Tzu Hsuan Huang

Porfessor Berg

CS 117

15 Decemeber 2023

Final Project

This final project follows the default project that Professor Berg provided. Its goal is to build a high-quality 3D model of an object using the pipeline outlined in the CS 117 class. Professor Berg provided three objects, and each one has multiple scans. In this project, the 'couple' object will be used, and the code will be implemented in a Jupyter notebook using Python. MeshLab will also be utilized for mesh alignment and screened Poisson surface reconstruction.

The first step of the project is to perform calibration. I utilized the function from the previous assignment to obtain the focal length, principal points, rotation vector, translation vector, and distortion matrix for both the left and right cameras, storing these values for further use. The focal length for both cameras is 1561.01, and the principal points are [1021.15, 755.84]. This step proceeded without any issues, and the results turned out well.

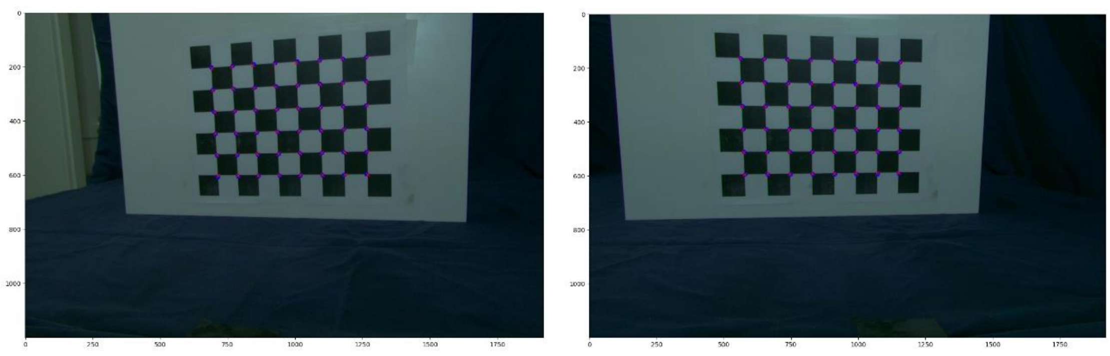


Figure 1 Calibration results

The subsequent step involves reconstruction, which relies on the 'decode' function for processing images with projected gray code, resulting in an array of decoded values (0 to 1023) and a binary mask denoting reliably decoded pixel. In the final project, colormasks are introduced, comparing colors through squared Euclidean distance to quantify differences between colors in two images, enabling localization of the object. Figuring out how to compute this took a lot of time, but after doing some research, the code is finished. The Reconstruct function takes a set of images from a pair of cameras and produces 3D point coordinates. It also computes the average pixel value from the two images.

