# Project 2B

### The Basics:

* The assignment was done in python.
* The wrapper function is called group3\_wrapper.py.
* Opening ipython and typing %run group3\_wrapper.py will generate the result shown in the image cars-mosaic.png.

### A Few Comments:

* We found better results when suppressing non-maxima to not use the 0.9-as-large condition. Instead, we suppressed based on one-to-one comparison to four nearest neighbors. The parameter can be adjusted, however, by changing the kernels called left, right, up, and down to the desired values.
* We also found for the images that we used that max\_pts = 1000 provided enough points to for matching, where as 500 provided too few and 2000 severely slowed computation.
* Feature descriptors are 8x8 windows (samples happening every 5 pixels) that’s rotated by the orientation of the interest points and sampled from the image after smoothing with a 5x5 Gaussian.
* Feature matching employed the annoy module and performed a forward and backward ratio test of the nearest neighbor and second nearest neighbor to a given interest point. The ratio we used to determine a match was 0.9 as opposed to the recommended 0.6 or 0.7, which proved to be too restrictive in our case
* For RANSAC, we kept the 0.5 pixel threshold and bumped up the iterations to 2000 since the computation time cost was little compared to the overall runtime of the assignment.
* We did not employ any blending techniques that Chenhao confirmed were for extra credit during image stitching