Object Tracking

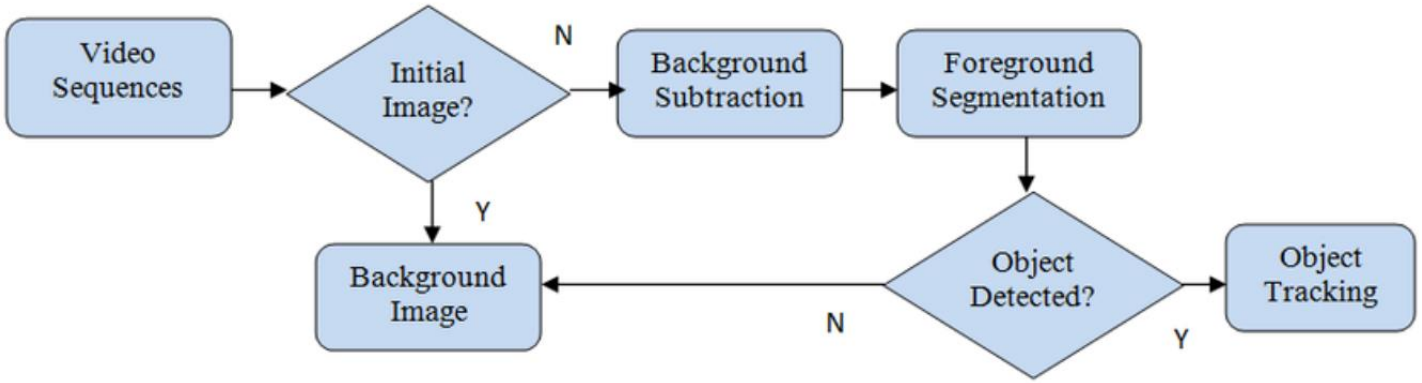
Objectives

- What is object tracking?
- The techniques used in object tracking.
- What is a future of object traking?
- Applications of object tracking

