BUilding in a day: Comparing Structure from Motion across scenes

Noah Markowitz, Kanghyun Lee

23.04.2024
CS 585

INTRODUCTION

Structure from Motion (SfM) is a photogrammetric technique for estimating three-dimensional structures from two-dimensional image sequences that may be coupled with local motion signals. First proposed in 1979[1] with a working algorithm presented in 1992 as the Tomasi-Kanade algorithm[2] ,it is a powerful tool in the field of computer vision and is used to generate 3D models of objects, scenes, and architecture. It has made headlines in media with its capability to create reconstructions of large structures such as the Roman Colosseum[3,4]. Part of the power behind this technique is the ability to reconstruct using sparse data making it an excellent method for analyzing large structures and landmasses[5,7].

In this project we collect data by taking images from around Boston University and apply SfM to the images in order to create reconstructions of some of the buildings and sculptures seen on the campus. We demonstrate how sparse data collection is capable of creating reconstructions of large structures.

Methods

Photos were taken of various scenes and buildings around the BU campus to attempt to reconstruct. These include the Integrated Life Science and Engineering building (ILSE), Marsh Chapel, and Labyrinth of Datalist sculpture. In addition photos were taken of one of the authors as well as a scene from the Reservoir closer to the Boston College campus. These types of images were chosen due to their difference in geometry, scale and elements in the scenes.

Reconstruction was done using the published SfM toolbox OpenSfM[6]. It was determined that, in order to perform enough reconstructions, a previously built toolbox would be of great value as it has pre chosen parameters for the various steps involved in SfM and therefore more time could be dedicated to reconstructing multiples scenes as opposed to enabling a single object to be reconstructed. The next section describes the workflow implemented in OpenSfM.

Workflow

CAMERA CALIBRATION

Before any processing can be done, the camera used to take images is first calibrated using a checkerboard pattern. This gives parameters to optimize reconstruction such as the camera's focal length, optical center and distortion coefficient. For comparison the ILSE building is reconstructed without camera calibration computed to gauge how SfM performs when camera intrinsics supplied by the image files are used.

Metadata

First step in processing is extracting the metadata. In this step, the EXIF (Exchangeable Image File Format) data will be extracted from the images. This data contains metadata such as the focal length, sensor size, and GPS coordinates (if available) from the images. This is useful if camera calibration has not been done and if precise position over a larger area is desired.

Feature Detection

Next step is to detect features, such as edges, corners, or distinct blobs. This can be done by using feature detectors, such as SIFT or HAHOG (Hessian Affine detector and Histogram of Oriented Gradients descriptor). The typical principle behind detectors is to calculate the intensity of different image gradients or intensity gradients across the

scene. For example, HAHOG will find points where the change in image intensity is locally maximal, and these points tend to be corners or blobs. A SIFT detector captures key points in the image at various scales and computes an orientation based on local image gradients.

Once those feature points are extracted an observation matrix W is created, which is a multiplication of M (camera motion matrix) and S (scene structure matrix). The goal is to solve for M and S, but it is not efficient to directly solve for M * S as there are numerous 0 rows and columns. Therefore, factorization, which will insert a bridge matrix to separately compute for M and S, can be introduced and once the equation is solved for M and S, this will be enough information for SfM to place those feature points in the 3D reconstruction plane.

Feature Matching

The next step is matching features, which will match those feature points that were detected through one of the above mentioned feature detectors. OpenSfM uses a matching algorithm that compares those feature points from different images and finds matches based on descriptor similarity. For computational efficiency images with no overlapping features (such as two images taken from opposite sides of a building) are split into subgroups so only relevant images are compared against all other relevant images.

Create Tracks

All images are compared against each other to find a set of feature points that are all of the same physical point in the scene to form tracks. A track is a set of feature points across images that are all of the same physical point in the scene. This step involves linking matched features that are believed to be of the same point into tracks.

