Project 3: Buildings built in minutes - An SfM Approach

Kartik Madhira
Masters of Engineering in Robotics
University of Maryland, College Park
Email: kmadhira@terpmail.umd.edu

Prateek Arora
Masters of Engineering in Robotics
University of Maryland, College Park
Email: pratique@terpmail.umd.edu

Abhinav Modi
Masters of Engineering in Robotics
University of Maryland, College Park
Email: abhi1625@umd.edu

Using one late day

I. Introduction

Structure from motion (SfM) is a technique for estimating three-dimensional structures from two-dimensional image sequences that may be coupled with local motion signals. In this project we are required to reconstruct a 3D scene and simultaneously obtain the camera poses of a monocular camera w.r.t. the given scene using just pictorial information.

II. Data

A set of 6 images of building in-front of Levine Hall at UPenn, using a GoPro Hero 3 with fisheye lens distortion corrected. Keypoints matching (SIFT keypoints and descriptors used) data has also provided in the same folder for pairs of images. The data folder contains 5 matching files named matching*.txt where * refers to numbers from 1 to 5. For eg., matching3.txt contains the matching between the third image and the fourth, fifth and sixth images, i.e., $\mathcal{I}_3 \leftrightarrow I_4$, $\mathcal{I}_3 \leftrightarrow I_5$ and $\mathcal{I}_3 \leftrightarrow I_6$.
Each matching file is formatted as follows for the i^{th} matching file:

nFeatures: (the number of feature points of the ith image each following row specifies matches across images given a feature location in the ith image.)

Each Row: (the number of matches for the jth feature) (Red Value) (Green Value) (Blue Value) ($u_{\text{current\_image}}$) ($v_{\text{current\_image}}$) (image id) ($u_{\text{imageid\_image}}$) ($v_{\text{imageid\_image}}$) (image id) ($u_{\text{imageid\_image}}$) ($v_{\text{imageid\_image}}$)
An example of matching1.txt is given below:

nFeatures: 2002 3 137 128 105 454.740000 392.370000 2 308.570000 500.320000 4 447.580000 479.360000
2 137 128 105 454.740000 392.370000 4 447.580000 479.360000
The images are taken at 1280 960 resolution and the camera intrinsic parameters K are given in calibration.txt file.

III. SFM approach

The first step is to find the fundamental between the first two cameras(or images). Once we have the **fundamental matrix**, we extract **essential matrix** using the intrinsic camera parameter K. Now that we have essential matrix, we find Camera poses. Here, we get 4 set of poses i.e. $\{C, R\}$,$\{-C, R\}$,$\{C, -R\}$,$\{-C, -R\}$. In order to get the correct rotation and translation we perform cheirality check. After obtaining the correct camera pose we perform non-linear triangulation to optimize the result. Now we have to repeat the process for all the images. In order to do so we need pose of each camera view, which we find using perspective-n-points. Further pipeline is performed for image 3 to last image:

1) Register the ith image using PnP, where is the image number ranging from 3 to last image. Perform PnPransac to get pose of the new camera view(or image) to get C_{new} and R_{new}
2) Using the C_{new} and R_{new} as initial guess perform non-linear optimization to get good estimate of C_{new} and R_{new}.
3) Store the poses obtained
4) Next we need to add new world point to the set of world points that are already existing. To do so, we perform linear triangulation and after that non-linear triangulation to obtain X_{new} i.e new world points.
5) Build visibility matrix and perform bundle adjustment

IV. Linear triangulation

Reprojection error for Linear triangulation between image 1 and 2 using pose obtained from decomposition of essential matrix is **9.97**. Mean reprojection error for linear triangultaion of all views after Linear and non-linear Pnp is *630.45*.

