3DCV Homework 2 Report

R14922126 閻睿杰

October 2025

Contents

1 COLMAP

1.1 Structure from Motion with COLMAP

We begin by converting the video to a sequence of images using FFmpeg, with 5 images per second. This gives us around 150 images in total. Next, all the images are fed into COLMAP [3, 4, 5] to perform Structure from Motion (SfM). Figure 1 shows the reconstructed point cloud.

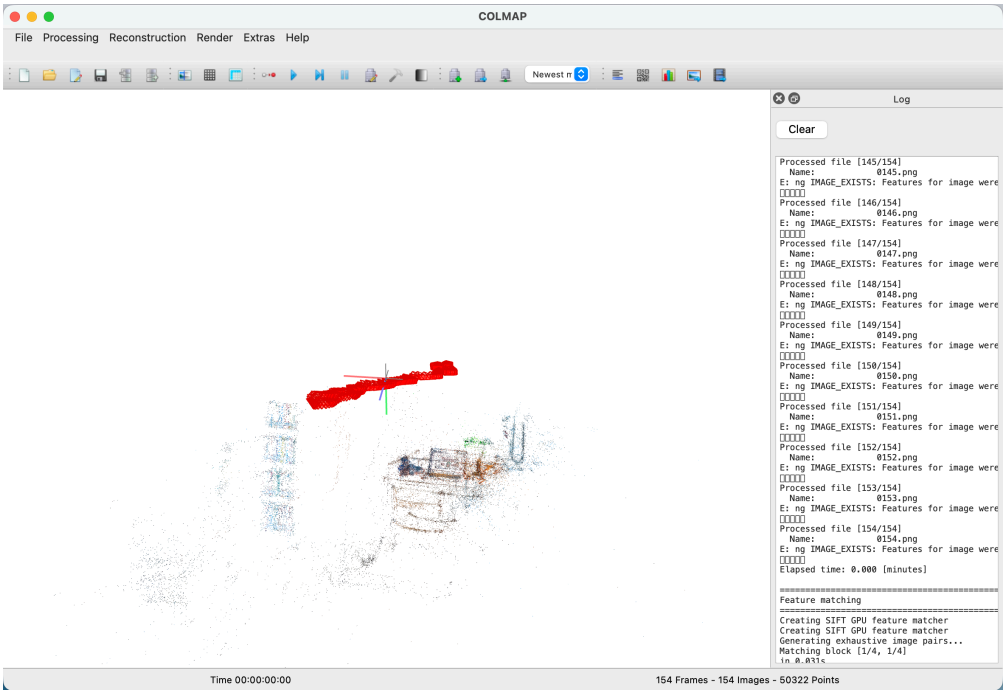

Figure 1: Point cloud of the scene produced by COLMAP.

1.2 Point Cloud to Triangle Mesh

We use alpha shape [1] to convert a point cloud to triangle mesh model. The result is illustrated in Figure 2.

Figure 2: Triangle mesh model from point cloud.

2 Camera Relocalization

2.1 Camera Pose Estimation and Visualization

We begin by performing descriptor matching with ratio test. For each query descriptor q we find the two nearest model descriptors m_1 and m_2. If $d(q,m_1) >= 0.75d(q,m_2)$, the match is discarded. Next, we solve the PnP problem [2] using a RANSAC [2] scheme of 100 iterations to deal with outliers. Solving PnP gives us the relative pose of each camera: rotation vector $\mathbf{r} \in \mathbb{R}^3$ in axis-angle representation, and translation vector $\mathbf{t} \in \mathbb{R}^3$. The median rotation error is around 0.00004 in radian, and the median translation error is 0.00012. Both errors are small, which demonstrates the robustness of RANSAC.

With $\mathbf{r}$ and $\mathbf{t}$ we can convert a point from camera coordinate frame to world coordinate frame:

$$\mathbf{x}_{\text{world}} = \mathbf{R}^\top(\mathbf{x}_{\text{cam}} - \mathbf{t}), \tag{1}$$

where $\mathbf{R} \in \mathbb{R}^{3\times3}$ is a rotation matrix that represents the same rotation as $\mathbf{r}$, $\mathbf{x}_{\text{cam}} \in \mathbb{R}^3$ is a point with camera coordinates, and $\mathbf{x}_{\text{world}} \in \mathbb{R}^3$ is the corresponding point with world coordinates. For each camera pose we select five points with camera coordinates as the vertices of the square pyramid for visualization:

$$\begin{aligned}
\mathbf{x}_{\text{center}} &= (0,0,0) \\
\mathbf{x}_1 &= (-a,-a,a) \\
\mathbf{x}_2 &= (-a,a,a) \\
\mathbf{x}_3 &= (a,a,a) \\
\mathbf{x}_4 &= (a,-a,a)
\end{aligned}$$

with the optical center $\mathbf{x}_{\text{center}}$ being the apex, and $a > 0$ so that the normal of the base will be the camera orientation. We set a to a small number such as 0.1 for better visualization quality. We convert these five points to world coordinates using Equation 1, and draw the pyramid alongside the point cloud. The result is illustrated in Figure 3.

