Scalable Structure from Motion for Street View

Yujia Huang (yujiah1), Yilin Yang (yiliny1), Group No. 60

Introduction

Motivation

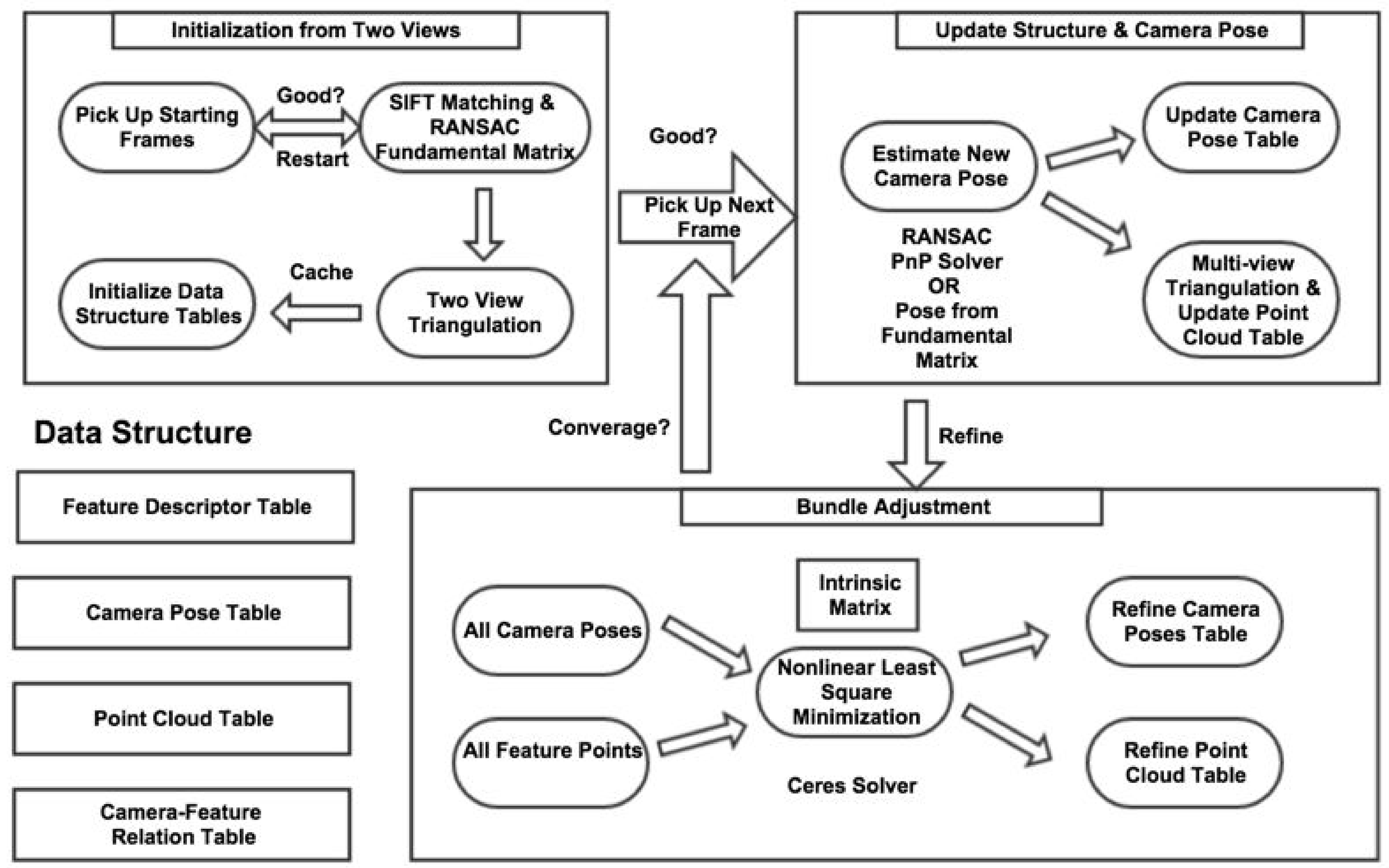
Structure from Motion (SfM) is a mature algorithm in computer vision. Traditionally, it is used to reconstruct the 3D point cloud given several images of an object. However, the camera poses are estimated at the same time, which provides further information and has many potential applications. For instance, we can track the trajectory of a vehicle with a camera. Furthermore, given Google Street View database, we can localize the trajectory more precisely on Google Maps than GPS. This information can assist autonomous driving system.