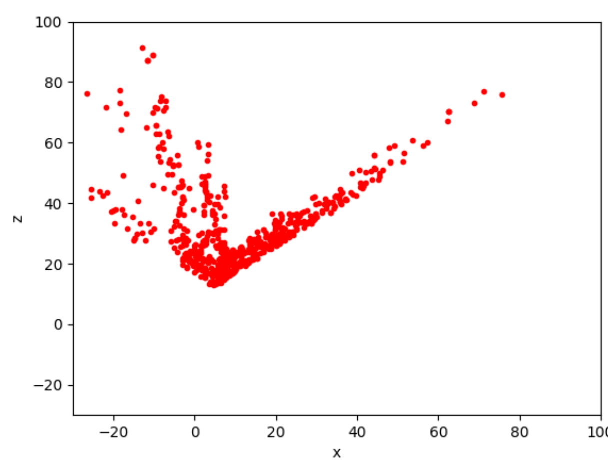

Figure 1: Linear triangulation between image 1 and image 2

V. Non-linear triangulation

Reprojection error for non-linear triangulation between image 1 and 2 after refining the initial guess from linear triangulation is **8.25**. Mean reprojection error for non-linear triangulation of all views after after Linear and non-linear Pnp is **406.32**

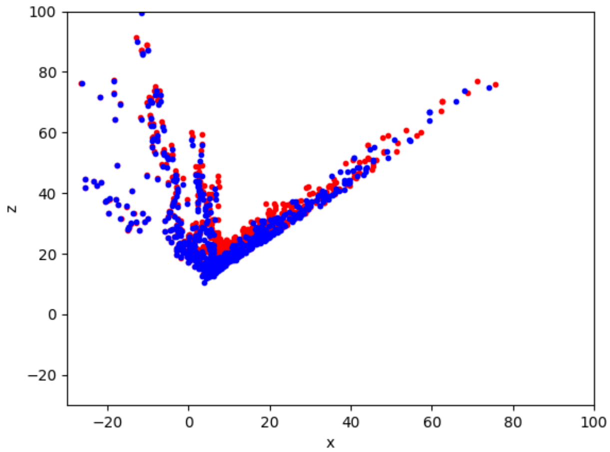

Figure 2: Non-linear triangulation between image 1 and image 2; Blue points are points after non-linear triangulation

VI. Results

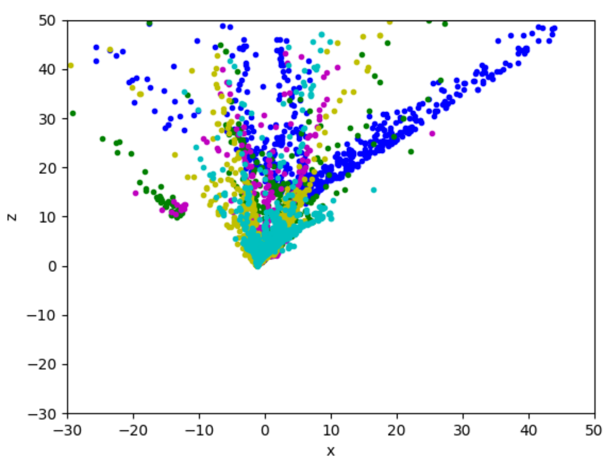

Figure 3: Top view of final output for all poses

VII. Discussion and Conclusion

Reprojection error have been presented above for Linear triangulation and Non-linear triangulation. In these step we notice that linear triangulation uses SVD to compute 3D projections. This estimate is not very good because it is obtained using linear least squares method. Once nonlinear optimization is performed by minimizing reprojection error,

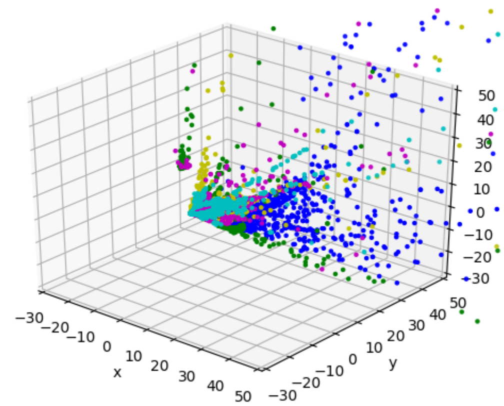

Figure 4: Scatter plot of final output for all poses

we observe that the projections obtained are far better than the previous estimates. The error exponentially accumulates after each step. Thus, we need bundle adjustment to reduce the reprojection error in all the images.