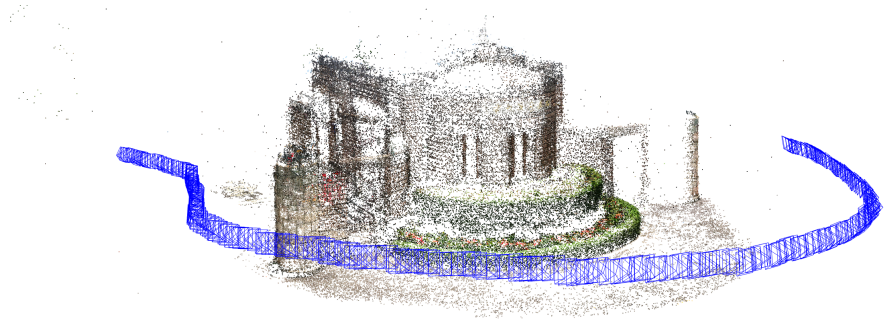

Figure 3: Visualization of the estimated camera poses.

2.2 Virtual Cube in AR Video

We create a cube which has 10×10 evenly space points on each side. To draw the cube in each frame, we first sort the points in descending order of their distances to the optical center. Then we convert each point to 2D pixel coordinates using the provided camera matrix and lens distortion coefficients. Finally, we draw the point onto the frame. Figure 4 illustrates the result.

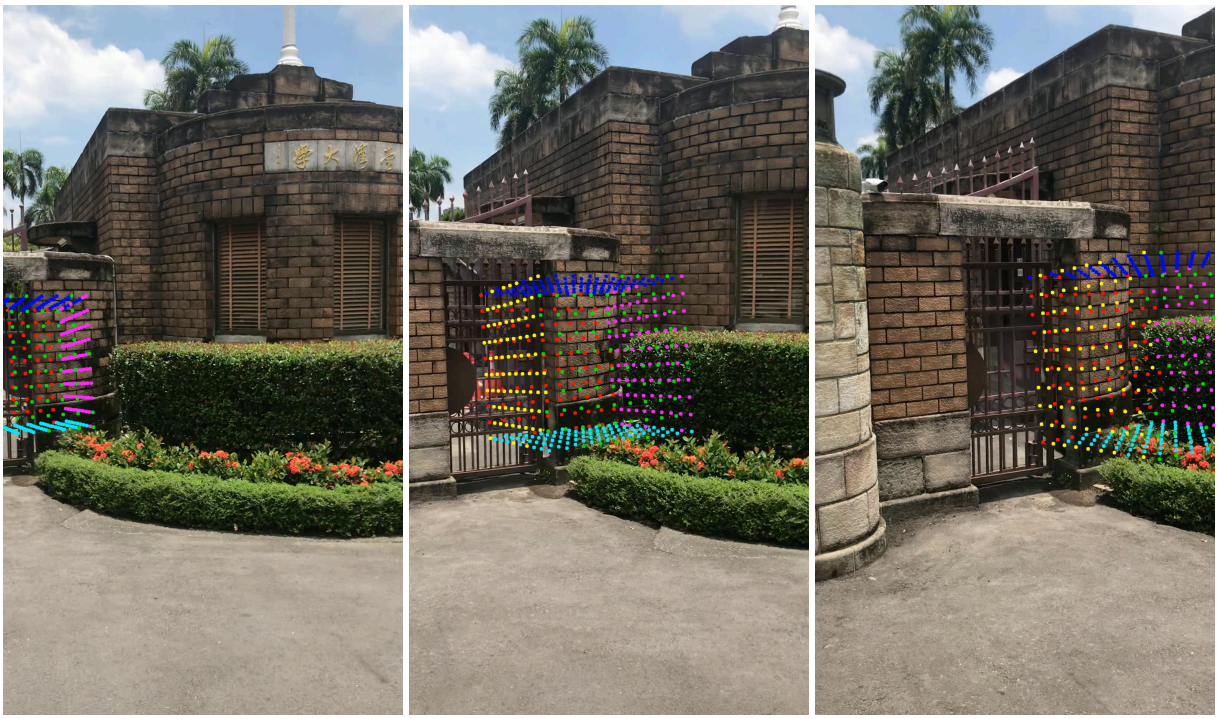

Figure 4: AR video with a virtual cube.

References

[1] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.

[2] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[3] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] Johannes Lutz Schönberger, True Price, Torsten Sattler, Jan-Michael Frahm, and Marc Pollefeys. A vote-and-verify strategy for fast spatial verification in image retrieval. In *Asian Conference on Computer Vision (ACCV)*, 2016.

[5] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.