CS 231A Computer Vision, Spring 2017
Problem Set 4

Due Date: June 2th 2017 11:59 pm

1 Face Detection with HoG

In the previous problem set, we explored histogram of oriented gradients and how they could be represented a as a feature detector. Now, we will explore using it in a simple object detection pipeline, which involves support vector machines and sliding windows. You will be editing and running the file `detector.py`.

(a) **[20 points]** Given HoG features, we can train an SVM to recognize whatever we want. We have supplied the code to train an SVM on a large number of positive face examples of 36×36 px images and an even larger number of negative examples. Use this SVM to run a sliding window detector to find bounding boxes in `run_detector()`. Specfically, you will be implementing the sliding window in order to find bounding boxes around positive areas on the test images. Submit code and output detections.

(b) **[20 points]** Now that we have a large number of bounding boxes, we will notice that these boxes will cluster around the same area. Therefore, we need to implement nonmaximal suppression in `non_max_suppression()`.. After doing so, submit your code and the final results.

2 Image Segmentation

Image segmentation is the process of partitioning an image into meaningful structures. Two classical segmentation methods use k-means and meanshift [1]. In this problem, you will implement significant parts of both methods in `segmentation.py`.

K-means

(a) **[25 points]** Complete image segmentation using kmeans in `kmeans_segmentation()`. Submit your code and the resulting segmentation.

(b) **[5 points]** Try the different images provided (`plates.jpg` and `rocks.jpg`). Play around with the number of centers to get good results on these images and explain the effect of changing it. Also provide your segmentations on these images (try to get the best you can, but don't spend too much time).

---

[1] https://courses.csail.mit.edu/6.869/handouts/PAMIMeanshift.pdf

Meanshift

(a) [**25 points**] Complete image segmentation using meanshift in `meanshift_segmentation()`. Submit your code and the resulting segmentation.

(b) [**5 points**] Try the different images provided (`plates.jpg` and `rocks.jpg`). Play around with the bandwidth parameter to get good results on these images and explain the effect of changing it. Also provide your segmentations on these images (try to get the best you can, but don't spend too much time).