Planar Monocular SLAM

Honor Duthie

May 2025

1 SLAM Pipeline and Experimental Process

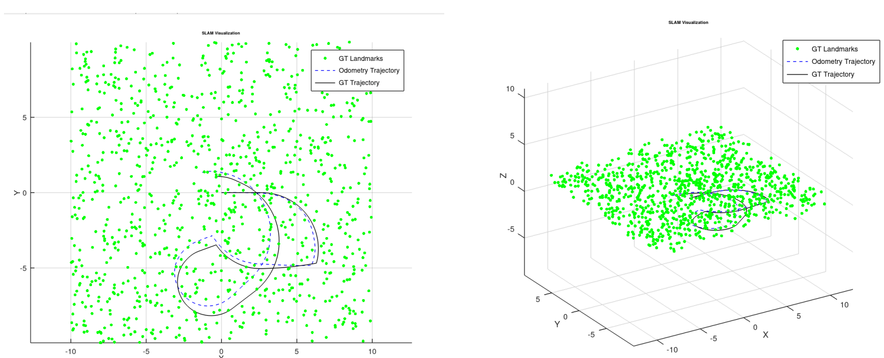
1.1 Data Loading and Inspection

The program starts by loading the dataset, which includes:

- `camera.dat`: camera intrinsics and extrinsics
- `trajectory.dat`: odometry and ground truth poses
- `world.dat`: ground truth 3D landmark positions
- `meas-*.dat`: image plane observations

To verify successful data parsing, I visualize:

- Odometry vs ground truth trajectories
- All known ground truth landmarks



(a) From above (b) From the side

Figure 1: Loaded data

1.2 Triangulation attempts

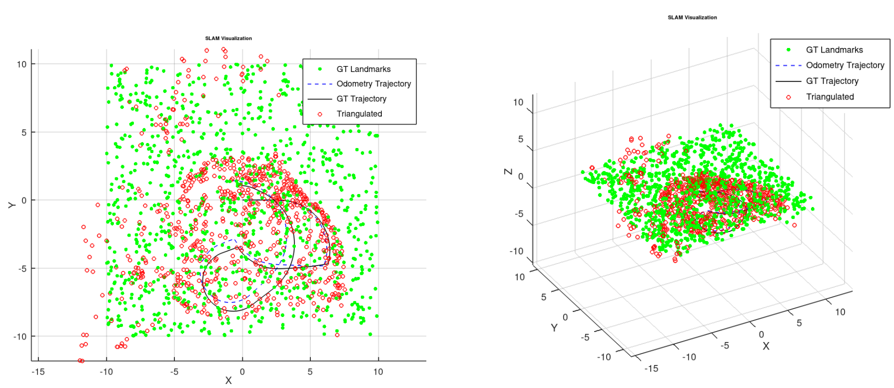
I implemented three triangulation methods to estimate initial 3D landmarks:

- **triangulate_simple:** Simple DLT triangulation using all valid observations. Landmark RMSE: 2.4579 units
- **triangulate_all:** Same as above, but filter landmarks using minimum parallax between views. Landmark RMSE: 2.5302 units
- **triangulate_best_pair:** Triangulate each landmark using the best pose pair (max parallax). Landmark RMSE: 7.5519 units

All triangulation functions return both the 3D point and its associated landmark ID, enabling consistent indexing during bundle adjustment.

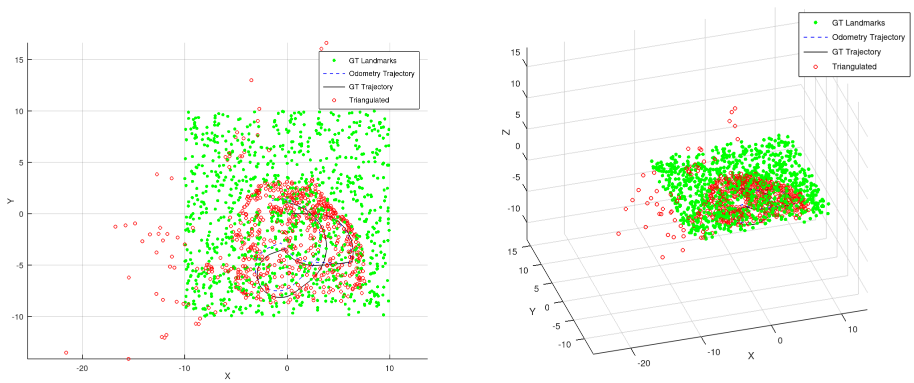
All the triangulation methods were able to correctly triangulate the landmarks given the ground truth trajectory.

The method that provides the best results, both visually and in terms of RMSE, is `triangulate_simple`.



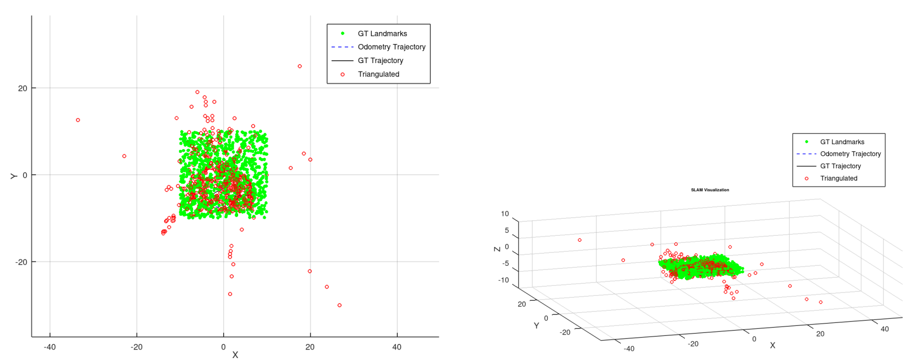
(a) Triangulate_simple from above (b) Triangulate_simple from the side

