Amanda Williams

Robotic Visions

Professor Mingshao Zhang

04/22/2024

Visual S.L.A.M. Project

The first step that the demo takes is to download the demo tar file that contains the photo set of the office and save it to a temporary folder. From this folder, the images and data are extracted to create an object that Matlab can use. This is where the map can begin materializing.

The first image of the 3D map is brought forth and various data points and landmarks are chosen. Landmarks on the second image are then found, and by using the *matchfeature* function, the two images can begin to be estimated to fit together. One estimation technique is a simple homography, where one picture simply needs to be resized or translated. If that cannot join the pictures together, then a fundamental matrix must be used to estimate the 3D rotation and lighting effects of the points of interest. After these assumptions are produced, the matched feature points are lined up using the *triangulate* function and a 3D space can be represented with the cameras' positions. This is looped for every picture taken.

Once the map is initialized, the two initial frames calculated are then used to set the key points and baseline for camera positions. Then, the loop of combining all of the pictures takes place, with the code naming and then calling forth various identified objects that can be used as reference points and creating the database of landmarks. Then, the images are refined, and errors are minimized. After, a map within Matlab is created to help visualise the position of the landmarks and the camera. To create the path that the camera is estimated to have taken, tracking is now introduced. This process is to ensure that key frames are not missed, and the video can be run smoothly without skipping any pictures. This combines all images and camera poses into a 3D map of the environment and subsequent poses of the camera. With this process complete, a ground truth can now be uploaded and compared to the estimations of the Matlab program. Then, the code is converted to C++ to help with image processing, and the estimated best path is calculated and presented.

