CS556: HOMEWORK 1
due 02/03/2017 by 12pm

Please submit your homework as a **single compressed folder** (e.g., zip file) to the TEACH website:

`https://secure.engr.oregonstate.edu:8000/teach.php?type=want-auth`

HW1 is about tracking an object in a given sequence of video frames. This can be done by finding matches of interest points in consecutive frames. The video frames, and ground-truth locations of bounding boxes of the object in each frame are provided on the class website (taken from `http://cmp.felk.cvut.cz/~vojirtom/dataset/`). Each ground-truth bounding box is specified by the coordinates of its top-left and bottom-right points: [topleftX, topleftY, bottomRightX, bottomRightX] in file gt.txt. HW1 consists of the following computational steps.

Step 1 – 30 points

In every image:

1) Detect at least 100 (strongest) interest points of your choice in each frame. For example, you may detect Harris corners, or other interest points that we have not covered in class (e.g. SURF).
2) Compute feature descriptors of your choice for every detected interest point. For example, you may use SIFT or SURF feature descriptors.
3) Plot top 100 strongest interest points superimposed onto the original image of the following three frames: img0001.jpg, img0050.jpg, img0100.jpg; Upload these three visualizations of your interest-point detection.

Step 2 – 40 points

Track the object – namely, the Transformer – in the given video sequence, starting from frame img0000.jpg, shown in Figure 1. Assume that we are given the initial bounding box around the Transformer in img0000.jpg as [190,52,341,264]. Track the Transformer using the following steps:

1) Match your interest points between every two consecutive video frames. You may use any method for point matching of your choice. For example, using MATLAB's built-in function *matchFeatures()*, or Hungarian algorithm (open-source MATLAB code available online).

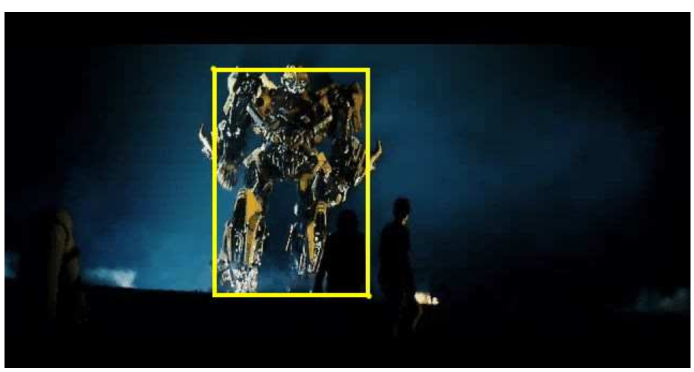

Fig. 1. An example of the ground-truth bounding box around the Transformer. The bounding box is specified by the coordinates of its top-left and bottom-right points: [topleftX, topleftY, bottomRightX, bottomRightX]

2) Starting from the given bounding box [190,52,341,264] of img0000.jpg, use the point matches to the subsequent frame img0001.jpg for estimating the optimal location of the bounding box in img0001.jpg. Continue this estimation along the entire video sequence.
3) Save your estimated bounding-box locations for all frames in the file: estimation.txt. Upload this file.

Step 3 – 25 points

Compute the Euclidean distances between your estimated and ground-truth locations of bounding boxes across the video frames. Specifically, for each frame, compute the Euclidean distance between the centers of the estimated and ground-truth bounding boxes. Save your results for all frames, and the average Euclidean distance across all frames in the file: error.txt. Upload this file.

Code – 5 points

Print out ONLY parts of the code that you wrote for implementing the above Step 1, Step 2, and Step 3, and upload this print-out.