Assignment 3
Due May 5, 2020 at 11.59 PM via ilearn Assignment-3 Submission folder

IMAGES. All images may be found in the TEST IMAGES folder on iLearn or in the folder uploaded specific to this assignment.

Independent Reading. (i) Normalized Cuts for Segmentation:
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=868688
(ii) Active Contours: https://link.springer.com/article/10.1007/BF00133570

Problem 1. [3 pts] On the `house.tif` and `peppers_color.tif` image in the TEST IMAGES folder, implement and apply a basic version of normalized cuts for segmenting the image in MATLAB. Visually analyze the results and comment on the accuracy of the results. Focus on implementing the basic mathematical steps properly.

Problem 2. [3 pts] On the `peppers_color.tif` image, implement the Expectation Maximization algorithm for mixture of Gaussian model based on color features for segmenting the image in MATLAB. Again, visually analyze the results and comment on the accuracy of the results. Focus on implementing the basic mathematical steps properly.

Problem 3. [4 pts] For this problem, you will be using the `atrium.mp4` video clip available in MATLAB. This video is recorded from a static over-head camera and contains people walking in a lobby. You need to detect the persons, extract features for the detected regions and then apply data association to obtain correspondence between persons from frame t to $t+1$. After the data association step, an optimization step is applied to obtain the tracklets of persons, but you do not need to implement it in this problem. A sample main code named `dataAssociationTracking.m` and `deepDataAssociationTracking.m` is provided for this problem. You should NOT modify these code.

Note. You need to install `torchreid` package from torchreid for deep features. Please note that feature extraction using deep network will be slower than HoG features.

These code calls the following functions, which you need to implement.

(a) It first loads the video and extracts the frames from it. For each frame, it calls the following functions.

(b) (1 pts) As the video is recorded from a static camera, a simple sum of difference between N frames will be able to highlight the moving objects. The `getSumOfDiff.m` function is called for this purpose with $N = 3$. Given a stack of N frames, it should return the following

$$D(m,n) = \frac{1}{\binom{N}{2}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} |I_i(m,n) - I_j(m,n)|$$

where I_i represent the i^{th} image in the stack. You need to implement this function.

(c) (2 pts) The sum of difference image is then used by the `getDetections.m` function to obtain the detections. This function should segment the blobs which are highlighted in the sum of difference image. You need to implement this function. It should return the bounding box details [topleft_x,topleft_y,width,height] of the segmented blobs. You may implement any algorithm or use built-in functions for segmentation, morphological operations, noise removal and regionprops for the problem.

(d) **Feature Extraction**

- *HoG Features.* The features are extracted from the bounding box regions using the `getFeatures.m` function. You can use the HoG features implemented in the previous assignment.

- *Deep Features.* The features are extracted from the bounding box regions using the `getFeatures.m` function. This code executes `extractDeepFeatures.py` script to extract deep feature. You need to complete this code.

(e) **Feature Matching**

- All the above steps are repeated for the next frame. Then, the `getMatches.m` and code is used to obtain the correspondences between the detected regions in the two frames. You can reuse the same function implemented in the previous assignment. Please choose the threshold as required. [Refer to Assignment-2 for mode details.]
- For deep features you need to write `getDeepMatches.m` to obtain correspondences between detected region. Use nearest neighbour feature to match detected regions. Please choose the threshold as required.

(f) Finally, the detections and the matched bounding box centroids are displayed in the sample code. Use `dataAssociationTracking.m` for HoG featrures and `deepDataAssociationTracking.m` for deep feature.

After completing all the above required functions, observe the detections and data associations using HoG and deep features.

Submission Protocol. All codes should be written in MATLAB. You should submit codes as well as explanations to each problem (if required). You should add comments to your codes to make them reader friendly. All coding related problems should be in separate scripts (.m files) named after the problem number. If you may require to call functions for a problem, you may do so, but include them in your submission. Keep all the images necessary to run a code in the same folder as the code, while you are submitting. You MUST also include a report written electronically (using the likes of LaTeXor MS Word). It should contain explanations, images, etc (as required).

Each student must do the assignment independently, although you may discuss prior to that. While discussion is allowed, we will be particularly careful about any plagiarism, whether from each other or from other sources.