What we have done

In this project, we completed a SfM system specifically on street view.

- First version: MATLAB prototype
- Second version: OpenCV C++ without bundle adjustment, but real-time
- Final version: OpenCV C++ with Ceres bundle adjustment solver
- Dataset: Our own street view data collected by driving in a vehicle around Pittsburgh with a 35mm-fixed-focus lens
- Other work: Verified our implementation on our own this dataset as well as online dataset; Compared some successful results with a unsatisfying one and analysed when the system may fail

Method



Estimate Camera Pose

- PnP Solver

Given n 3D reference points, in the object reference framework, and their corresponding projections, PnP Solver retrieves the rotation matrix R and the translation vector t, accounting for camera orientation and position, respectively.

Triangulation

- Multi-view Triangulation

Solve 3D points (X, Y, Z, W) based on all observations (x, y) and projection matrix P

$$x_j = \frac{p_{00}^{(j)}X + p_{01}^{(j)}Y + p_{02}^{(j)}Z + p_{03}^{(j)}W}{p_{20}^{(j)}X + p_{21}^{(j)}Y + p_{22}^{(j)}Z + p_{23}^{(j)}W} \qquad y_j = \frac{p_{10}^{(j)}X + p_{11}^{(j)}Y + p_{12}^{(j)}Z + p_{13}^{(j)}W}{p_{20}^{(j)}X + p_{21}^{(j)}Y + p_{22}^{(j)}Z + p_{23}^{(j)}W}$$

Bundle Adjustment

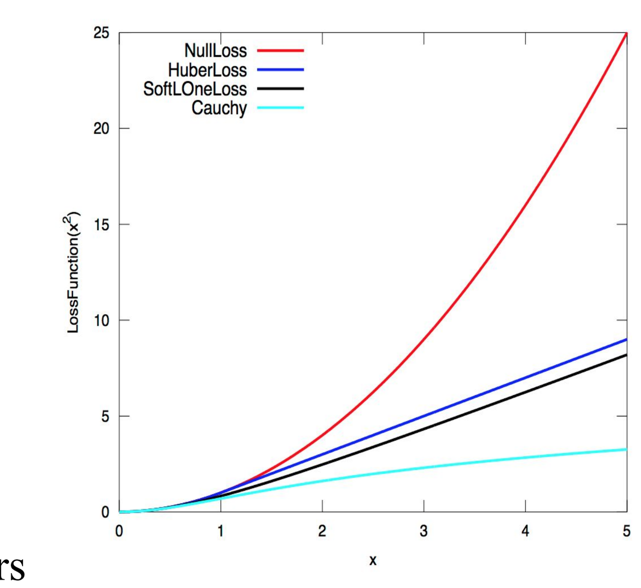
- Objective Function

Minimize camera poses C and 3D points $\boldsymbol{X}$ at same time

$$\min_{C,X} \sum_{i=1}^{n} \sum_{i=1}^{m} w_{ij} \left\| \boldsymbol{p}_{ij} - P\left(C_i, \boldsymbol{X}_j\right) \right\|^2$$

