

# Spacee Take-Home Interview Question

## Computer Vision Engineer Position

### Question 1)



Given a 30 second aerial traffic footage (see traffic.mp4), write a program that extracts the following information:

- Total number of cars in the video
- A congestion heatmap, which highlights the areas with most traffic
- Detection of cars that exceed a configurable speed limit

Preferred tech stack is OpenCV + Python. Feel free to use other libraries as you see fit. You can approach the problem in any way. Accuracy is important, but your submission will also be graded based on architecture, code quality, performance, presentation, and how well your algorithm will generalize. You should be spending about 3-4 hours on this task.

#### Hints:

If you cannot accurately count the number of cars, you can try to estimate it. You can assume that trucks, buses etc. are cars. However, try not to count motorcycles as cars.

The speed limit will be given in mph, but you can make assumptions about mile-to-pixel ratio. Detection is a loose term in this context and can just be a print statement whenever a car exceeds the limit. Alternatively, you could draw a marker or a shape around speeding cars. You can assume the video is in real time (is not sped up/slowed down).

You are free to implement the heatmap in any way you would like, but red colors overlaid on heavily congested areas, and green or blue on empty roads would be a good start.

**Video Link:** [Video](#)

**Alternative Link:** [https://www.youtube.com/watch?v=Yle2\\_RFccZY](https://www.youtube.com/watch?v=Yle2_RFccZY)

**Question 2)**

Write the code to calculate and plot the Precision-Recall curve for any existing Deep Learning model you like. Could be a classifier or detector. Use any ground truth dataset you like. For example, it could be of cats, dogs, birds, etc., or bicycles, cars, trucks, etc., or any objects like that. However, the dataset choice is yours. Turn in the code sample, GT dataset, and PR curve plot.

\*Bonus for adding the P-R results of any second (but different) classifier or detector to the same chart for comparison purposes.