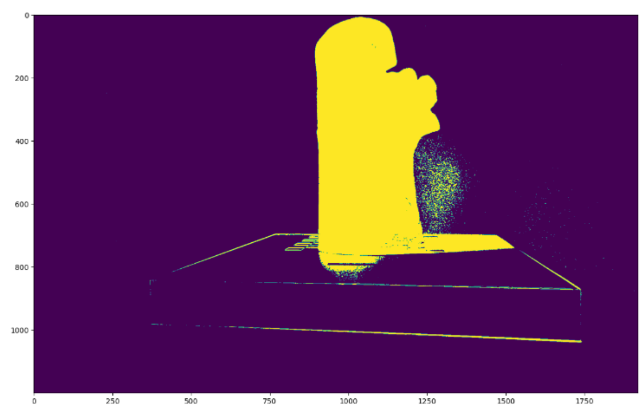


Figure 2 Colormask Example

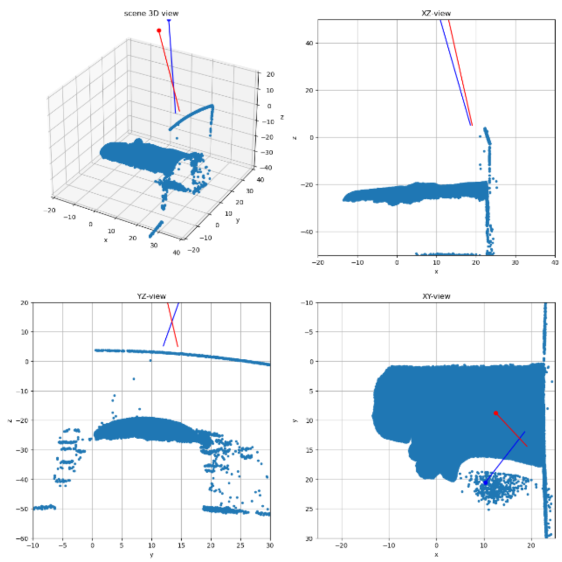


Figure 3 Visualize reconstruction

After the reconstruction, the process requires mesh cleanup. In Assignment 4, I successfully implemented bounding box pruning, triangle pruning, and removal of noisy points from the meshes. I created a new function called 'getmesh' to include all mesh cleanup steps from the previous assignment and added a neighbor smoothing algorithm to refine the meshes. The code defines a function to find neighboring vertices in a 3D mesh. In a loop, each vertex in the mesh is smoothed by averaging its coordinates with those of its neighbors, resulting in a smoother 3D mesh stored in the pts3 array. I used this algorithm in getmesh function three times to achieve smoother meshes. Finally, use the writeply function, provided by Professor Berg, and turn these values into .ply 3D model files.

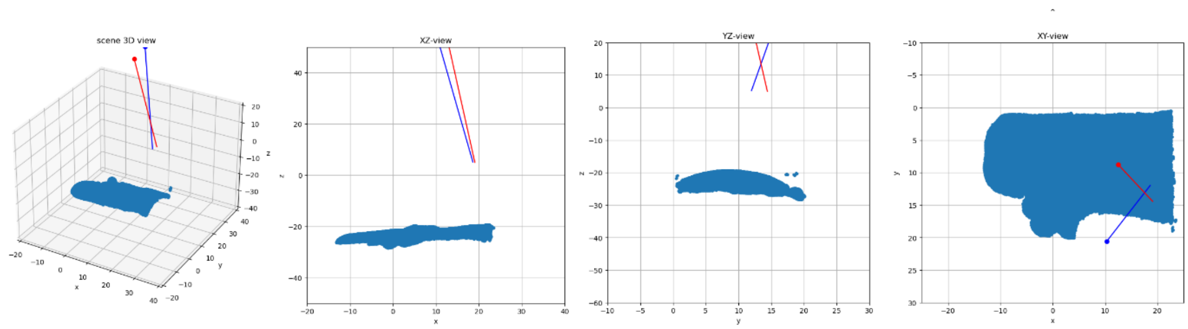


Figure 4 mesh after cleanup

Next, utilize the **reconstruct** and **getmesh** functions to generate a .ply file for the remaining four scans. Due to missing images in the 'grab_5' folder, I skipped processing these scans. Finding the optimal values for box limitation, threshold, and trithresh to achieve the best outcome for each scan consumed the majority of my time. Some require a larger 'trithresh,' while others require a smaller threshold. It took me over two days to thoroughly test and adjust these values.

After completing all the meshes, alignment was done using a software called MeshLab. This was my first time using the software, so I watched YouTube tutorials to understand the alignment process. Initially, I attempted point-based gluing, but it didn't produce satisfactory results, prompting me to manually align all the meshes. For improved alignment, it was suggested to use vertex color instead of the default mesh color. After the alignment, I conducted the screened Poisson surface reconstruction and experimented with various input values to generate different outputs of Poisson meshes. Ultimately, the optimal input values are a reconstruction depth of 11, a minimum number of samples set to 15, and enabling pre-cleaning. The initial outcome(Figure 6) was unsatisfactory, leading me to revisit the reconstruction step and adjust some values to obtain cleaner and clearer meshes. Subsequently, I realigned the meshes and performed the screened Poisson surface reconstruction once more. The results were an improvement over the initial attempt.