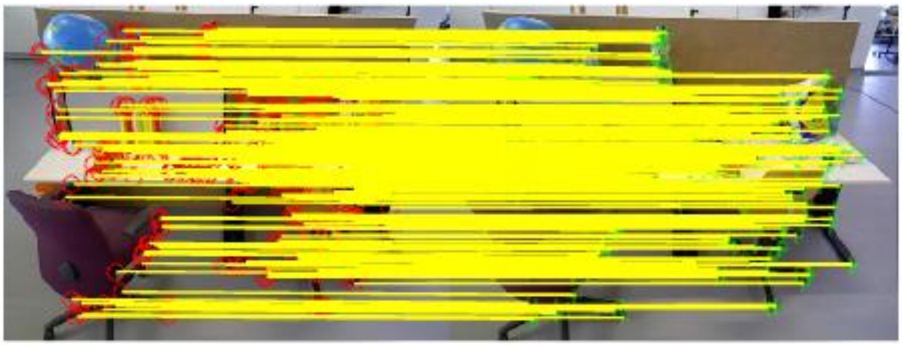

Figure 1: Showing Matching Landmarks

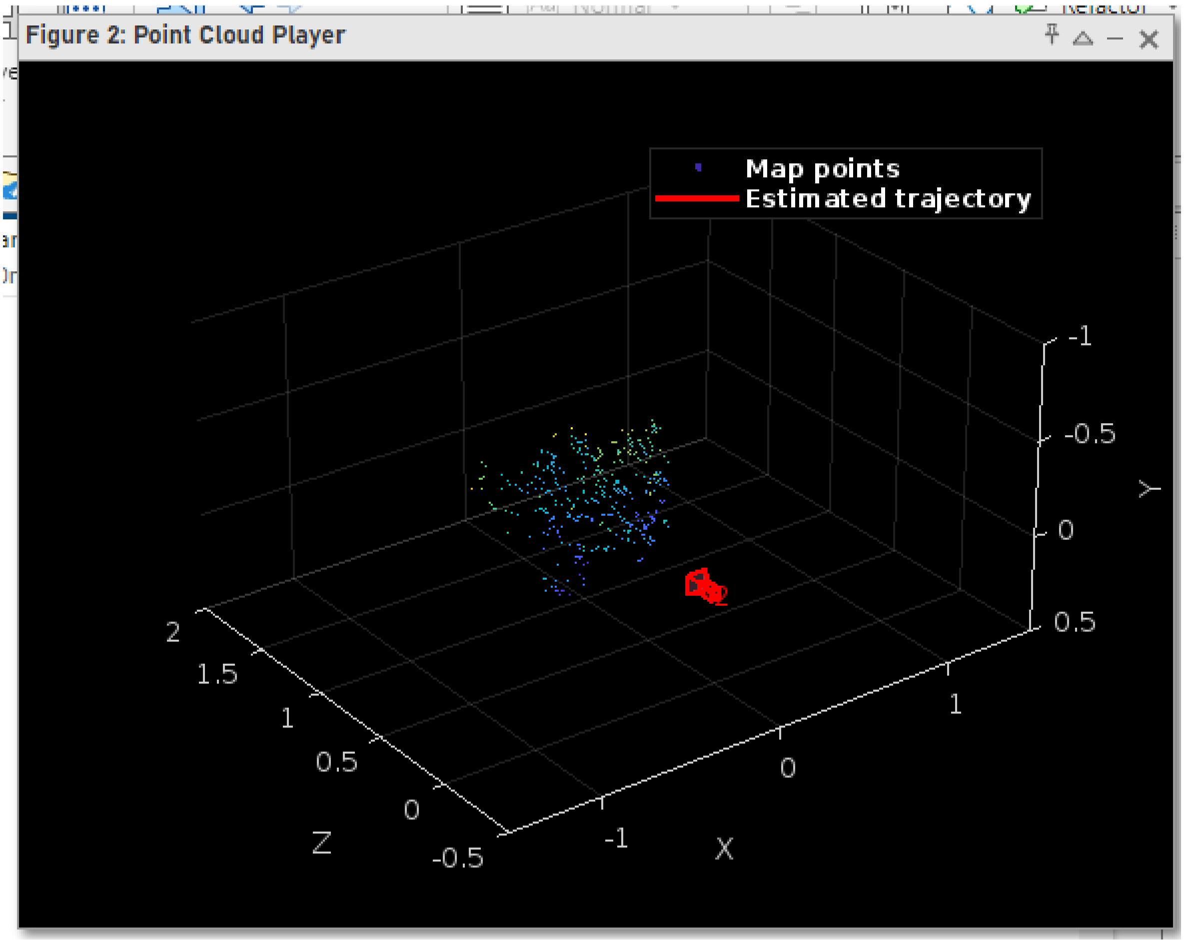

Figure 2: Point Projection and Camera Pose

Figure 3: Landmark Points

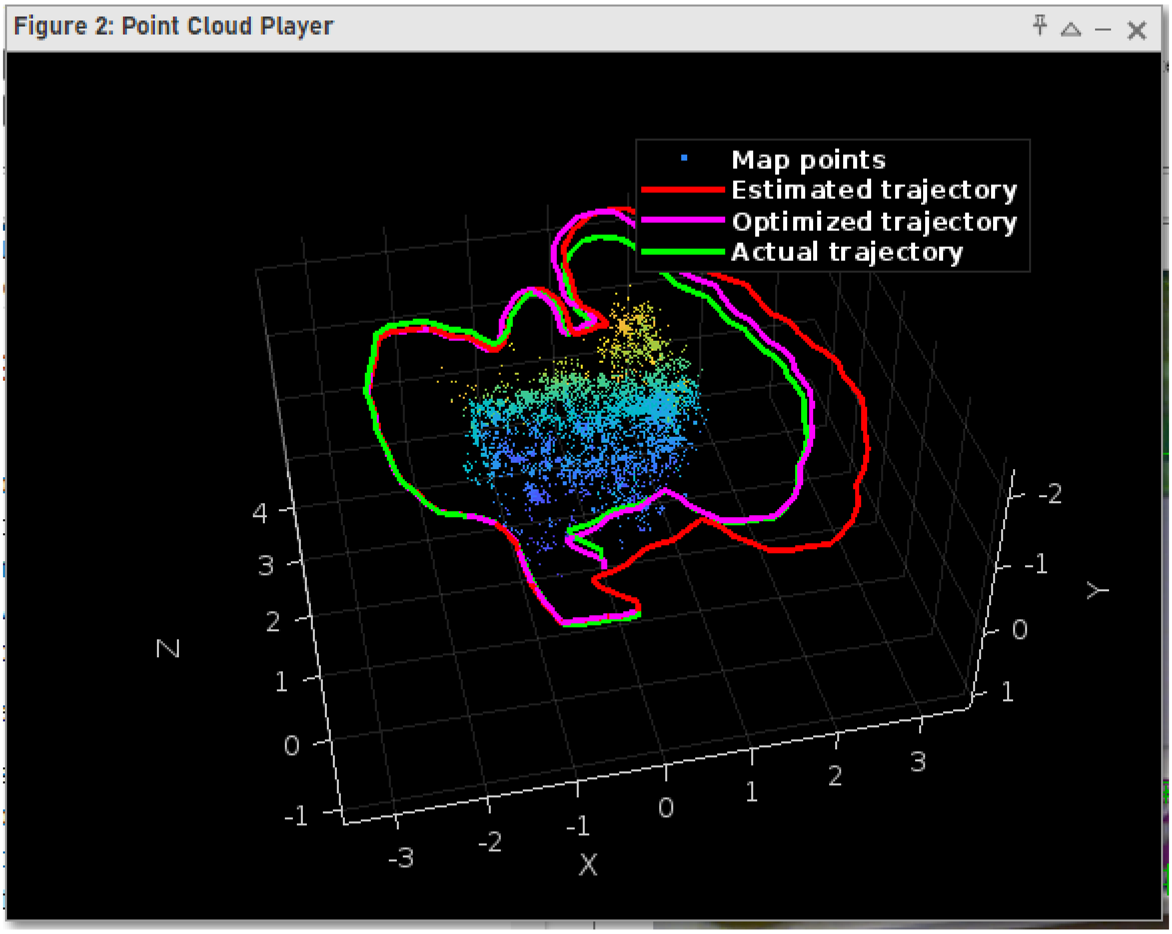

Figure 4: Estimated, Optimized, and Actual Paths

Part 2

Completing the 3D map was not possible due to implicit data missing from the uploaded photos. The proof of ownership for my pictures is my bookbag and water bottle.

