In this supplementary material, we aim to show the effectiveness of the proposed 4-view triangulation problem using a synthetic multiview dataset from [2] and a real dataset [4]. Evaluating from a synthetic multiview dataset, we randomly select 200 points from random 4 pairs of views. To simulate the noise on the observed points and the calibration, the poses are perturbed with $\mathcal{N}(0, 2.0)$ and the observed points are perturbed with $\mathcal{N}(0, 1.0)$. We compare our 4-view triangulate result with 3-view triangulation [3], Figure 1, using the proposed GPU-HC. It is clear to see that the error between the optimized 2D points and the ground truths of 4-view triangulation is smaller than 3-view triangulation. Quantitatively, the mean error of 4-view triangulation is 2.9712 pixels, while in 3-view triangulation, the error is 4.2074 pixels. Therefore, adding one more view to do triangulation would drop the projection error significantly.

![Figure 1: The 2D error histogram of 3-view and 4-view triangulation with synthetic Multiview curve dataset [1].](image1)

Employing a real dataset, we use the dinosaur sequence from [4] which contains 36 images with 4,983 corresponding points. To deploy our 4-view triangulation method, only 1,516 points that are co-visible by more than 3 views are used. The triangulation result is shown in Figure 2. Note
that after GPU-HC computes the optimal positions of the image points, the projection lines from all four images always intersect in the 3D space. In such a case, we are free to select any two views to find the position of the 3D points given the camera extrinsic matrix of these two views. The processing time of this whole sequence using the proposed GPU-HC is around 10.6 seconds.

References


