**Understanding Multiple Object Tracking using Deep SORT**

## 1. Introduction to Tracking

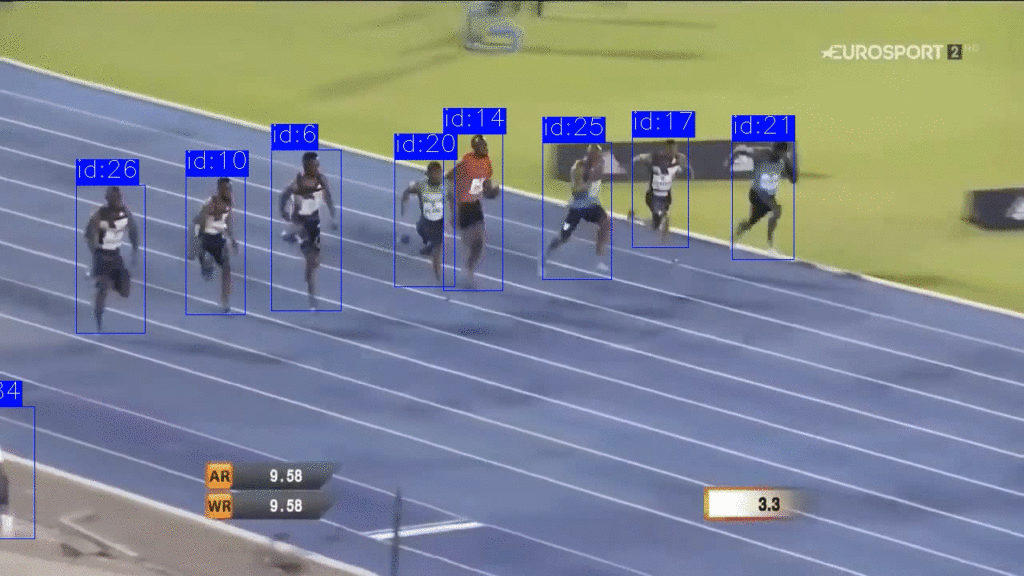
Tracking in deep learning is the task of predicting the positions of objects throughout a video using their **spatial** as well as **temporal** features. More technically, Tracking is getting the initial set of detections, assigning unique ids, and tracking them throughout frames of the video feed while maintaining the assigned ids. Tracking is generally a two-step process:

* **A detection module for target localization:** The module responsible for detecting and localization of the object in the frame using some object detector like YOLOv.x, CenterNet, etc.
* **A motion predictor**: This module is responsible for predicting the future motion of the object using its past information.

### 1.1 Need of Tracker

Some questions must be arising in your mind. Why do we even need an object tracker? Why can’t we just use an object detector? Well, you are asking the right questions. There are many reasons why a tracker is needed.

* **Tracking when object detection fails:** There are many cases where an object detector might fail. But if we have an object tracker in place, it will still be able to predict the objects in the frame. For example, consider a video where a motorbike running through the woods and we apply a detector to detect the motorbike. Here’s what will happen in this case, whenever a bike gets occluded or overlapped by a tree the detector will fail. But, if we have a tracker with it, we will still be able to predict and track the motorbike.
* **ID assignment:** While using a detector, it only showcases the location of the objects, if we just look at the array of outputs we will not know which coordinates belong to which box. On the other hand, A tracker assigns an ID to each object it tracks and maintains that ID till the lifetime of that object in that frame.
* **Real-time predictions:** Trackers are very fast and generally faster than detectors. Because of this property, Trackers can be used in real-time scenarios and has many applications in the real world.



## 2. Types of Trackers

Trackers can be classified based on many categories like methods of tracking or the number of objects to be tracked. In this section, we will see different trackers types with some examples.

### 2.1 Single and Multiple Object Trackers

* **Single Object Tracker:**

These types of trackers track only a single object even if there are many other objects present in the frame. They work by first initializing the location of the object in the first frame, and then tracking it throughout the sequence of frames. These types of tracking methods are very fast. Some of them are CSRT, KCF, and many more which are built using Traditional computer vision. However, deep learning based trackers are now proved to be far more accurate than traditional trackers. For example, SiamRPN and GOTURN are examples of deep learning based single object trackers.

* **Multiple Object Tracker:**

These types of trackers can track multiple objects present in a frame. Multiple object trackers or MOTs are trained on a large amount of data, unlike traditional trackers. Therefore, they are proved to be more accurate as they can track multiple objects and even of different classes at the same time while maintaining high speed. Some of the algorithms include DeepSORT, JDE, and CenterTrack which are very powerful algorithms and handle most of the challenges faced by trackers.