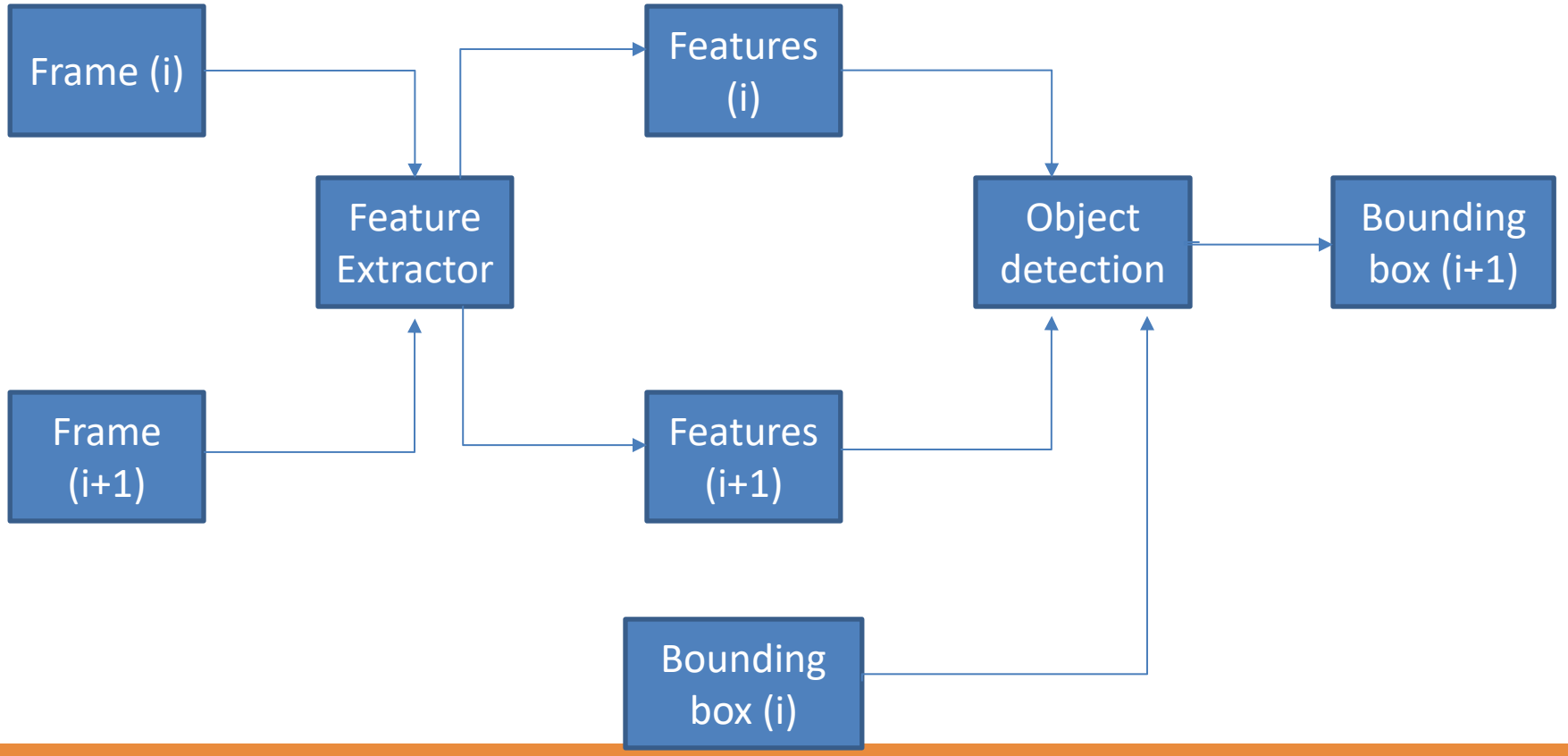
Object Tracking

What is Object Tracking?

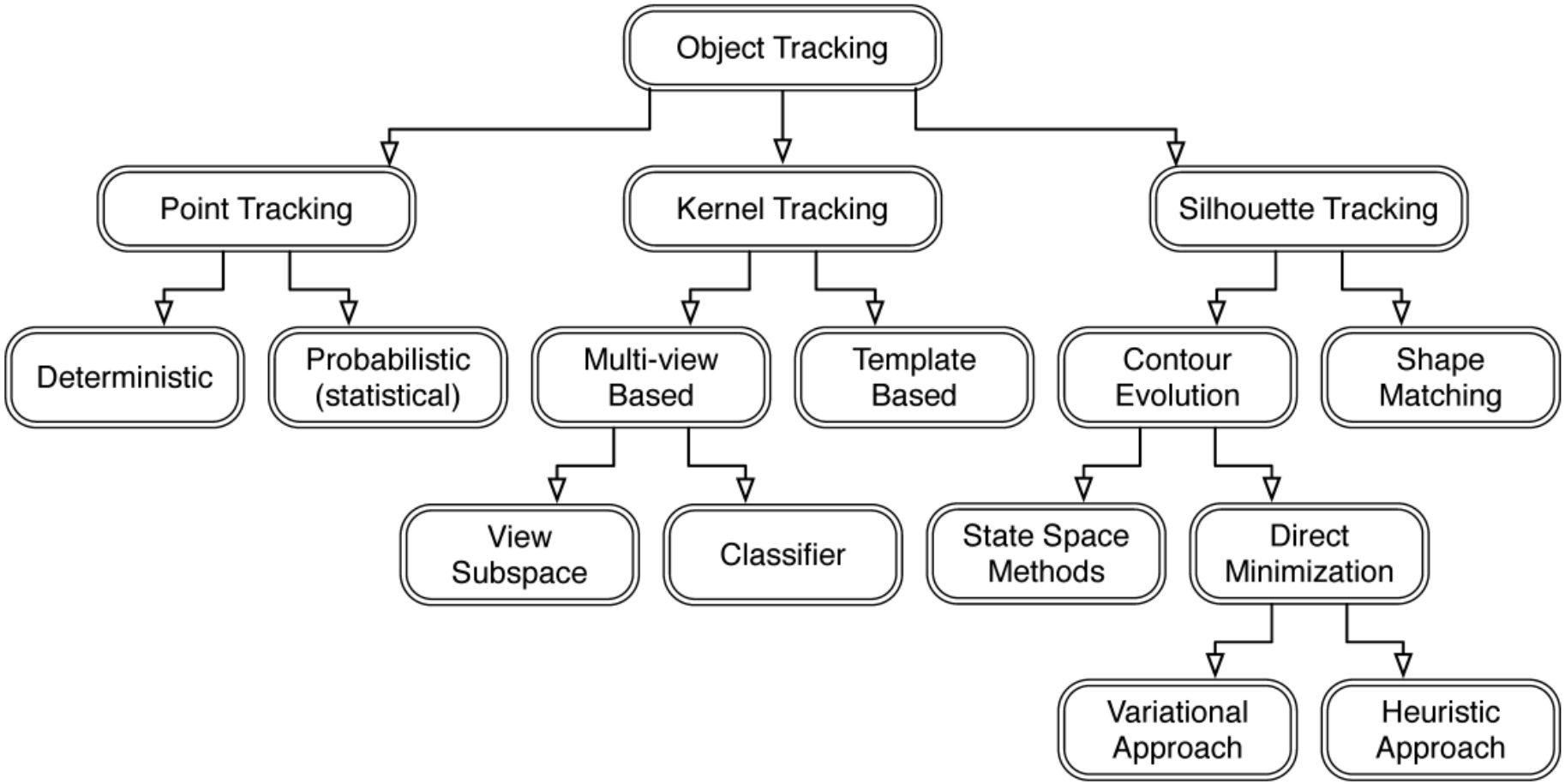
- Object tracking is the task of taking an initial set of object detections, creating a unique ID for each of the initial detections, and then tracking each of the objects as they move around frames in a video, maintaining the ID assignment.