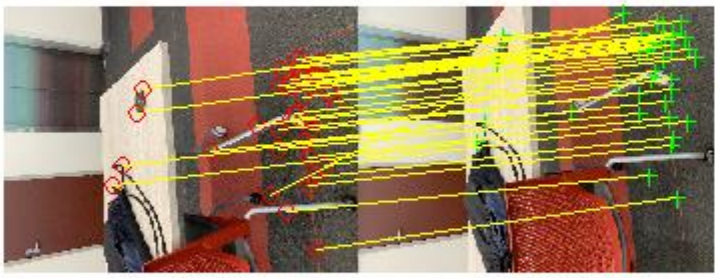

Figure 5: Landmark Data Matching

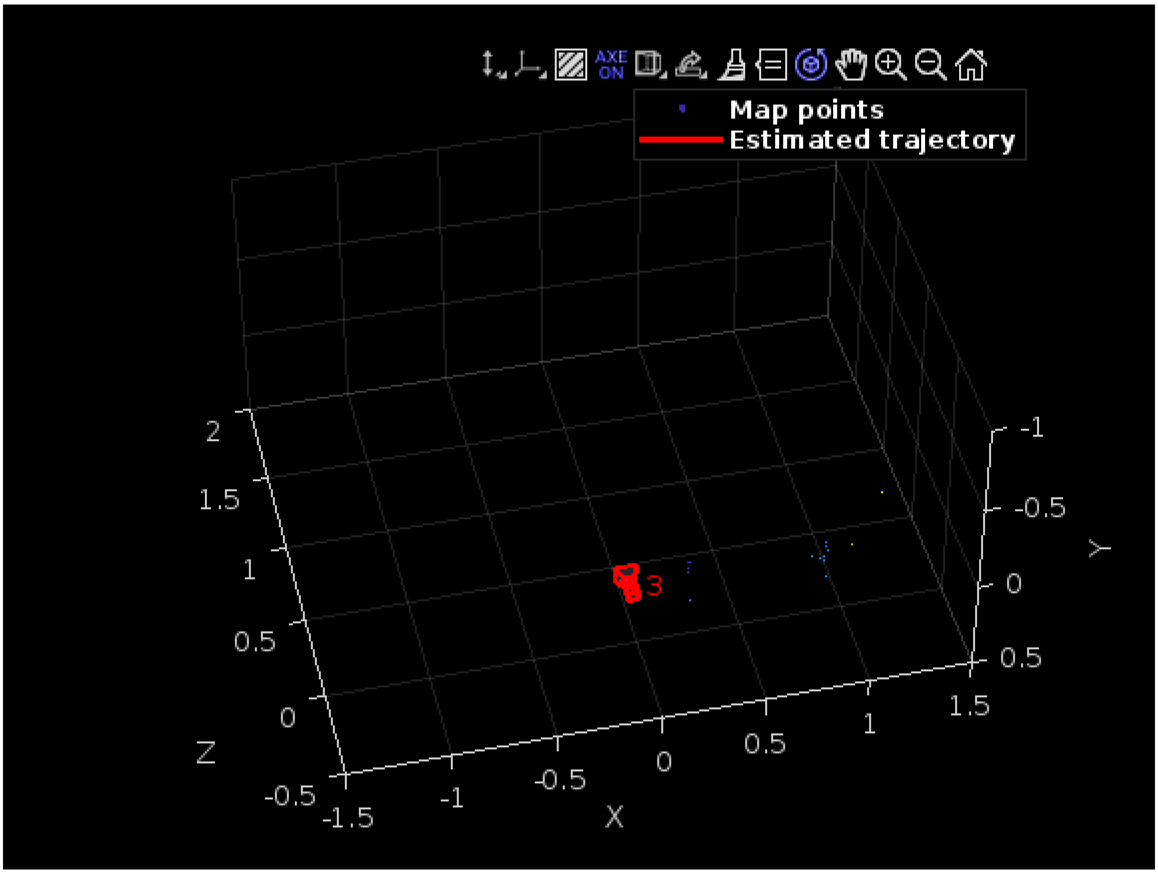

Figure 6: 3D Data Map - Only 3 Loops Possible