

Vehicle Detection Project

The goals / steps of this project are the following:

1. Perform a Histogram of Oriented Gradients (HOG), binned color and histograms of color feature extraction on a labeled training set of images and train a classifier Linear SVM classifier
2. Implement a sliding-window technique and use your trained classifier to search for vehicles in images.

Run the pipeline on a video stream (start with the test_video.mp4 and later implement on full project_video.mp4) and create a heat map of recurring detections frame by frame to reject outliers and follow detected vehicles.

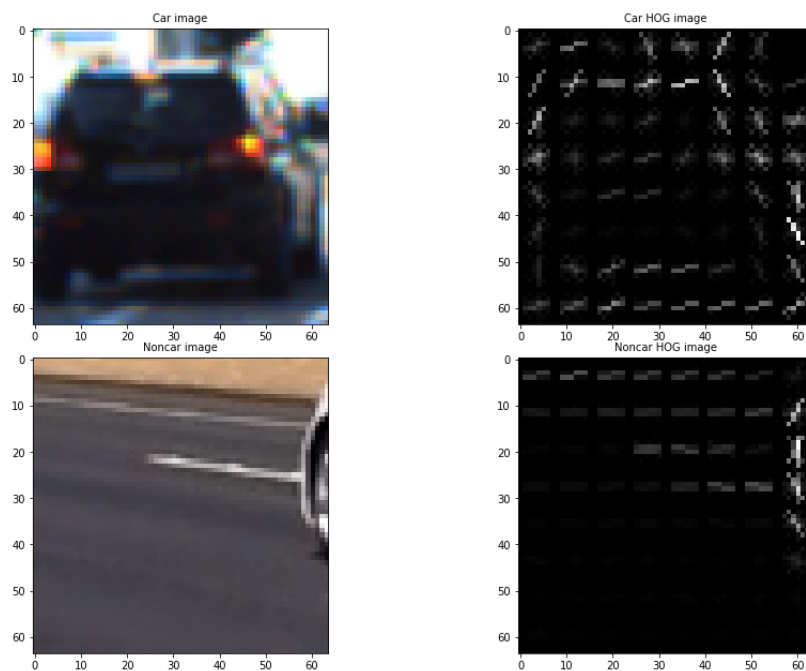
3. Estimate a bounding box for vehicles detected.

Example of data

Histogram of Oriented Gradients (HOG)

1. Explain how (and identify where in your code) you extracted HOG features from the training images.

I started by reading in all the 'vehicle' and 'non-vehicle' images. Here is an example of one of each of the 'vehicle' and 'non-vehicle' classes and the correspondent HOG images.



For the combination of color space features and HOG features extraction, I tried various combinations of parameters and after comparison, and the parameters I used in the feature extraction are:

colorspace = 'YCrCb'

orient = 9

pix_per_cell = 8 (for both x and y directions)

cell_per_block = 2 (for both x and y directions)

hog_channel = 'ALL' # Can be 0, 1, 2, or "ALL"

spatial_size=(32, 32),

histogram bin size=32

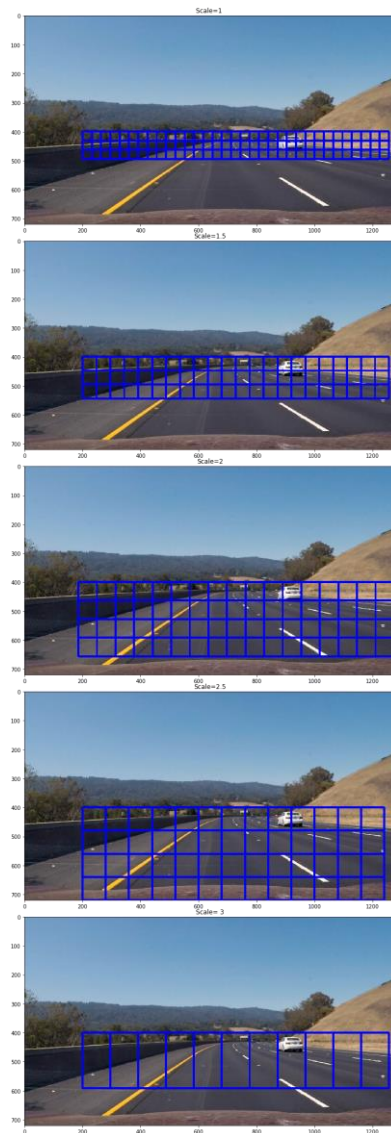
2. Describe how (and identify where in your code) you trained a classifier using your selected HOG features (and color features if you used them).

I trained a linear SVM using the extracted HOG features and color space features with car images and noncar images. The car features and noncar features are scaled to eliminate the influence of different feature value.