What is Object Tracking?



The Approach

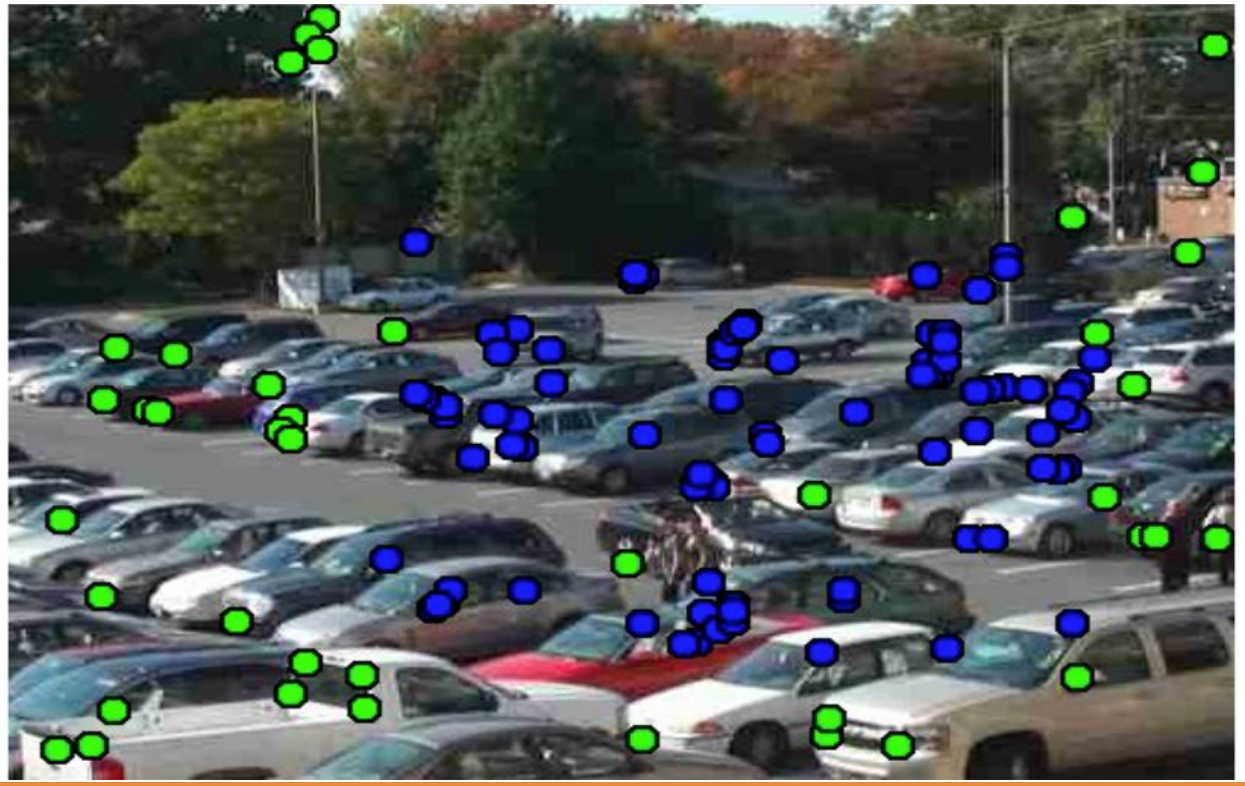


The Approach

- Type 1: Point Tracking: Objects detected in consecutive frames are represented by points: deterministic, probabilistic methods, and the association of the points is based on the previous object state which can include object position and motion. This approach requires an external mechanism to detect the object in every frame.

The Approach

- Point Tracking: Tracked point features in an image sequence. Blue dots are older tracks and green dots are newly spawned tracks.



The Approach

- Type 2: Kernel Tracking: kernel tracking is usually performed by locating the moving object, which is represented by an embryonic object region, from one frame to the next: multi- view based, template based. It refers to the object shape and appearance.

The Approach

- Kernel Tracking: Kernel based Object Tracking using Colour Histogram Technique.

