Multi-Object Tracking (MOT) with Deep Learning

Suvrat Bhooshan, Aditya Garg

Introduction

Goal: Track and Tag Multiple Objects (people) in a video stream using Deep Learning models.

- **Motivation:** Traditionally, tracking has relied on geometric algorithms to correlate objects between frames, and is largely an unexplored territory for deep learning.
- **Objective:** Create our own novel deep learning tracking algorithm, assuming perfect object detection.
- **Approach:** Given the objects in a crowded video frame, our model would learn the positions, velocity and image features of each object in the video stream and continue to track it across multiple frames.
- **Applications:** With known applications in security and surveillance like theft prevention, traffic violations, object tracking can also be used commercially for applications like inventory management, automated checkout etc.

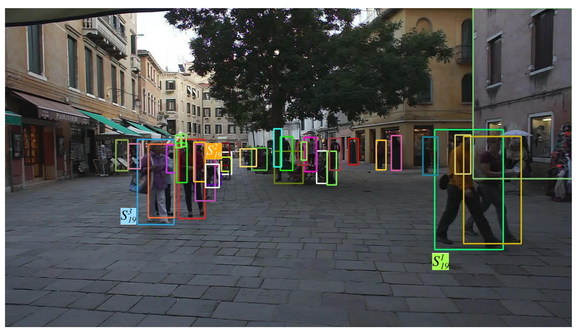
Problem Statement

MOT is a multi-variable estimation problem.

Given: For a sequence of frames: $S=\{S_1,S_2,S_3....,S_t\}$

For each frame: $S_t=\{s_t^1,s_t^2,s_t^3...,s_t^m\}$, where t is the video frame in consideration and m is number of objects detected in the frame.

Each s_t^i represents the state of the object i in frame t in the form of the bounding box coordinates for the object in the frame.



S_{19} : Frame 19 from the MOT16-02 Dataset

Target: Tracking requires extracting $s_{i_s:i_e}=\{s_{i_s}^i,s_{i_s+1}^i,...,s_{i_e-1}^i,s_{i_e}^i\}$ that is find the sequential frames for object i from when it enters the frame (i_s) to when it exits (i_e).

Evaluation: Based on error metrics defined on the right.

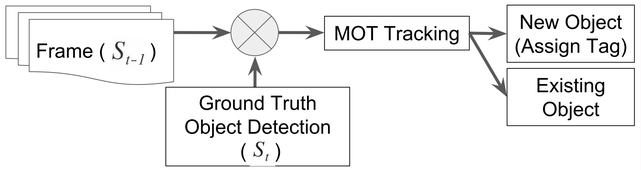
Datasets

We are using the dataset from MOT challenge 2016 which contains 14 videos (7 training, 7 test) in unconstrained environments. The videos used for training & testing have following metrics:

- Frames per Second (FPS) : 30
- Resolution: 1920 x 1080
- Total Training Frames: 3579 frames
- Total Testing Frames: 4725 frames

Approach & Algorithms

- **Detection Based Tracking**



- **Online Tracking:**
  - Involves a real-time approach where we gradually extend the trajectory of the object.
  - On the contrary, an offline model would look at the entire sequence in a go and create a trajectory for the object across all the sequences.

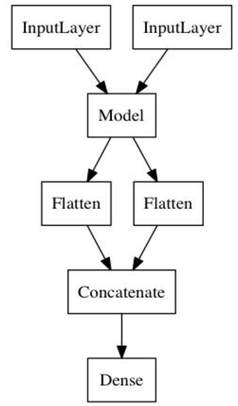
- **k-Nearest Neighbor**
  - Assigns current object to the object with the highest Intersection over Union (IoU) area overlap in the previous frame.
  - If IoU area overlap is below a threshold (80%) for all object permutations, tag it as a new object.

- **LSTM based Tracker**
  - Pass the bounding box coordinates for the previous 9 frames (timesteps), and the new bounding box as the 10th timestep.
  - The intuition behind modelling the data as a time-series for a LSTM is that the LSTM will learn the velocity and the direction of motion from the bounding box coordinates, and infer if the new bounding box can belong to the existing trajectory or not.
  - If no object maps to an existing objects with over 60% confidence, assign new object id, else map to existing objects.

Approach & Algorithms

Siamese Network:

- The VGG CNN model initialized from ImageNet is used to learn image representations.
- The Siamese network passes in the object from the current frame, and an object from previous frame through the CNN model.
- A softmax classifier outputs 1 if the two objects are the same, 0 otherwise.

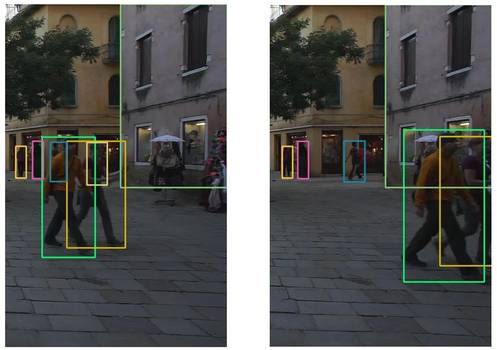


Error Metrics

- **Completeness**
  - A correlation between the number of ground truth objects in each frame and the number of detected objects for each frame.
  - Since we use ground truth for detection, our completeness has 100% accuracy, that is we identify and tag the same number of objects as the ground truth.
- **Accuracy**
  - **ID Switches**
    - Compared to ground truth, count the number of times the Object Id (Tag) for the object changes.
    - Normalize this by dividing by sum of total frames of each object. **(Lower is better)**
    - Intuition: How does model perform on consecutive frames? How often does it switch person A and person B?
  - **Mismatch Rate**
    - Calculate the maximum number of frames, object is assigned the same id.
    - Normalize it by dividing by sum of total frames of each object. **(Higher is better)**
    - Intuition: How long does the model track the same person?

Results

Frame 1 to Frame 40 Crop of Tracking



Modal	ID Switched	Mismatch Rate
KNN	0.03	0.78
LSTM	0.67 (Preliminary)	0.30 (Preliminary)
Siamese	97% Model Acc. (Test Pending)	

Future Work

- Account for object occlusion by:
  - Training the model to predict bounding boxes.
  - Using a combination of object based and object free detection.
- Connect pipeline to an existing object detection model instead of using ground truth for detection.
- Create better visualizations to understand what all the model learnt (velocity, image features, trajectory etc.)

References

[1] Luo, Wenhan, et al. "Multiple object tracking: A literature review." *arXiv preprint arXiv:1409.7618* (2014).

[2] Bernardin, Keni, Alexander Elbs, and Rainer Stiefelhagen. "Multiple object tracking performance metrics and evaluation in a smart room environment." *Sixth IEEE International Workshop on Visual Surveillance, in conjunction with ECCV*. Vol. 90. 2006.

[3] "MOT Challenge." *MOT Challenge*. N.p., n.d. Web. 05 June 2017. <https://motchallenge.net/>.