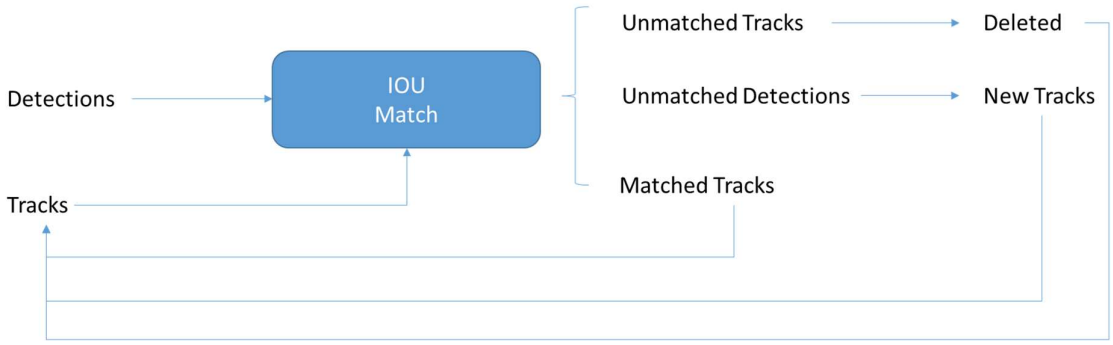
Exercise 2: Multi-object IOU Tracker (Bounding-Box Tracker)

Objective: Develop a simple IoU-based tracker without using image information. Extend the algorithm to handle multiple object tracking simultaneously.

- Object representation: bounding box
- Data: pre-generated detections loaded from text file

Before you start, download input data from MOT Challenge website. Go to the Data sidebar, then choose ADL-Rundle-6 sequence (images) corresponding text files with detections, and ground truth from MOT15 repository. You can also you can also do a download by clicking on this link. All images are in JPEG and named sequentially to a 6-digit file name (e.g. 000001.jpg). Detection and annotation files (det and gt folders) are simple comma-separated value (CSV) files.

1. Load detections (det) stored in a MOT-challenge like formatted text file. Each line represents one object instance and contains 10 values (fieldNames = [<frame>, <id>, <bb_left>, <bb_top>, <bb_width>, <bb_height>, <conf>, <x>, <y>, <z>]
   - frame = frame number
   - id = number identifies that object as belonging to a trajectory by assigning a unique ID (set to −1 in a detection file, as no ID is assigned yet).
   - bb_left, bb_top, bb_width, bb_height: bounding box position in 2D image coordinates *i.e.* the top-left corner as well as width and height
   - conf: detection confidence score
   - x,y,z: the world coordinates are ignored for the 2D challenge and can be filled with -1.
2. Implement IoU for tracking
   - Compute similarity score using the Jaccard index (intersection-over-union) for each pair of bounding boxes
   - Create a similarity matrix that stores the IoU for all boxes
3. Associate the detections to tracks in a greedy manner using IoU/ threshold *sigma_iou*. A track gets the detection with the highest intersection-over-union to its last known object position (i.e. the previous detection of the track) assigned.
4. Track management
   - Each object can be assigned to only one trajectory (ID)
   - Create and update lists for matches, unmatched detections and unmatched tracks
     - ✓Matches: IoU ≥ *sigma_iou* -> matched track
     - ✓Unmatched tracks -> delete track
     - ✓Unmatched detection -> create new tracks



5. Develop an interface for tracking results check to see if the tracker properly keeps track of objects by associating the correct IDs in the video stream
   - Draw rectangular bounding box around the detected object in images
   - Draw attributed ID to each tracked objects
   - Draw the trajectory (tracking path) in an image