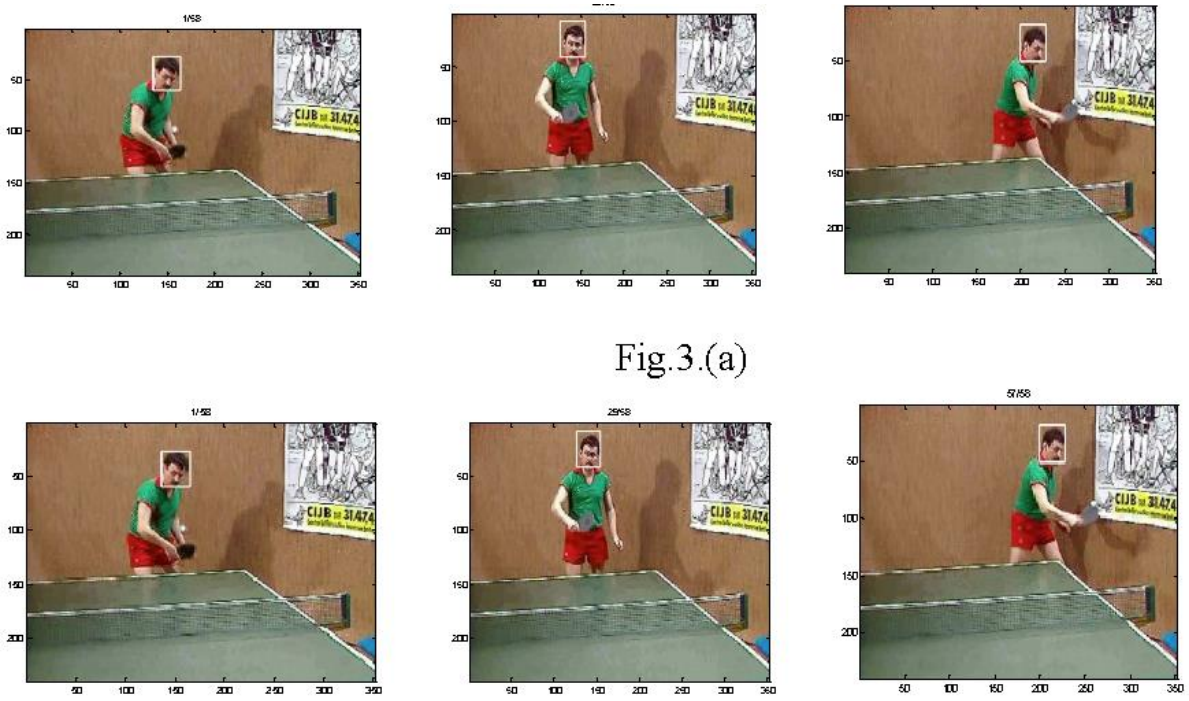


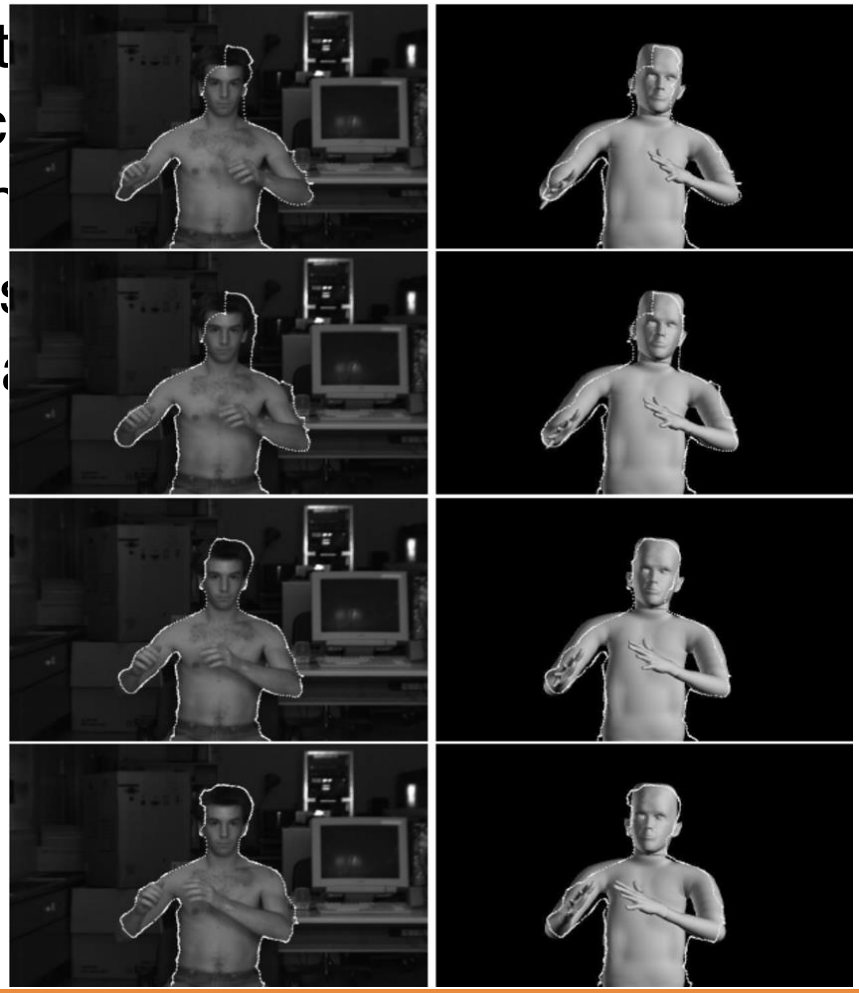
Fig.3.(a)