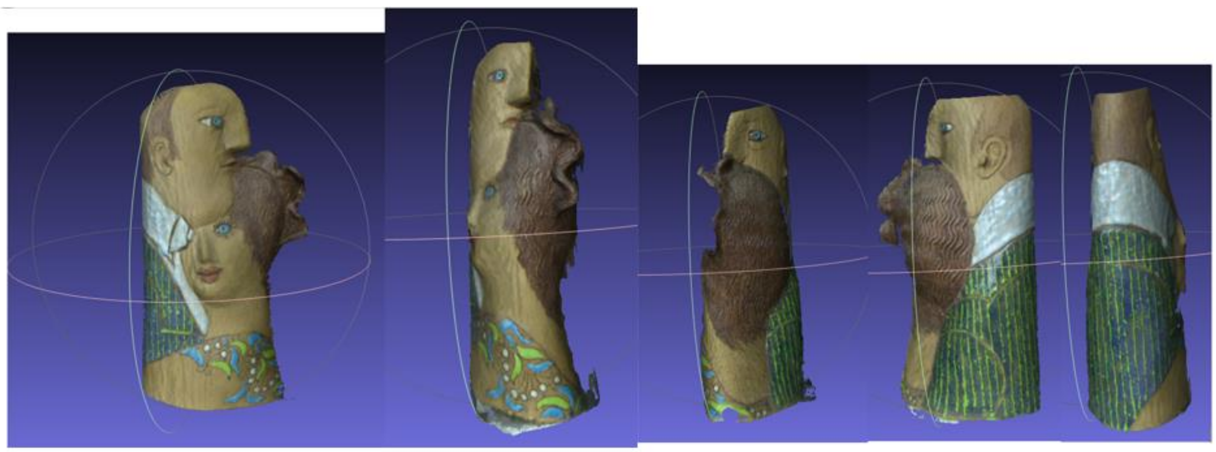
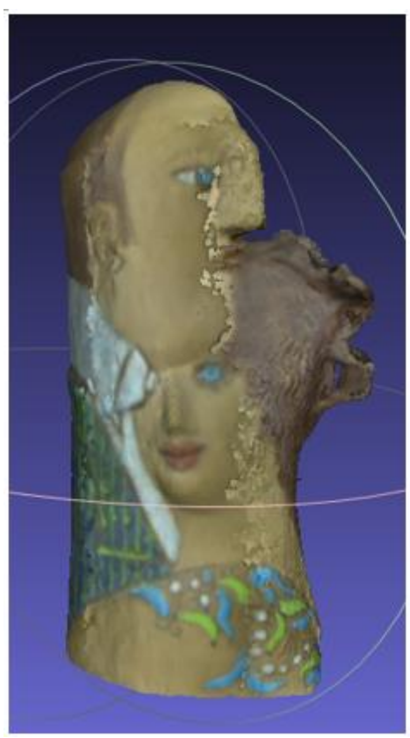


Figure 5 meshes from each scan



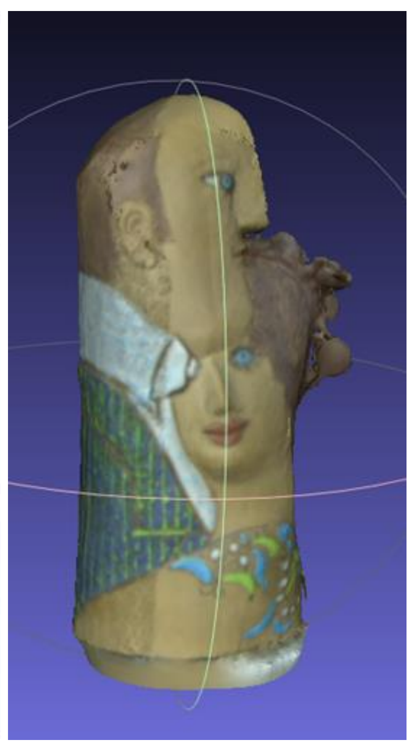


Figure 6 First result *Figure 7 Second result*

The main challenge I faced in this project was determining the optimal input values for each function and in MeshLab. The only solution was to invest time in testing every possible value and comparing the results to identify the best ones.