Homework #7
Due Mo. 4/17

You must turn in your code as well as output files. Please generate a report that contains the code and output in a single readable format.

Visit the book website to download companion software, including all the example problems.

Test Images
Download the sample images from the class website
http://www.ee.unlv.edu/~b1morris/ee482/docs/hw07

1. Viola and Jones
   In this problem, you will examine the Viola-Jones cascade object detector for faces. Use the code below to answer the following questions.

   (a) Compare the results on visionteam.jpg with the default MergeThreshold=4 and 0. Show the result side-by-side and comment on the parameter. Be sure to explain what the parameter does and how it affects the results.

   (b) Compare the results between the default model and the FrontalFaceLBP model (in order to get all detections use MergeThreshold=0). Show results side-by-side and comment on the performance comparison.

   (c) Repeat the part (b) comparison by testing the performance on the soccer team images. Give your processing time for each image (tic.m, toc.m).

```matlab
%% Viola Jones Face Detector
% read image
I = imread('visionteam.jpg');

MergeThreshold = 4;

% create Viola Jones detector
VJ = vision.CascadeObjectDetector('MergeThreshold', MergeThreshold);

detect faces
tic
bboxes = VJ(I);
% time detector
t_VJ = toc

% annotate detected faces
Ifaces = insertObjectAnnotation(I, 'rectangle', bboxes, 'Face');

figure;
imshow(Ifaces, []);
title('Viola Jones Face Detector');
```
23 % more parameters https://www.mathworks.com/help/vision/ref/vision.cascadeobjectdetector-system-object.html
24 %clean up
25 release(VJ);

2. Dalal and Triggs HOG Detector

In this problem, you will examine the HOG pedestrian detector. Use the code below to answer the following questions.

(a) Compare the results on visionteam.jpg and visionteam1.jpg using different parameter values. Show your “best” results side-by-side and give the parameters you used to generate each. Comment on where the HOG detector works well and where it fails.

(b) Compare the results between the default model and with different WindowStride values. Show results side-by-side and comment on the performance comparison. What are the advantages and disadvantage of different strides?

(c) Repeat the part (b) comparison by testing the performance on the soccer team images. Give your processing time for each image (tic.m, toc.m).

1 1
2 1 % HOG detector
3 2 %read image
4 3 I = imread('visionteam.jpg');
5 4 %I = imread('visionteam1.jpg');
6 5
7 6 %increase value when there are too many false positives
8 7 ClassificationThreshold = 0;
9 8
10 9 %adjust to group multiple responses
11 10 MergeDetections = false;
12 11
13 12 HOG = vision.PeopleDetector('ClassificationThreshold',
14 13 ClassificationThreshold, 'MergeDetections', MergeDetections);
15 14 tic
16 15 [bboxes,scores] = HOG(I);
17 16 t_HOG = toc
18 17
19 18 Iped = insertObjectAnnotation(I,'rectangle',bboxes,scores);
20 19
21 20 figure;
22 21 imshow(Iped,[]);
23 22 title('HOG Pedestrian Detector');
24 23 % notice detections contain area around the person (bounding boxes are not tight)
25 24 % more parameters https://www.mathworks.com/help/vision/ref/vision.peopledetector-system-object.html
3. Training a Detector

In this problem, you will train a cascade object detector on a custom dataset and compare the performance for different feature types.

(a) Read the Matlab [Get Started with Cascade Object Detector](#) page to better understand the training process and considerations. Be sure to read the help for `traincascadeobjectdetector` to understand the parameters.

(b) Implement the 5-stage stop sign detector using `stopSignsAndCars.mat`. Show your output using on `stopSignTest.jpg` using `MergeThreshold=1`.

(c) Increase the number of stages to 20. Compare the detectors using all three available `FeatureType`. Provide side-by-side comparison of test images for the different detectors (HOG, Haar, LBP). Give the final number of stages and training time for each. You may want to use the datastore input of `textttvision.CascadeObjectDetector` to process all four the sample images at once.

(d) Repeat part (c) the evaluation using the rear car labels. Comment on results by comparison stop signs.

(e) Try to improve the performance of the traffic sign detector using the [Tiny Lisa Traffic Sign Dataset](#). You do not need a Kaggle account since the zip is part of the homework files. You need to only use the stop sign images. Again show your results on the four sample images.