Reconstruction

After creating tracks, a sparse 3D reconstruction is created. This process can be summarized by matching feature points across different images and creating initial Reconstruction from initial pairing, and adding images on top of the initial reconstruction to create a full 3D reconstruction. Once initial pairing is done, relative camera angle and pose by triangulation is computed to give hints as how to place each feature point in a 3D reconstructing plane. New images will be added to this initial reconstruction one by one while new relative camera angle and pose can be computed

Bundle Adjustment

Bundle adjustment is a process to minimize the difference between observed 3D points and their estimated 2D projection onto the camera's image plane. The mathematical equation of Bundle Adjustment will be the minimization of the summation of each feature point's squared value of difference between observed 3D point and their estimated 2D projection. From this optimization, accuracy of this 3D reconstruction can be significantly improved.

Undistorting Images

Using the previously done camera calibration, pixels are undistorted. Each pixel can then be mapped from distorted images to corrected and undistorted images.

Compute Depthmaps

Depthmaps provide per-pixel depth information, which can be used to create a denser point cloud or a textured 3D model. These are computed for each image based on the matches and are merged to form a more dense pointcloud.

Reconstruction with NeRFs

Recently Neural Radiance Fields (NeRFs) are being used for reconstruction using more dense data. It has recently become available to the public in the form of a smart-phone application called Luma.ai[8]. Using Luma.ai three reconstructions were done on the same objects as those reconstructed using SfM to compare the two methods by taking short videos of the same objects.

RESULTS

After processing we find that SfM is able to create reconstructions from small amounts of information that provide detail such as depth. Figure 1 shows Marsh Chapel reconstructed with differing amounts of photos. Seven photos are enough for reconstructing part of the building. Using 27 images the trees become more clear. Using over a hundred images more of the walls of Marsh Chapel were attempted to be reconstructed. The back wall of the building is partially reconstructed but not the side walls. Figure 2 shows ILSE building and the Labyrinth Datalist sculpture. Both provide excellent detail for the scale and geometry that each displays.

SfM reconstruction of the Chestnut Hill Reservoir (figure 3) worked for rigid

objects such as the trees and ground. However, the water, a dominant feature of the images taken, is not represented in the reconstruction due to not enough matching features being found.

NeRF reconstruction, shown in figure 4, shows finer detail for two of the three comparisons including Marsh Chapel from a single perspective and a human in a parking lot. However, the sculpture image shows lots of noise in the NeRF reconstruction. NeRF appears to have finer detail when dense data is present but is susceptible to noise when not enough overlapping features are detected.

CONCLUSION

Here we've demonstrated the capability of SfM to reconstruct large structures in a resource-efficient manner while requiring less data than other available techniques. The limited resolution, though not ideal for structures that are within the range of human size, is ideal for buildings and landscapes that have larger surface areas and are less practical for methods such as NeRFs and LIDAR.

Extending this work would involve the use of multi-perspective aerial photographs such as those taken from a drone. Though BU has tall buildings and windows to take pictures from, they don't provide easy multi-perspective photographs. Though limited in spatial resolution, SfM is ideal for reconstruction when data is sparse (either in terms of number of images or data per area) and structures of interest are large such as landscapes and cityscapes.

REFERENCES

1. Ullman, S. (1979). The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, *203*(1153), 405-426.
2. Tomasi, C., & Kanade, T. (1992). *Shape and motion from image streams: a factorization method: full report on the orthographic case*. Cornell University.
3. Snavely, N. (2011). Scene reconstruction and visualization from internet photo collections: A survey. *IPSJ Transactions on Computer Vision and Applications*, *3*, 44-66.
4. Schonberger, J. L., & Frahm, J. M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4104-4113).
5. Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J., & Rosette, J. (2019). Structure from motion photogrammetry in forestry: A review. *Current Forestry Reports*, *5*, 155-168.
6. Adorjan, M. (2016). *Opensfm: A collaborative structure-from-motion system* (Doctoral dissertation, Wien).
7. Carrivick, J. L., Smith, M. W., & Quincey, D. J. (2016). *Structure from Motion in the Geosciences*. John Wiley & Sons.
8. https://lumalabs.ai/