The Approach

- Type 3: Silhouette Tracking: Tracking is performed by estimating the object region in each frame. Silhouette tracking methods use the information encoded inside the object region. Ex: contour evolution, shape matching

The Approach

- Silhouette Tracking:
  - Tracking results using t
    extraction method of Sec
    and 100 of the sequence ir
  - Snake- optimized contours
    images on the left as well a
    right



Object tracking levels

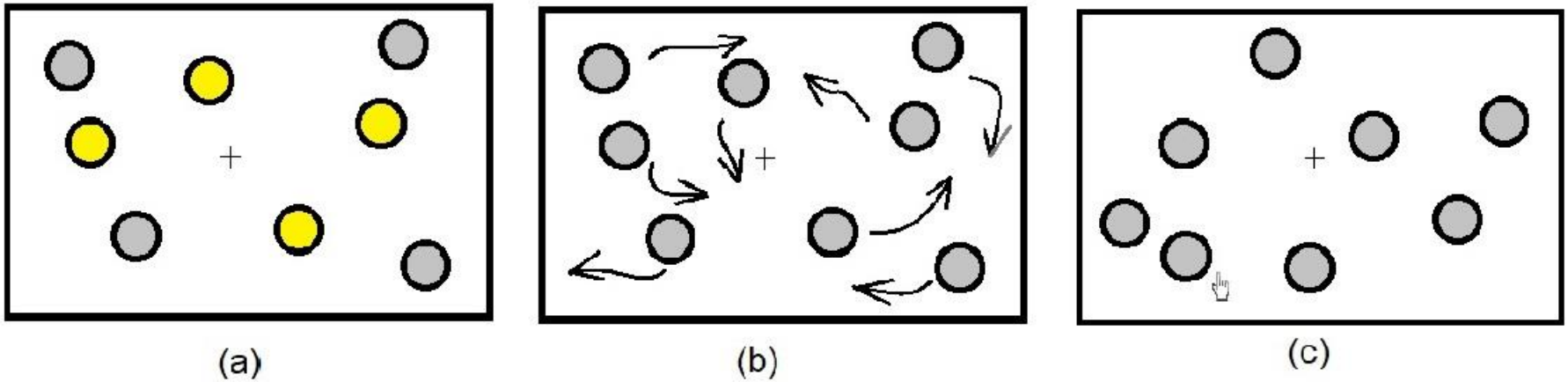
- Type 1: Single Object Tracking (SOT)
  - It creates bounding boxes that are given to the tracker based on the first frame of the input image.
  - Single Object Tracking is also sometimes known as Visual Object Tracking.
  - SOT implies that one singular object is tracked, even in environments involving other objects.

Object tracking levels

- Single Object Tracking (SOT)