Sliding Window Search

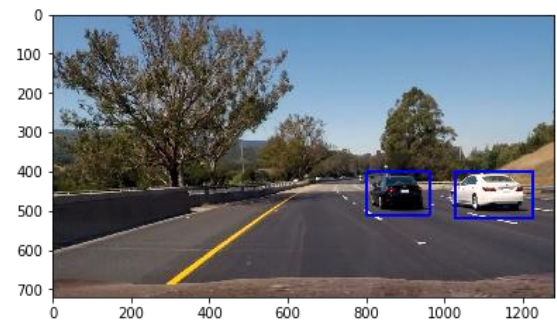
1. Describe how (and identify where in your code) you implemented a sliding window search. How did you decide what scales to search and how much to overlap windows?

I tried several scale values and search regions at different scales is shown in the following image:



2. Show some examples of test images to demonstrate how your pipeline is working. What did you do to optimize the performance of your classifier?

In order to optimize the performance of classifier, I added heatmap threshold to filter out the false positive and apply detection on certain area twice to prevent the undetection of the white car. The images below show that my pipeline implementation result on single images. The results are good but at the bottom left image, one of the rectangular is way too big.



Video Implementation

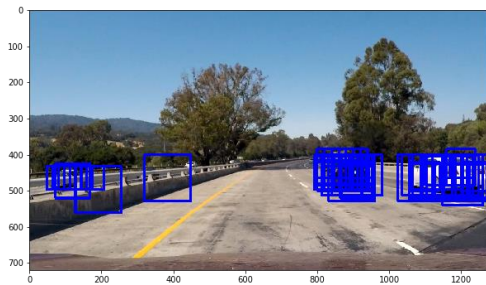
I applied the `process_image()` on each frame of the video, however, there are still several false positive on the output video. In order to improve the performance of the pipeline, I take the following actions:

1. Set the value on the far left to be 0 on the heatmap image to prevent the detection of car on the opposite direction;
2. Add the heatmap of 7 previous frames to the current heatmap and apply heat threshold on it; because the heatmap of false positive is not always at the same location within the 8 frames as the ordinary vehicle heatmap do. After comparison of the detection result, the threshold I use in the video pipeline for single image is 3 and 8 for final heatmap image.

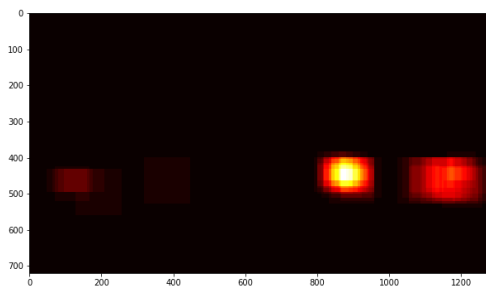
3. I also found that under certain parameters, there are several false positive detections with very small size, so I added another size filter to remove them;

Here is a example of the vehicle detection process on a single image:

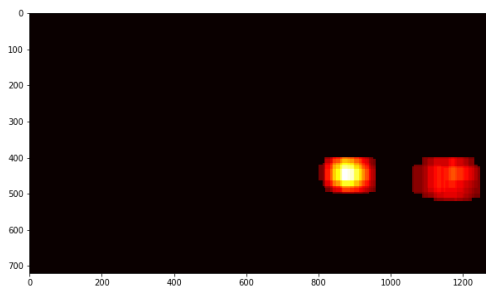
1. Scale the image and draw rectangles of different scales



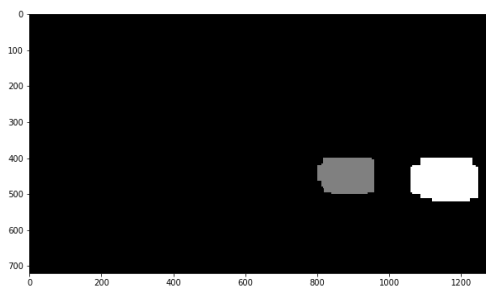
2. Draw the heatmap of the image based on the number of the rectangles



3. Apply threshold on the heatmap



4. Label the detected region



5. Draw a rectangle on the detected vehicle



Discussion

1. Briefly discuss any problems / issues you faced in your implementation of this project. Where will your pipeline likely fail? What could you do to make it more robust?

As is shown in the “project_video_output.mp4”, there is a frame with false positive at the shadow of trees and some frames with false negative on the white car detection. I will add more scans at scale 1.25 and/or 1.5 and also increase the threshold of either single image or the final heatmap image of several frames. In addition, I will apply restrictions on the window scan region in the x direction, which can not only increase the detection speed but also reduce the false positive.