Journal 9

Throughout these past two weeks, I have primarily been focused on strengthening my naive vehicle crash detection algorithm. Specifically, I was able to create a temporary solution to my problem of not being able to consistently detect the same vehicles throughout the video. I did this by reducing the threshold for the required confidence before my vehicle detection algorithm labels a particular region of the frame as a vehicle. In doing so, I increased the likelihood of detecting all of the vehicles in a given frame, however, this solution causes another issue to arise. Namely, by reducing the required confidence threshold, I make my algorithm more prone to incorrectly detecting vehicles in a given frame. I resolved this problem by noticing that most of these randomly detected regions are smaller in comparison to the actual detected vehicles in the frame. Consequently, I created a method that filters out the detected vehicles by using a size threshold, where the size for a given detected car is simply the length of the diagonal of its rectangular detection area. After a few trials, I found that using a filter threshold of 70 pixels eliminated a large majority of the arbitrary predictions. The two images below illustrate the performance of my filtering algorithm.

Pre-Filtration   Post-Filtration
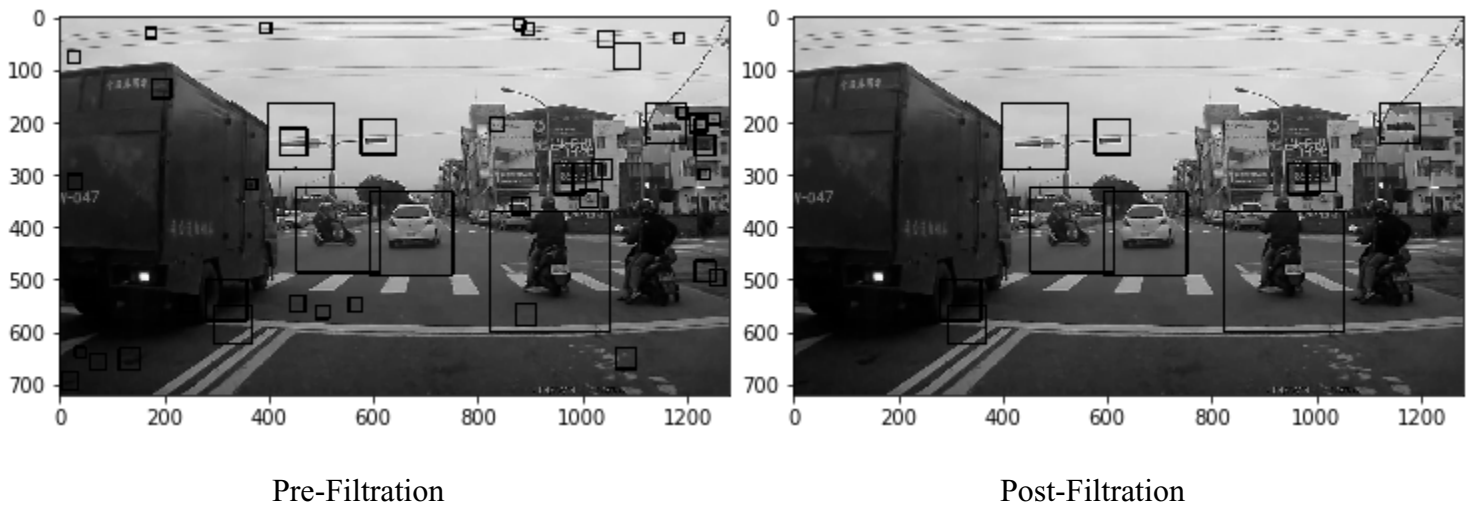
The image on the left shows an unfiltered image which explains why there are so many random vehicle identification boxes in the frame. The image on the right shows the frame post-filtration. Evidently, this frame has far fewer incorrect car identifications.

Currently, I am attempting to resolve the problem of determining which vehicle in a previous frame corresponds to one in the present frame. To address this, I have created a method that, when given the cartesian coordinates of a vehicle in a frame, determines which vehicle in the previous frame is geometrically the closest and then assumes that this vehicle is the same one. However, my solution doesn't address the case of when two *different* cars are in close proximity. In the coming weeks, I will incorporate the size of the cars and other external attributes of the image to clarify this distinction. Overall, in the last two weeks, I was able to make significant progress on my naive car crash detection algorithm which will allow for a smoother transition when I begin implementing Machine Learning.