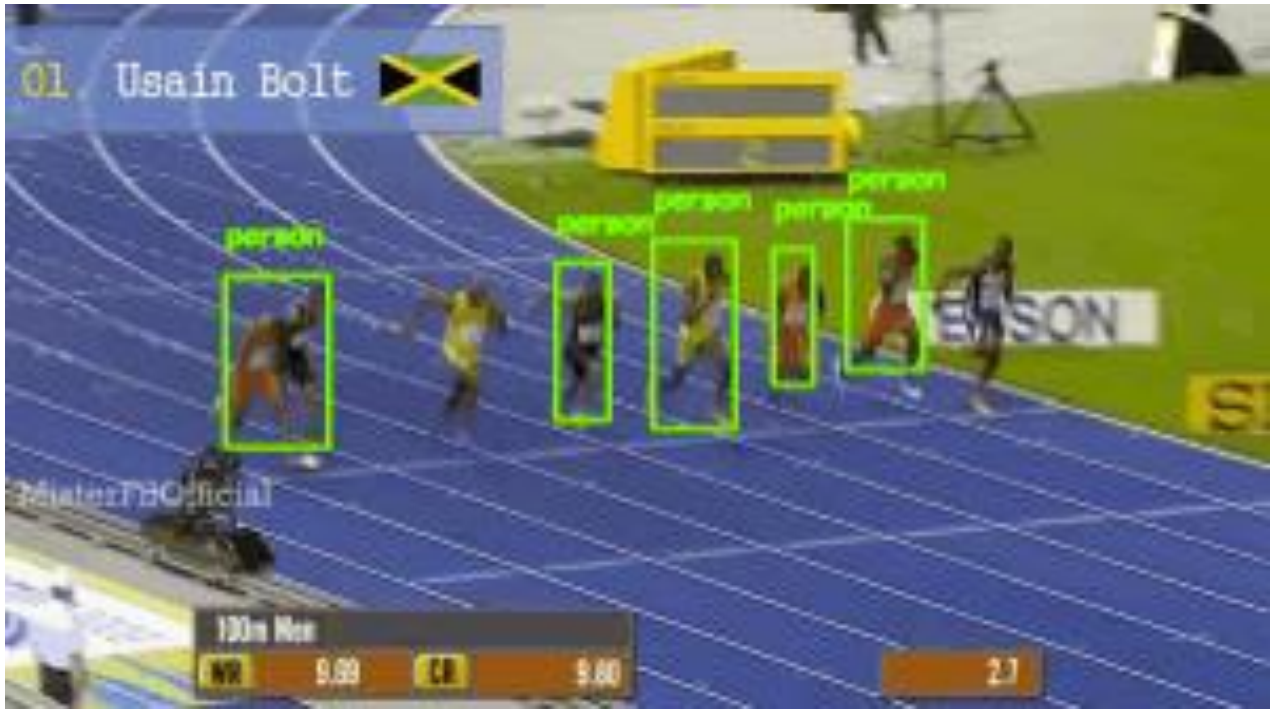
Object tracking levels

- Type 2: Multiple Object Tracking, or MOT
  - It is an experimental technique used to study how our visual system tracks multiple moving objects.



Object tracking levels

- Multiple Object Tracking (MOT)



Object tracking Algorithms

FPT Education
FPT UNIVERSITY

– OpenCV:

- CSRT: is best when the user requires a higher object tracking accuracy and can tolerate slower FPS throughput
- KCF tracker: is not as accurate compared to the CSRT but provides comparably higher FPS
- The MOSSE tracker is very fast, but its accuracy is even lower than KCF.

– DeepSORT: One of the most widely used, object tracking framework is Deep SORT, an extension to SORT (Simple Real time Tracker).

– MDNet is fast and accurate, CNN- based visual tracking algorithm inspired by the R-CNN object detection network

Object tracking Algorithms

- Optical Flow: Optical flow, or motion estimation, is a fundamental method of calculating the motion of image intensities, which may be ascribed to the motion of objects in the scene.
- Meanshift: Mean Shift is a non-parametric iterative algorithm that can be used for a lot of purposes like finding modes, clustering
- Kalman Filters: The Kalman filter for tracking moving objects estimates a state vector comprising the parameters of the target, such as position and velocity, based on a dynamic/measurement model

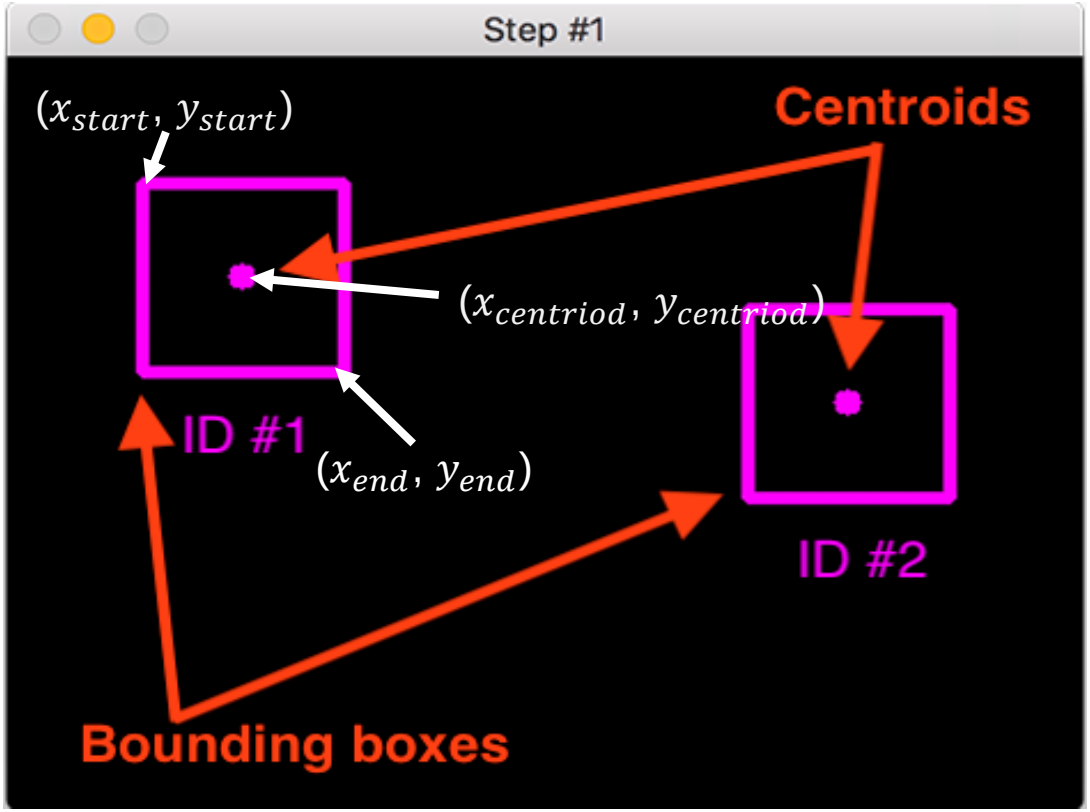
Object tracking

- Object tracking is the process of:
  - Taking an initial set of object detections (such as an input set of bounding box coordinates)
  - Creating a unique ID for each of the initial detections
  - And then tracking each of the objects as they move around frames in a video, maintaining the assignment of unique IDs