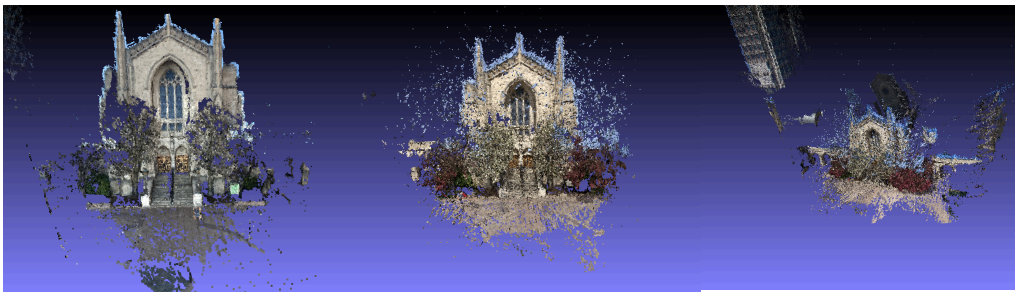

Figure 1: Reconstruction of Marsh Chapel using various quantities of images. Left) Seven images used. Middle) 27 images used. Trees become more visible. Right) Over a hundred images used. Neighboring building is visible and back wall of Marsh Chapel is visible

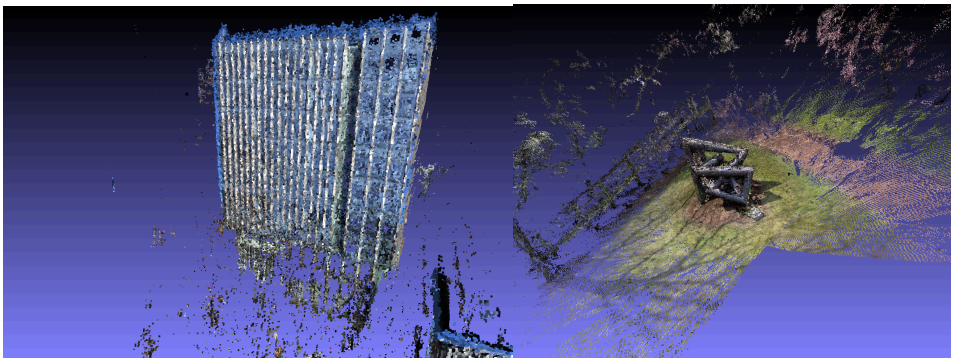

Figure 2: Images of LISE building and Labyrinth Datalist sculpture.

Figure 3: Left) Reconstructed pointcloud of Chestnut Hill Reservoir. Right) One of the images used in reconstruction. This highlights the trouble SfM faces with water. Possibly due to difference in optical flow and reflectance across frames

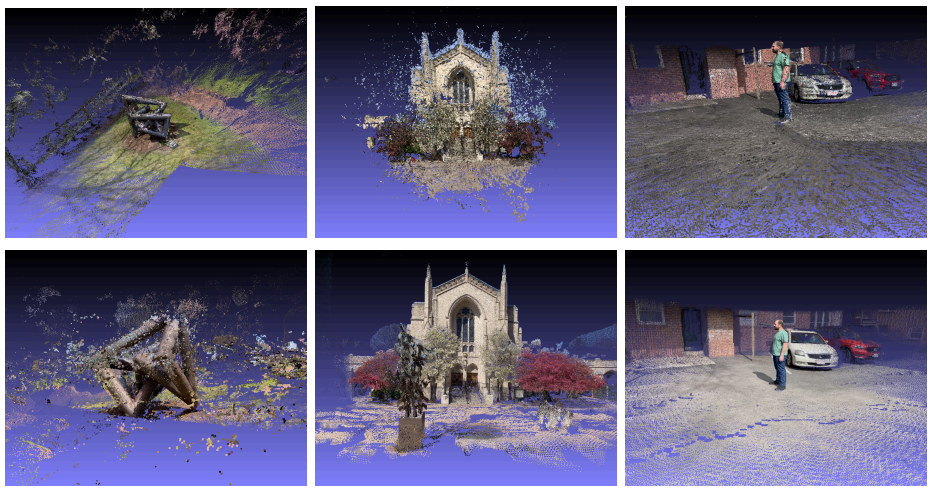

Figure 4: Top images are pointcloud reconstructions of areas using SfM while bottom images are NeRF reconstructions of the same scenes. NeRF provides finer resolution compared to SfM for close and limited perspective scenes.