3D Reconstruction from Multi-View Stereo: From Implementation to Oculus Virtual Reality

Anonymous CVPR submission

Paper ID ****

Abstract

Multi-View Stereo reconstructs a 3D model from images. Each image is a projection of a 3D model onto the camera plane ($\mathbb{R}^3 \rightarrow \mathbb{R}^2$), which inherently results in a loss of information. With enough images taken from a variety of perspectives, an reasonable model of the original scene can be reconstructed. These reconstructions usually begin by determining where each image was taken from. Once the cameras are calibrated, a dense, colored point cloud can be generated. A mesh can be fit over the point cloud to represent structures in the original scene. This entire process can be visualized through an Oculus Rift.

1. Introduction

Most current approaches to Multi-View Stereo can be broken down into three steps: feature matching, camera calibration, and dense reconstruction. In pipelines which result in a point cloud, surface reconstruction is applied as a final step.

1.1. Feature Matching

The first step in most Multi-View Stereo pipelines is finding correspondence points between images. Once we know these points, we can determine the relative position of the cameras which took the images.

First, we select features of interest in each image. There exist numerous algorithms for selecting these points (SIFT, SURF). The general idea behind most of them is to find a feature which is unique enough such that finding a similar feature in another images indicates with high probability that the two features corresponds to the same object in the scene.[1]

Once features have been extracted from each image, pairwise matches must be found. Matches will not exist for all features, so some criteria must be specified for when to accept a match. One such criteria is to match two features if the first is the best match for the second, and the second is the best match for the first. Another approach is to match one feature to another feature if the second best match is a much worse match than the best match.[1]

1.2. Camera Calibration

Now that we have correspondence points, we want to compute a homography relating one image to another. If we only wanted to find the orientation of one camera relative to another, we could use RANSAC to fit a homography (with the Discrete Linear Transform to make it linear).[2] However, if we only found the optimal pairwise relative positions, we would not be guaranteed that they would be consistent.

Instead, we want to find the *global* optimal camera positions. This process is known as Sparse Bundle Adjustment, and can been seen as minimizing a series of nonlinear equations. The LevenbergMarquardt for nonlinear least-squares is commonly used as a subroutine. Sparse Bundler Adjustment incrementally alters the positions of the cameras to as to minimize the *reprojective error* of the found correspondence points with respect to the images in which they appear. At the end of this process, we have a calibration matrix for each camera, relating the pose of each camera to a global coordinate system.[9, 7]

1.3. Dense Reconstruction

We can now reconstruct the scene from calibrated cameras. We want to find eventually output a scene which is *photo consistent*. Common approaches to this problem include: (1) building up a scene from points whose locations are found by triangulating between images; (2) starting with a volume which encloses the region of interest, and removing *voxels* which are not photoconsistent; and (3) generating stereo depth maps for pairs of images, and then fusing them together.[4]

1.3.1 Point Based Approaches

Once we have calibrated cameras with a sparse reconstruction, we can search along equipolar lines to find more correspondence points. The real-world locations of these points can be found through triangulation. In Patch-based Multi-View Stereo, these points were further expanded to *patches* which included a color and normal vector.[4]

1.3.2 Volumetric Approaches

Another approach to dense reconstruction is to start with a volume which encloses the region of interest, and iteratively remove small sections (*voxels*) which are not photoconsistent. Constructing the initial visual hull requires segmenting the input images into foreground and background.[6]

1.3.3 Stereo Depth Approaches

A final approach is to build off research in stereo matching. Here, we generate depth maps for all pairs of images with overlapping fields of views. Methods include SemiGlobal Matching, Graph Cuts, and Dynamic Programming.[5, 8] These depth maps can be fused to extract the structure of the scene.[3]

1.4. Surface Reconstruction

2. Our Approach

Figure 2. Sample input image (one of 312)

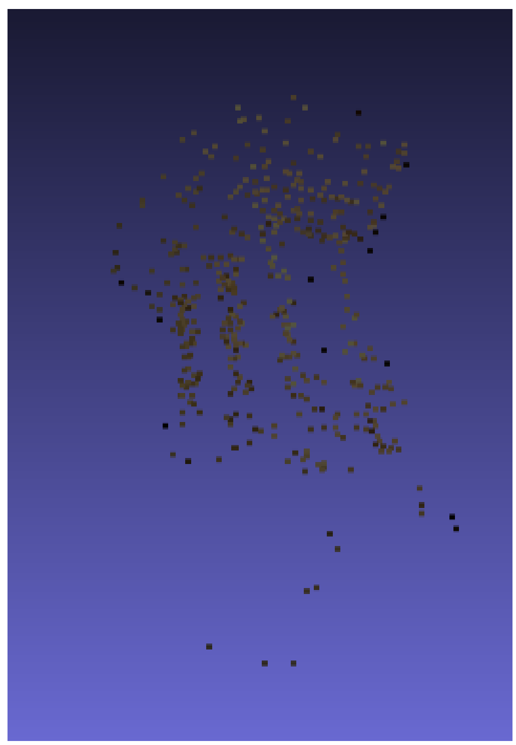

Figure 3. Sparse Reconstruction

References

[1] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *International journal of*

