Homework 2 Writeup

Instructions

- This write-up is intended to be 'light'; its function is to help us grade your work.
- Please describe any interesting or non-standard decisions you made in writing your algorithm.
- Show your results and discuss any interesting findings.
- List any extra credit implementation and its results.
- Feel free to include code snippets, images, and equations.
- Use as many pages as you need, but err on the short side.
- **Please make this document anonymous.**

Implementation Detail

In my implementation of RANSAC, I used the distance of $x^T F x'$ from 0 as the distance metric and a threshold of 0.025 (found empirically). I used least squares in matches_to_3d to estimate the 3D points in the scene and SVD in estimate_fundamental_matrix to estimate F using 2D image point correspondences. In both bases, I expressed the systems of equations in the form $Ax = b$ and solved for the unknown parameters in x.

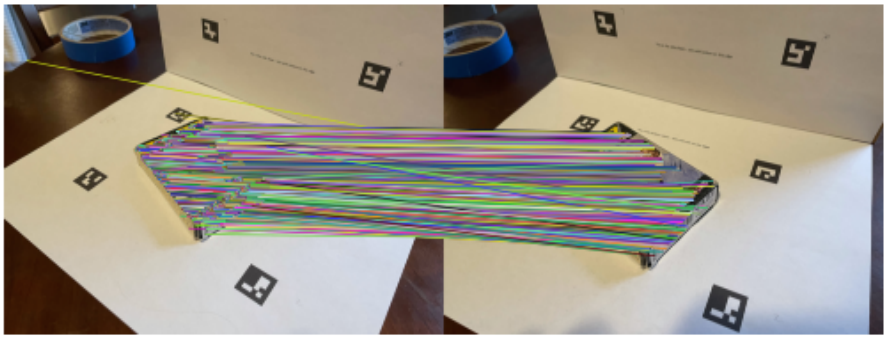
Result

When tuning the hyperparameters for RANSAC, I noticed that the percentage of inliers over the entire point set was highly sensitive to the threshold. When I started with a high threshold (0.3), almost all point correspondences were retained as inliers after running RANSAC, meaning many of the spurious matches were passed into the F estimator. With a very low threshold of 0.001, however, only a small handful of the strongest correspondences remained inliers. I also noticed that most of these strong matches were along the lower half of the book. The sweet spot I found was at around 0.025, where there was still some noise but almost all important matches were kept as inliers.

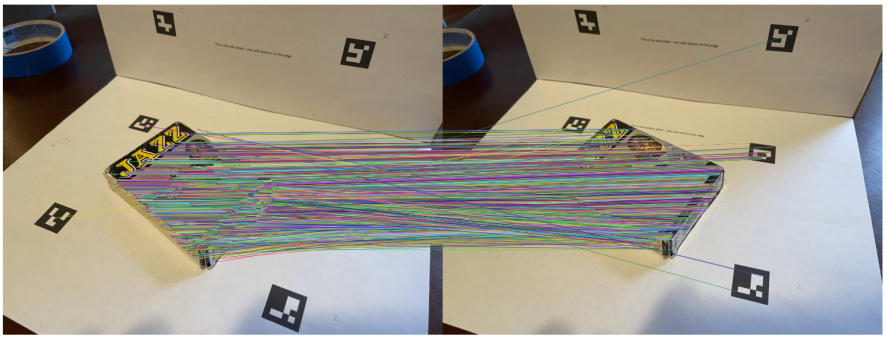
I also noticed that increasing the number of iterations in RANSAC increased the inlier percentage. However, for almost all image pairs, increasing the number of iterations past 500 resulted in few increases to the best inlier percentage, as the optimal ones are found within the first few hundred iterations.

Before RANSAC:

50 iterations at 0.025 threshold:



500 iterations:



Point cloud reconstruction