The centroid tracking algorithm

- **Tracking object**
  + Step 1: Compute centroids from box coordinates

$$(x_{centriod}, y_{centriod}) = (\frac{x_{end}-x_{start}}{2}, \frac{y_{start}-y_{end}}{2})$$



The centroid tracking algorithm

- **Tracking object**

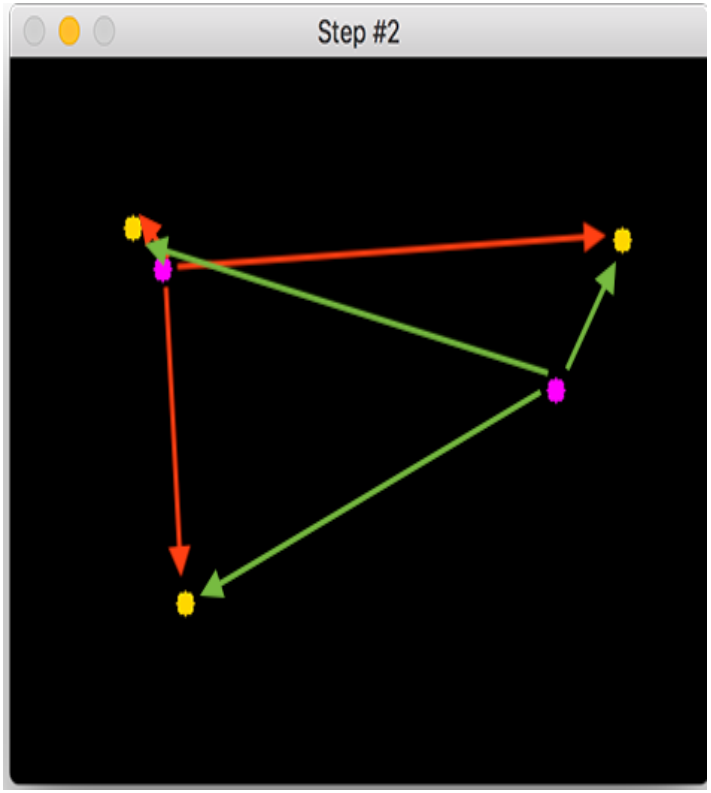
+ Step 1: Compute centroids from box coordinates

$$(x_{centriod}, y_{centriod}) = (\frac{x_{end}-x_{start}}{2}, \frac{y_{start}-y_{end}}{2})$$

+ Step 2: Compute distance between new and existing objects

Euclidean formula: 2 point $p(p_1, p_2), q(q_1, q_2)$

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$$



The centroid tracking algorithm

- **Tracking object**

+ Step 1: Compute centroids from box coordinates

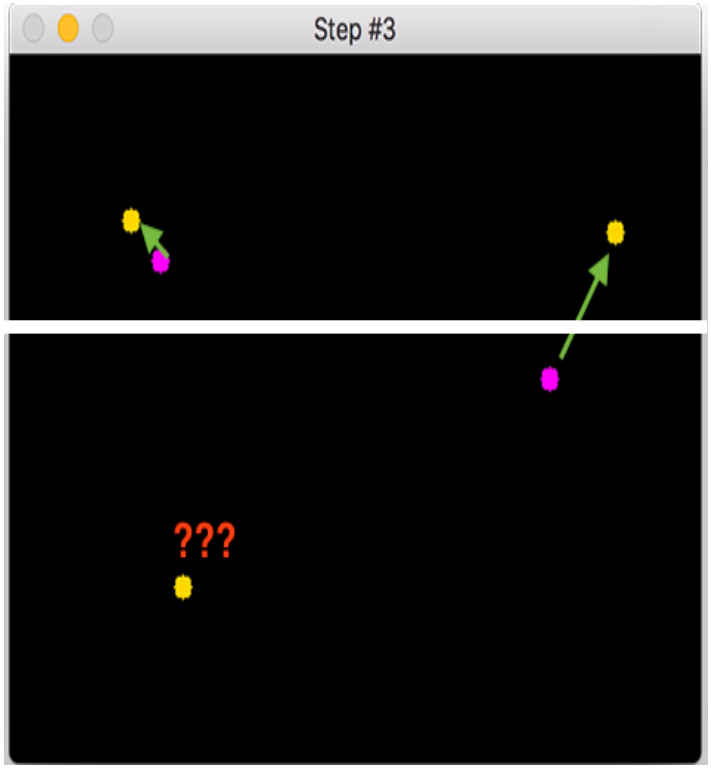
$$(x_{centriod}, y_{centriod}) = (\frac{x_{end}-x_{start}}{2}, \frac{y_{start}-y_{end}}{2})$$