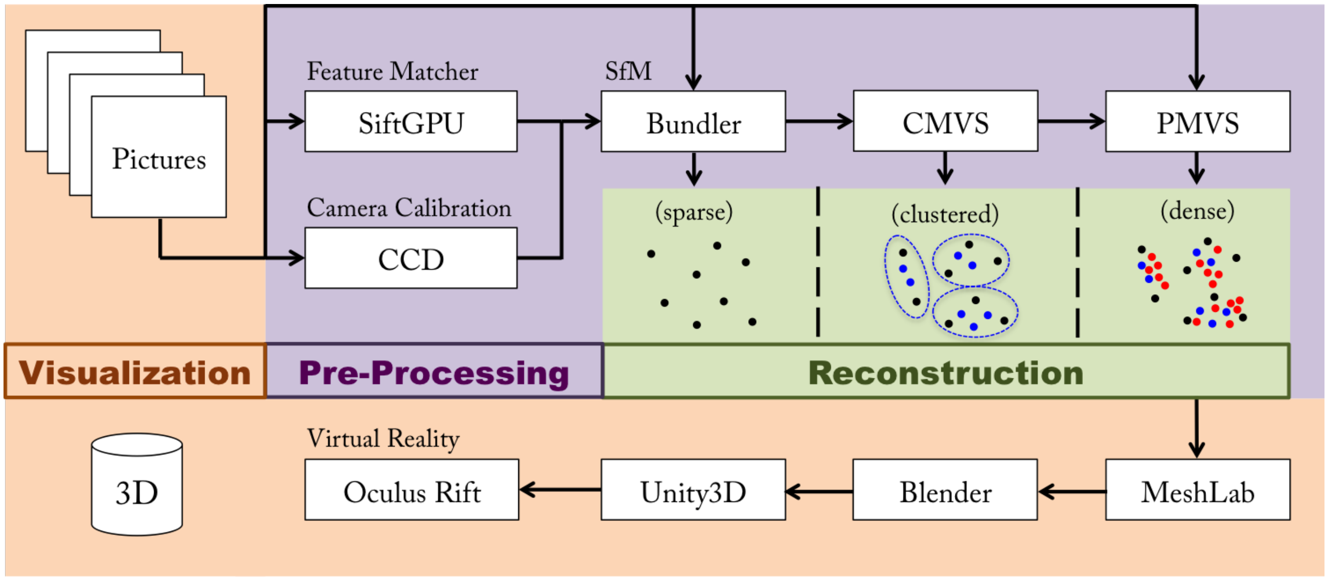

Figure 1. Proposed Multi-View pre-processing and visualization pipeline. After extracting features and camera calibration parameters from imported source images, point cloud reconstructions can be created from robust computer vision packages such as Bundler and CMVS/PMVS. As a result, 3D editing software tools such as MeshLab and Blender can assist in further configuration and production of detailed meshes/textures. These can be visualized in the Unity3D game engine, which supports Oculus Rift OVR integration.

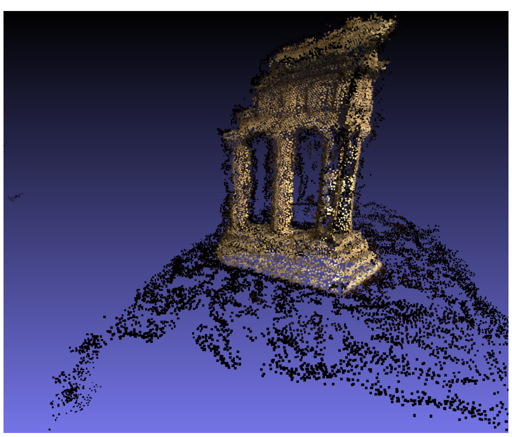

Figure 4. Dense Patch reconstruction

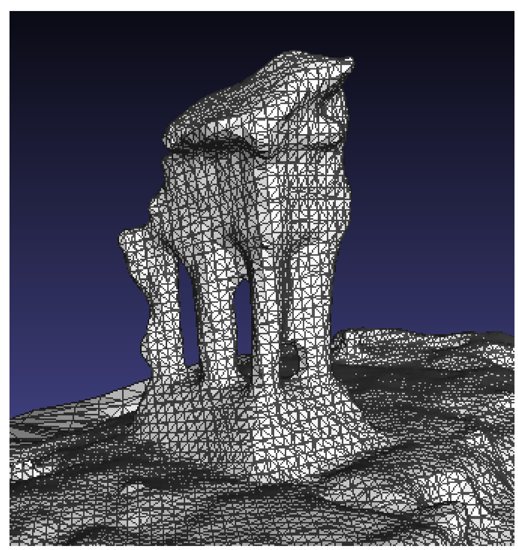

Figure 5. Poisson Surface Reconstruction to fit mesh to patches

computer vision, 74(1):59–73, 2007. 1

[2] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1

[3] S. Fuhrmann and M. Goesele. Fusion of depth maps with multiple scales. In *ACM Transactions on Graphics (TOG)*, volume 30, page 148. ACM, 2011. 2

[4] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1362–1376, 2010. 1, 2

[5] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2):328–341, 2008. 2

[6] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000. 2

[7] M. I. Lourakis and A. A. Argyros. Sba: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1):2, 2009. 1

[8] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Interna-*

tional journal of computer vision, 47(1-3):7–42, 2002. 2

[9] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. *ACM transactions on graphics (TOG)*, 25(3):835–846, 2006. 1