Mean Shift HW

1. Implement mean-shift in Python
2. Implement CAMSHIFT algorithm and compare the results with the MeanShift
3. Develop tracking demo using your tracker and OpenCV's counterpart and test on 5 datasets from http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html
4. Use MeanShift to cluster the data from here: https://cs.joensuu.fi/sipu/datasets/
   Use G2 sets and S sets. Compare the data with ground truth.

Create a command-line tool in Python that accepts the ROI of the target (the portion of the image that should be tracked in the video) and the path to the dataset and performs tracking of the target over the image sequence from the dataset. Visualize the current image and highlight the target's location by a rectangle. Find conditions where tracking is lost, explain why.

For clustering create Python notebook and embed the code with explanations into it

References:

[1] D. Comaniciu, P. Meer, "Mean shift: a robust approach toward feature space analysis," IEEE Trans. Pattern Analysis Machine Intell., Vol. 24, No. 5, 603-619, 2002, available at http://comaniciu.net/Papers/MsRobustApproach.pdf

[2] G. R. Bradski, "Computer video face tracking for use in a perceptual user interface," Intel Technology Journal, Q2, 705–740, 1998, available at http://www.dis.uniroma1.it/~nardi/Didattica/SAI/matdid/tracking/camshift.pdf

[3] http://www.cse.psu.edu/~rtc12/CSE598C/

[4] https://docs.opencv.org/3.4.1/db/df8/tutorial_py_meanshift.html