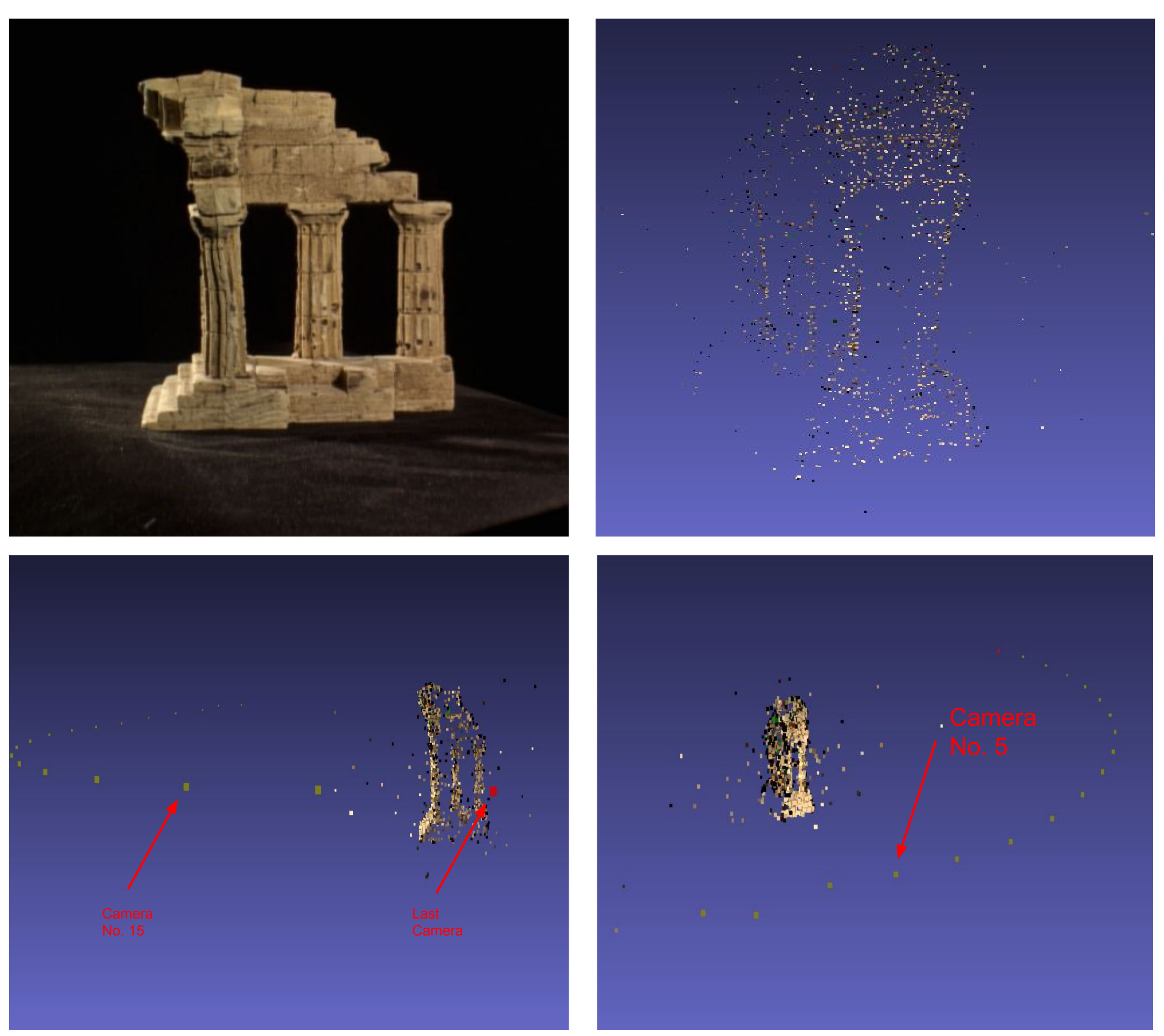
- Loss function

Cauchy function: $\rho(s) = \log(1+s)$, which depress outlier errors

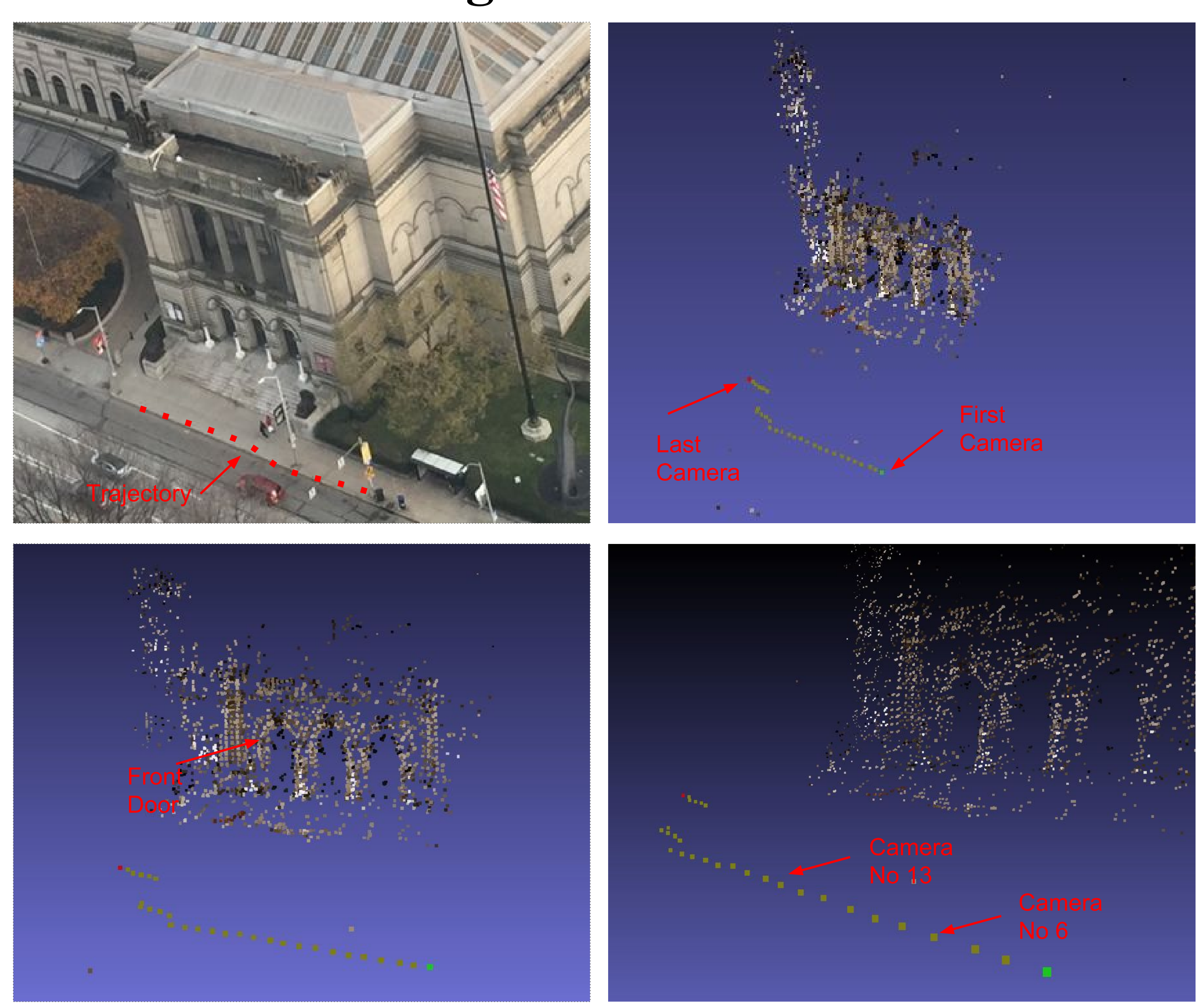


Results

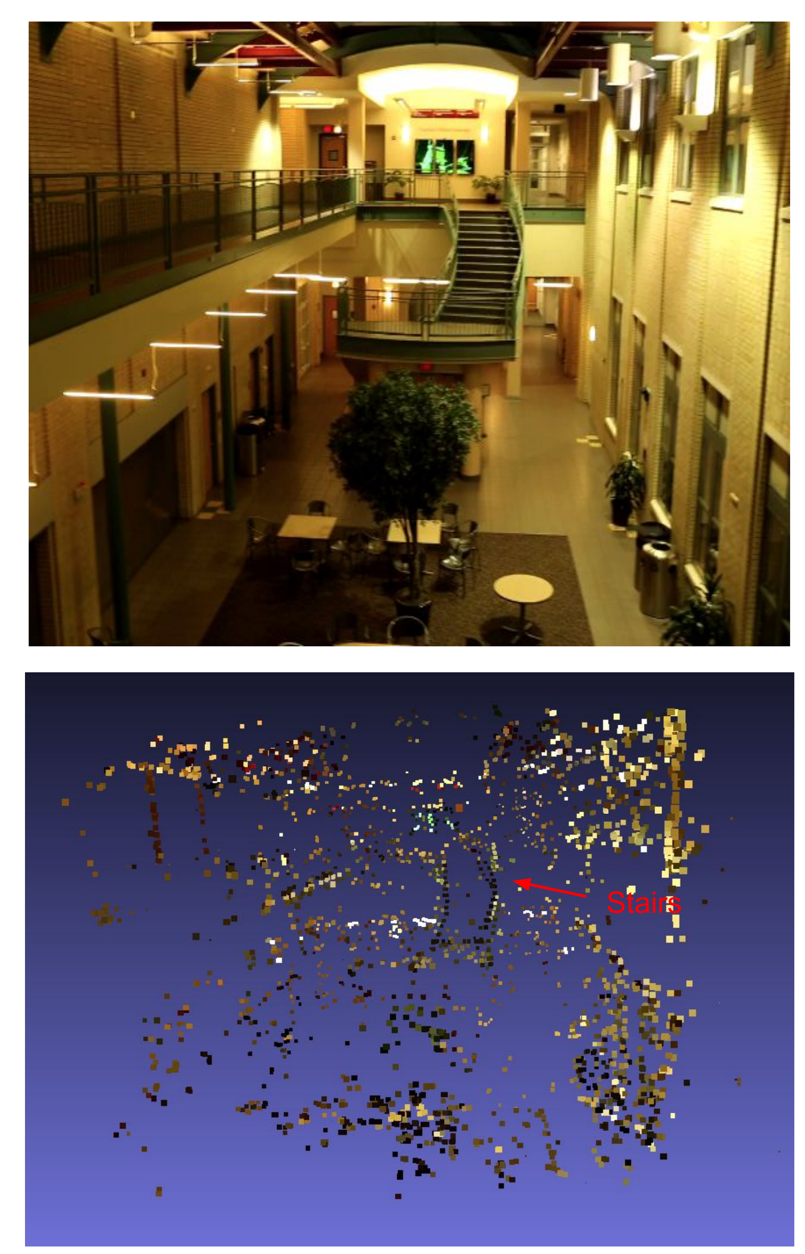
Middlebury Data: Temple



Real Data: Carnegie Museum of Art



NSH Indoor:



Future Work

We have completed SfM algorithm that works well on street view data, but there are many improvements available in the future:

- Robust feature matching
  Our system may break down if the new frame have insufficient features matched to the features in the table. This often happens when the speed is fast and the images are blurred. One proposal is to reduce the exposure time of the camera. Another proposal is to pre-process the frame and eliminate the motion blur.
- Handle outlier
  Bundle adjustment may not converge because of these outliers. One solution is to remove the point if it is not in front of all the cameras when triangulation is finished.
- Real time
  Bundle adjustment takes several seconds to converge, which makes the system not real-time. In future work, we can make Bundle adjustment converging faster utilizing the linear solver in Ceres.
- Loop Closure
  The error accumulates after each frame, even with bundle adjustment. However, if a trajectory has a loop, we can adjust the trajectory such that the camera positions where we see the same scene coincide with each other. This method is called loop closure.

Availability

Github: https://github.com/erichuang0771/OpenSLAM.git