### 2.2 Tracking by Detection and without Detection

* **Tracking by Detection:**

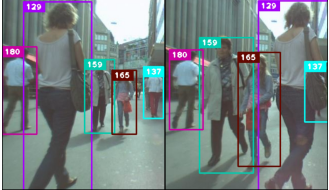
The type of tracking algorithm where the object detector detects the objects in the frames and then perform data association across frames to generate trajectories hence tracking the object. These types of algorithms help in tracking multiple objects and tracking new objects introduced in the frame. Most importantly, they help track objects even if the object detection fails.

* **Tracking without Detection:**

The type of tracking algorithm where the coordinates of the object are manually initialized and then the object is tracked in further frames. This type is mostly used in traditional computer vision algorithms as discussed earlier.

## 3. Introduction to DeepSORT

DeepSORT is a computer vision tracking algorithm for tracking objects while assigning an ID to each object. DeepSORT is an extension of the SORT (Simple Online Realtime Tracking) algorithm. DeepSORT introduces deep learning into the SORT algorithm by adding an appearance descriptor to reduce identity switches, Hence making tracking more efficient.

[](https://learnopencv.com/wp-content/uploads/2022/06/02-person-tracking.png)

### 3.1 Simple Online Realtime Tracking (SORT)

SORT is an approach to Object tracking where rudimentary approaches like Kalman filters and Hungarian algorithms are used to track objects and claim to be better than many online trackers. SORT is made of 4 key components which are as follows:

**Detection:** This is the first step in the tracking module. In this step, an object detector detects the objects in the frame that are to be tracked. These detections are then passed on to the next step. Detectors like FrRCNN, YOLO, and more are most frequently used.

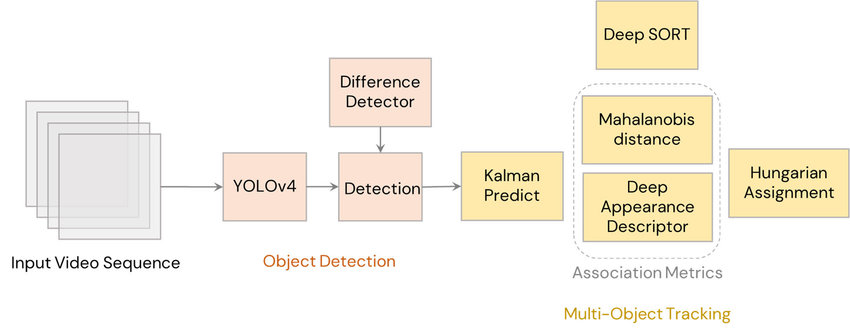
**Estimation:** In this step, we propagate the detections from the current frame to the next which is estimating the position of the target in the next frame using a constant velocity model. When a detection is associated with a target, the detected bounding box is used to update the target state where the velocity components are optimally solved via the Kalman filter framework

**Data association:** We now have the target bounding box and the detected bounding box. So, a cost matrix is computed as the [intersection-over-union (IOU)](https://learnopencv.com/intersection-over-union-iou-in-object-detection-and-segmentation/) distance between each detection and all predicted bounding boxes from the existing targets. The assignment is solved optimally using the Hungarian algorithm. If the IOU of detection and target is less than a certain threshold value called IOUmin then that assignment is rejected. This technique solves the occlusion problem and helps maintain the IDs.

**Creation and Deletion of Track Identities:** This module is responsible for the creation and deletion of IDs. Unique identities are created and destroyed according to the IOUmin. If the overlap of detection and target is less than  IOUmin then it signifies the untracked object. Tracks are terminated if they are not detected for TLost frames, you can specify what the amount of frame should be for TLost. Should an object reappear, tracking will implicitly resume under a new identity.

The objects can be successfully tracked using **SORT** algorithms beating many State-of-the-art algorithms. The detector gives us detections, Kalman filters give us tracks and the Hungarian algorithm performs data association.

### 3.2 DeepSORT



SORT performs very well in terms of tracking precision and accuracy. But **SORT** returns tracks with a high number of ID switches and fails in case of occlusion. This is because of the association matrix used.

**DeepSORT** uses a better association metric that combines both motion and appearance descriptors. **DeepSORT** can be defined as the tracking algorithm which tracks objects not only based on the velocity and motion of the object but also the appearance of the object.

For the above purposes, a well-discriminating feature embedding is trained offline just before implementing tracking. The network is trained on a large-scale person re-identification dataset making it suitable for tracking context. To train the deep association metric model in the **DeepSORT** cosine metric learning approach is used.

According to **DeepSORT’s** paper, “The cosine distance considers appearance information that is particularly useful to recover identities after long-term occlusions when motion is less discriminative.” That means cosine distance is a metric that helps the model recover identities in case of long-term occlusion and motion estimation also fails. Using these simple things can make the tracker even more powerful and accurate.