Skyline Classifier - Part 3 Report
Benedict Becker

Dataset

For my test dataset, I assembled a new set of 175 images of the same four skylines disjoint from the training set. To make these images different than the training images, I only picked pictures that were taken at night. While some images in the training dataset were taken at night, the majority weren't and I thought this would be a good test of my model.

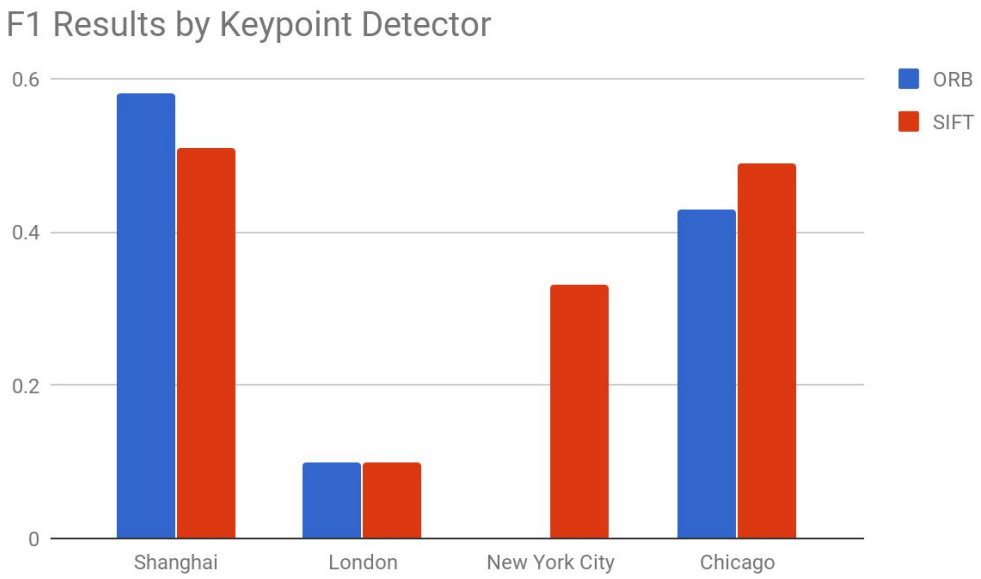
Testing Methods

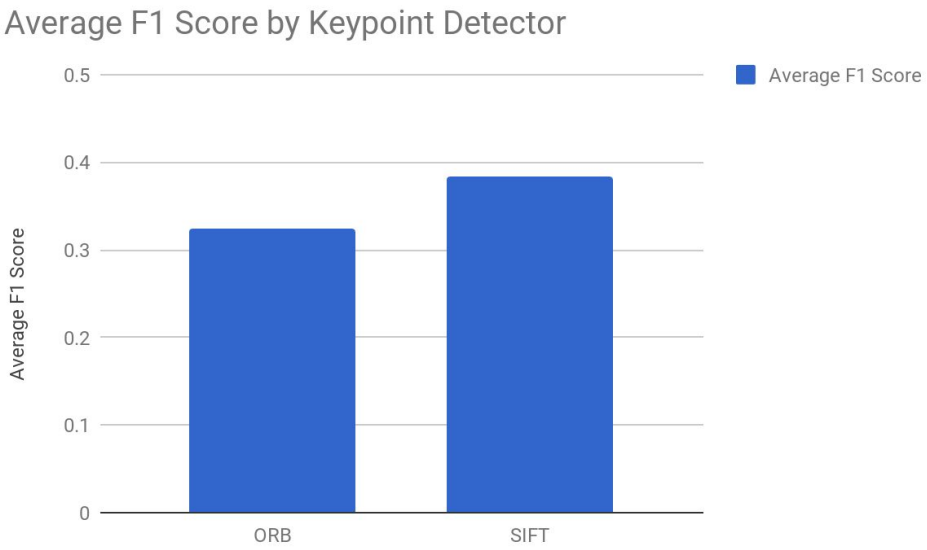
As I referenced in Part II of the project, I still had some other methods to try, namely SURF, SIFT and an ensemble of the three (with ORB). I built the contrib portion of opencv, and was able to use both with success, getting better validation results than with ORB detected keypoints. Unfortunately, when I started to perform my final tests, the underlying C code for a component of the SURF pipeline kept raising segmentation faults, so I had to carry on without this method. This also renders the ensemble method unusable because I only have two keypoint detectors. Therefore, I used SIFT and ORB to perform my final tests.

In addition, I tried chopping off different amounts of layers of the VGG network at another attempt at transfer learning (as you suggested), but this was unsuccessful on all fronts. Maybe deep learning isn't the key to everything!

Results

	ORB	**SIFT**
Weighted Accuracy	0.434285714286	0.4





Discussion

As expected, the model performed much worse on the test set. However, compared to the validation performance, the model performed better than I expected given the different nature of the test set. As I hypothesized in Part II, the SIFT detector did a better job of detecting high quality keypoints, which made for better performance on the whole. While ORB did have a test performance above the baseline, I think that it performed worse than its F1 score indicates.

ORB fails to detect New York City even once, and failing to accurately classify an entire category is abysmal on such a small number of classes. SIFT, while still scoring low in classifying London, had more consistent class-to-class results. ORB did have a significantly higher weighted accuracy however, so the different models performed better according to different metrics (another reason why an ensemble might have been a good idea).

My model performed worse than the validation because in validation, there were a lot of day shots and a few night shots, so if the model was good at day shots but bad at night shots, it wouldn't affect the accuracy too much. However, when you make a test set of *all* night shots, then suddenly this weakness in the model gets exposed and it does a lot worse. This shows why it's important to train your model on lots of different types of data, even if it would be a less common occurance.

Improvements

I think my model would improve a lot with scale. My dataset didn't have a lot of data, and clearly it wasn't varied enough to perform well on the night dataset. Additionally, I would've tried larger parameters (more features, etc.) but I kept running out of memory so I wasn't able to try everything I wanted.

SURF working might have helped as well, because three different keypoint detectors working together might perform better than just one, as one detector might detect things that the others don't. Since my metrics are distance-based, a close match with a few keypoints would result in a confident classification and overrule two weak dissenting opinions (in an ideal world).

Appx A

Illustration of some problems

Part of why London was so difficult was that the “skyline” was actually many different places. So without that much data, it was difficult for the model to classify it. These two pictures show how different it could be. (The detector was good at avoiding watermarks)