Intro to Image Understanding (CSC420)
Assignment 5

Posted: Nov 20, 2015 Submission Deadline : Nov 28 (Sat), 11.59pm, 2015

Max points: 12, max extra credit points: 1

(1) You are given a short video clip broken down into consecutive frames. For each frame you are also provided with a set of detection boxes for the *person* class obtained via the Deformable Part-based Model (DPM). The data is structured in the following way:

- The frames are in DATA/FRAMES
- The detections are in DATA/DETECTIONS. Each frame has its own mat file which contains a variable DETS. This variable has the same structure as DS in DEMO_CAR function in Assignment4. Thus, each row of DETS is $[\text{left}, \text{top}, \text{right}, \text{bottom}, \text{id}, \text{score}]$, specifying the coordinates of the detection box, and its confidence (score).

Your task is to complete a function called CODE/TRACK_OBJECTS.M (Matlab users) or CODE/TRACK_OBJECTS.PY (Python users). This function loads detections of two consecutive frames, and computes similarity between each detection in frame i and each detection in frame $i+1$. Your goal is to find an assignment between detections in consecutive frames of the videos, that are believed to correspond to the same, possibly moving, object. The assignment between detections in multiple frames is called a *track*. There are several different options of how to do that, and you should pick one (you can implement both, but only one will count in your grade):

(a) **(3 points)** Greedy method: Initialize tracks as an empty list. Visit each frame. Assign two detections (one in the current and one in the next frame) with the highest similarity to the same track and add the track to the track list. Pick the next two most similar detections (one in the current and one in the next frame, both detections should be disjunct from the already selected pair), add them to the next track, etc. When no more disjunct pairs of detections exist, move on to the next frame. Again pick two detections (one in the current and one in the next frame) with the highest similarity. If one of the tracks in the existing track list contains any of the picked detections, then just assign the new detection to this track. If not, add the new pair as a new track. Pick the next pair of most similar detections, etc. Once you visit all the frames, remove tracks that contain only two detections (this is probably noise).

(b) **(5 points)** Dynamic Programming: Find the best path across detections in all frames. The path consists of a detection in the first frame, a detection from the second frame, etc. The best path is the path with:

$$\max_{i_1 \in F_1, i_2 \in F_2, \dots i_n \in F_n} (\text{sim}(det_{i_1}, det_{i_2}) + \text{sim}(det_{i_2}, det_{i_3}) + \cdots + \text{sim}(det_{i_{n-1}}, det_{i_n}),$$

where F_j is the set of all detections in the j-th frame, and det_{i_j} is a detection in the j-th frame. One path represents one object track. Remove the path from the list. Find the next best path.

Note that a greedy solution is easier and can only give you up to 11 total points for the assignment (assuming all other exercises are done correctly). A perfect dynamic programming solution can lead up to 12 points plus 1 extra credit point.

In your solution document, include a short explanation of your method. Include also code (the completed TRACK_OBJECTS function) plus any other function you may have written for this purpose.

(3) **(2 points)** Visualize your solution. Plot each frame with all detections (boxes) that have been tracked for more than 5 frames. Each different track should have a rectangle (box) of different color, however all detections corresponding to the same track should have the same color. Store a visualization of each frame in a directory called TRACKS. You do not need to insert the frames into your solution document. Simply include the TRACKS directory in a zip file along with your document and upload to CDF.

(4) **(1 point)** Do not worry if you don't track all the players. Some of them may not be detected in all the frames. Any idea how to deal with missing detections? No need to write code, just write down your idea.

(5) **(1 point)** Among all your tracks how would you find the soccer player that was running the fastest? Be careful with your answer: a player very far away seems to move less in an image, but the player may in fact have been running like Flash. You do **not need to provide any code**, just a written answer will do.

(6) **(1 point)** How would you find a circle of an unknown radius in an image? Don't write code, a written answer will do.

(7) **(1 point)** How does a Hough voting type of approach to object detection (e.g., to detect a car) work? Don't write code, a written description will do. Hough voting was used roughly 10 years ago – what is the current best known strategy to detect objects?

(8) **(1 point)** What was your favorite topic in the course? Why?

(9) **(1 point)** In your opinion, what are the most important computer vision problems that need to be solved in order to enable a "cognitive" home robot, that will for example help your grandma wash the dishes, walk the dog, make sure she eats well, etc?