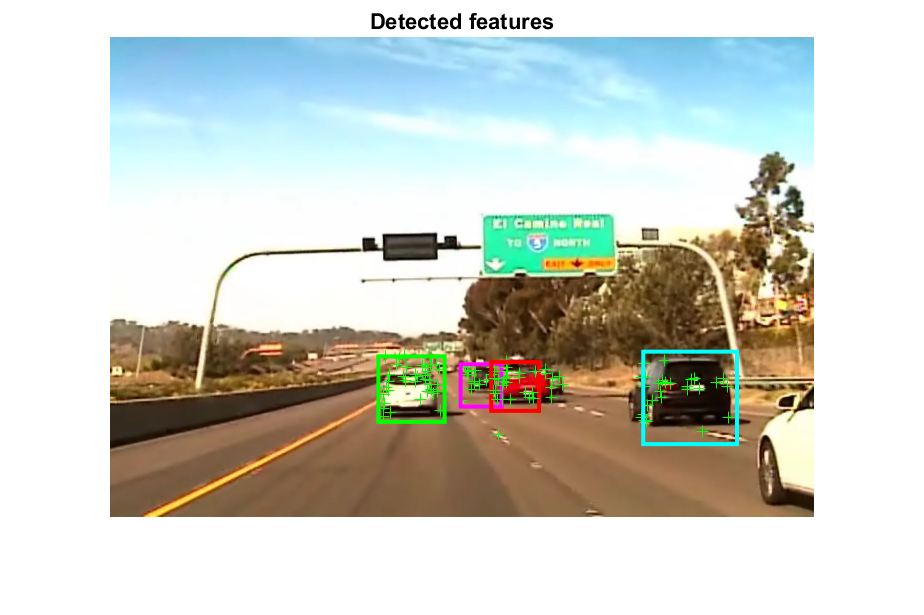
**ENPM 673 Project 3 Part 1**

The first step in tracking and detecting cars is to generate an .xml that will be used to train the HAAR cascade. For this assignment we were given a large set of sample images of cars and objects that are not cars that can be used to create .xmls. In addition to the data, there are various parameters that we could alter in the .xml generation such as the false alarm rate and number of cascades used. After some heuristic experimentation we were able to determine that 4 cascades and a false alarm rate of .02 create the most optimal set of training information.

Despite the good results from using all of the positive car data, we were still getting a decent amount of false detections. We then thought that we could have been over training the cascade. We got significantly better results when just using the GTI\_middleclose images. To further improve on our recognition we went through all of the car data samples and selected the best defined images in each set and created a diverse set of data that did not over train the classifier. This produced the best results and is what we used for the final video.

The image above shows the features recognized from the trained cascade. The green “+” show features that are recognized as features from the cascade.

Now that the program can properly identify the cars, the next step is to be able to track them. In order to do this, the program implements a combination of KLT tracking and closest bounding box tracking. We closely follow the code outline in the face-tracking matlab example (https://www.mathworks.com/help/vision/examples/face-detection-and-tracking-using-the-klt-algorithm.html) in order to implement KLT tracking. In order to get better results, we restrict the bounding box to be within a certain region that is defined by the most recent detection. This solves a problem we faced with false detections being tracked and incorrect bounding boxes being generated. This location update occurs once every 5 frames. In summary the code detects the cars every 5 frames and then uses the location of the detection to restrict the location of the KLT tracking.