metric

5.2 Other ideas for improvement (add at least one additional improvement)

- Integrate more efficient deep Learning-based object detector e.g. YOLO for pedestrian detection.
- Track Management Strategies: Develop strategies for track creation and termination based on the longevity, appearance consistency, detection confidence, or other factors. Tracks that are no longer reliable or relevant can be terminated, while new tracks can be initiated.
- Efficient Search Strategies: Optimize the search for existing tracks during the matching phase. Filter out unlikely matches based on the size of the bounding boxes. For example, if an object has a significantly different size than the predicted size, it can be excluded from further consideration.
- Robust Data Association: Explore robust data association techniques beyond the Hungarian algorithm, such as the Multiple Hypothesis Tracking (MHT) algorithm.
- Handling Occlusions strategies: Implement track merging to consolidate tracklets when occluded objects become visible again. This involves associating tracklets that likely belong to the same object.
- Optical Flow Analysis: Analyze optical flow information to predict the likely positions of occluded objects. This can provide valuable cues about the object's motion even when it is not directly visible.
- Integrate Long Short-Term Memory (LSTM) network into the object tracking algorithm to predict the future motion of tracked objects.

- …. Feel free to suggest and integrate other improvements ☺

Then:

3. Evaluate the performance of our multi-object tracking system (*e.g.* tracking speed and accuracy). Use ground truth data file (gt). Compute HOTA, MOTA, IDF1, ID_Switch and Fragmentation metrics
    - Download official evaluation kit TrackEval : https://github.com/JonathonLuiten/TrackEval

For the ground truth, the 7th value (conf) acts as a flag whether the entry is to be considered. A value of 0 means that this particular instance is ignored in the evaluation, while a value of 1 is used to mark it as active.

4. Report. Provide a description and diagram blocks of the main functions of your tracking algorithm. Discuss the improvements made and challenges faced. Show some quality metrics. Compare and contrast results with and without improvements.
5. Submit your results as a *single* **.zip** *file*, including Python code (with comments), txt file with a list of libraries required for execution, and final report.