Suchet Sapre
Period: 3
12/5/19

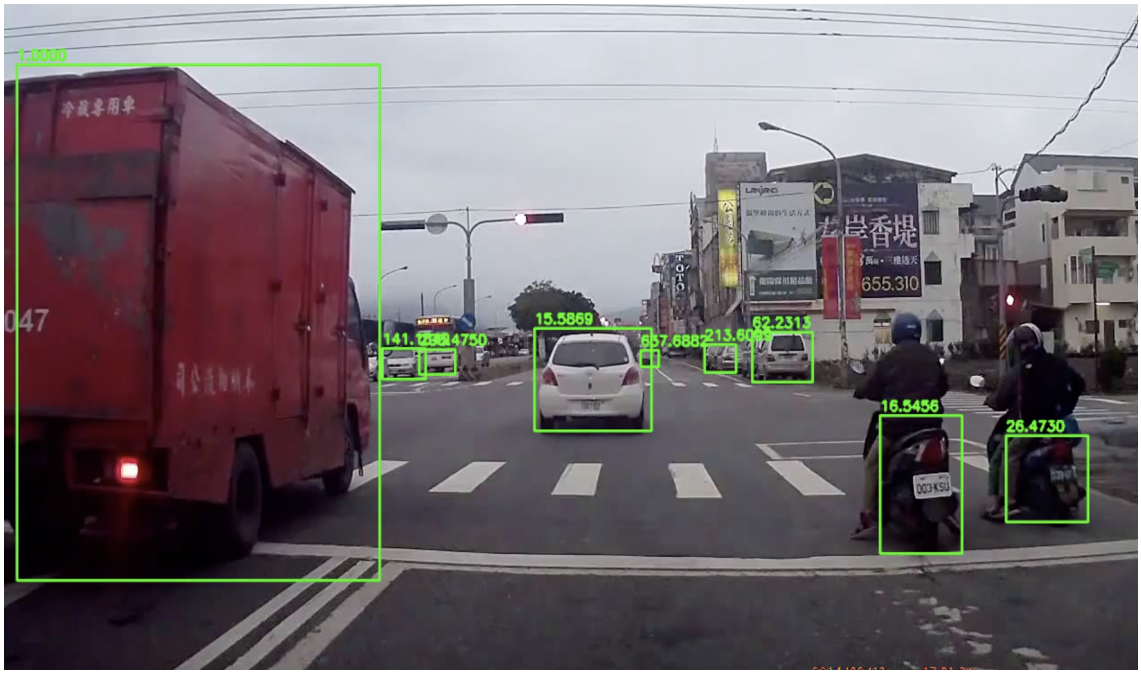
Journal 11

Throughout this past week, I focused on improving my vehicle crash prediction algorithm's depth "awareness." Specifically, the problem I'm trying to address is that when two vehicles pass by each horizontally, my crash prediction algorithm incorrectly predicts that these vehicles will crash with each other. This is because I'm not accounting for the relative depth of the vehicles when determining whether their theoretical trajectories intersect. To resolve this, I decided to use the area of a vehicle's bounding box to calculate its relative depth in the frame.

My relative depth calculation algorithm works as follows,

1. Choose a random vehicle in the initial frame of the dashcam video, V_{ref}, and calculate the area of its detected bounding box, A_{ref}. This vehicle will serve as the reference vehicle and has a relative depth of 1.
2. For each of the other vehicles in the initial frame, V_i
    a. Calculate V_i's detected bounding box area, A_i.
    b. The relative depth of this vehicle is then A_{ref} / A_i.

This algorithm has fairly high accuracy because vehicles that are farther from the camera (deeper in the frame) will have a smaller bounding box area. In the long-term, I will use this algorithm by checking whether vehicles whose trajectories intersect have close enough depths before I predict them as vehicles that will crash. Figure 1 shows the results of my algorithm. The numbers above the bounding boxes of the vehicles represent their calculated relative depths.



One of the main problems with my algorithm is that it doesn't consider the fact that the relative depths of vehicles can change between frames. I disregarded this for now because my object tracking algorithm doesn't change the size of the bounding boxes between frames. However, in the future, I will find an object tracker that also tracks the size of the vehicle in the image so that I can account for this. In addition, another problem within my algorithm is that all types of vehicles—motorcycles, cars, trucks, etc.—are treated the same. For instance, trucks are characteristically larger than cars, so even if a car and a truck are at the same depth, my depth calculation algorithm will incorrectly determine that the car is at a greater depth than the truck. In the near future, I will resolve this problem by determining a conversion factor between the sizes of cars, motorcycles, and trucks (among others), so that I can convert between the depths of different vehicles.