+ Step 2: Compute distance between new and existing objects

Euclidean formula: 2 point $p(p_1, p_2), q(q_1, q_2)$

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$$

+ Step 3: Update *(x, y)*-coordinates of existing objects



The centroid tracking algorithm

- **Tracking object**

+ Step 1: Compute centroids from box coordinates

$$(x_{centriod}, y_{centriod}) = (\frac{x_{end}-x_{start}}{2}, \frac{y_{start}-y_{end}}{2})$$

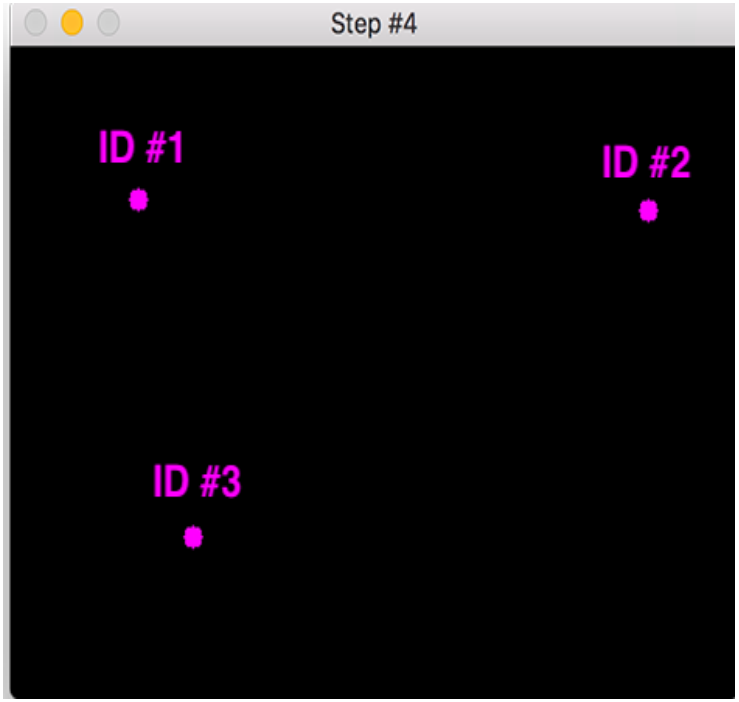
+ Step 2: Compute distance between new and existing objects

Euclidean formula: 2 point $p(p_1, p_2), q(q_1, q_2)$

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$$

+ Step 3: Update *(x, y)*-coordinates of existing objects

+ Step 4: Register new objects



The centroid tracking algorithm

- **Tracking object**

+ Step 1: Compute centroids from box coordinates

$$(x_{centriod}, y_{centriod}) = (\frac{x_{end}-x_{start}}{2}, \frac{y_{start}-y_{end}}{2})$$

+ Step 2: Compute distance between new and existing objects

Euclidean formula: 2 point $p(p_1, p_2), q(q_1, q_2)$

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$$

+ Step 3: Update *(x, y)*-coordinates of existing objects

+ Step 4: Register new objects

+ Step 5: Deregister old objects

Multiple Object Tracking

- Step 1: Create a Single Object Tracker
  - A multi-object tracker is simply a collection of single object trackers.
- Step 2: Read First Frame of a Video
  - A multi-object tracker requires two inputs
    - A video frame
    - Location (bounding boxes) of all objects we want to track.
  - Given this information, the tracker tracks the location of these specified objects in all subsequent frames.
- Step 3: Locate Objects in the First Frame
  - Next, we need to locate objects we want to track in the first frame. The location is simply a bounding box.

Multiple Object Tracking

- Step 4: Initialize the MultiTracker
  - Until now, we have read the first frame and obtained bounding boxes around objects. That is all the information we need to initialize the multi-object tracker.
- Step 5: Update MultiTracker & Display Results
  - Finally, our MultiTracker is ready and we can track multiple objects in a new frame. We use the update method of the MultiTracker class to locate the objects in a new frame. Each bounding box for each tracked object is drawn using a different color.

Applications

- Behavioral analysis of objects or human, such as gesture analysis, facial expression analysis.
- Robotics such as autonomous vehicle
- Smart camera such as video surveillance

Object tracking challenges

- Occlusion: It occurs when an object we are tracking is hidden by another object.
- Scale change
- Background clutter: Background near object has similar color or texture as the object.
- Appearance change: Different viewpoints of an object may look very different visually and without context.
- ➔ The main challenge is to hold balance between computational efficiency and performance.

Summary

- What is Object Tracking?
- The techniques used in Object Tracking.
- What is a future of object tracking?
- Applications of Object tracking