Figure 2: Results of triangulate_simple



(a) Triangulate_all from above (b) Triangulate_all from the side

Figure 3: Results of triangulate_all



(a) Triangulate_best_pair from above (b) Triangulate_best_pair from the side

Figure 4: Results of triangulate_best_pair

1.3 Root Mean Square Error

To quantitatively assess the accuracy of estimated 3D landmark positions, I compute the Root Mean Square Error (RMSE) between the estimated and ground truth landmark coordinates using the following function:

- The function `evaluate_map(landmarks_est, data)` compares estimated 3D landmark positions (`landmarks_est`) against ground truth positions stored in `data.world`.
- It first extracts landmark IDs and their positions from both estimated and ground truth sets.
- Using `intersect`, it identifies common landmark IDs and aligns the corresponding 3D coordinates.

- It then computes the squared Euclidean distance for each matched pair and averages the result to produce the RMSE:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i^{\text{est}} - \mathbf{x}_i^{\text{gt}}\|^2}$$

  where $\mathbf{x}_i^{\text{est}}$ and $\mathbf{x}_i^{\text{gt}}$ are estimated and ground truth positions of the i-th landmark, respectively.

1.4 Reprojection Error

In addition to landmark RMSE, I compute the reprojection error for each observed landmark prior to bundle adjustment to assess the initial quality of triangulated points:

- For each image observation in `Zp`, I identify:
  - the pose index (`pose_idx`) and landmark index (`lm_idx`) via `projection_associations`.
  - the camera pose estimate (`XR_guess(:,:,pose_idx)`) and the 3D landmark estimate (`XL_guess(:,lm_idx)`).
- I then project the 3D point into the image plane using the function `projectPoint`.
- The reprojection error is calculated as the Euclidean norm between the projected point and the observed measurement:

$$\text{error} = \|z_{\text{projected}} - z_{\text{measured}}\|$$

- These errors are collected in a per-landmark array `errors_per_landmark`, allowing analysis of which landmarks contribute large residuals.

This step helps identify poorly triangulated landmarks or incorrect data associations prior to optimization. Large reprojection errors often indicate inconsistency between geometry and observation, which can destabilize bundle adjustment.

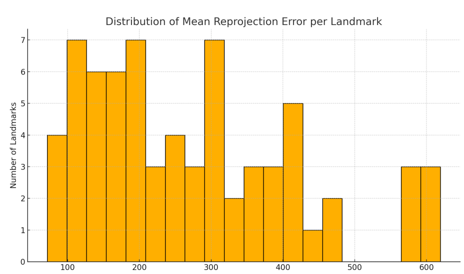


Figure 5: Results of reprojection error for triangulate_simple

Despite attempting different forms of triangulation, the results reveal that errors remain persistently high, with the majority of triangulated landmarks exhibiting mean reprojection errors exceeding 150-200 pixels, and several exceeding 400 or even 600 pixels. This indicates substantial inconsistency between estimated 3D geometry and observed image measurements, likely due to poor initial triangulation, incorrect data associations, and noisy odometry.

1.5 Bundle Adjustment Preparation

Bundle adjustment requires aligning several data structures:

- `XR_guess`: initial SE(3) poses from odometry
- `XL_guess`: triangulated landmarks
- `Zp, projection_associations`: image observations and their associated pose and landmark indices
- `Zr, pose_associations`: odometry-based relative transformations

Landmark IDs are consistently remapped to compact indices to match the structure of `XL_guess`.

1.6 Bundle Adjustment Configuration

The implementation is based on the provided code from the Probabilistic Robotics repository with some changes to support:

- Robust kernel thresholds to down-weight outliers
- Diagonal damping
- Gauge fixing by anchoring the first pose to identity

The optimization should minimize reprojection and pose error jointly over multiple iterations.

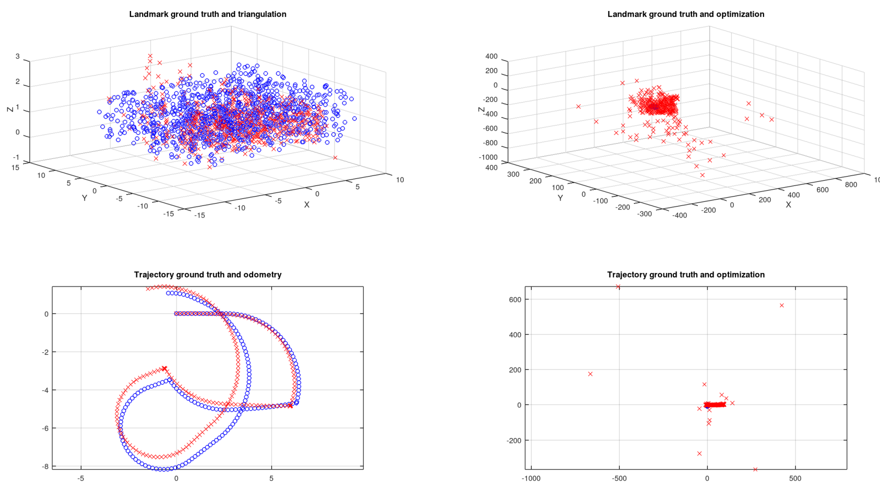


Figure 6: Results of bundle adjustment using total_least_squares with triangulate_simple after 5 iterations

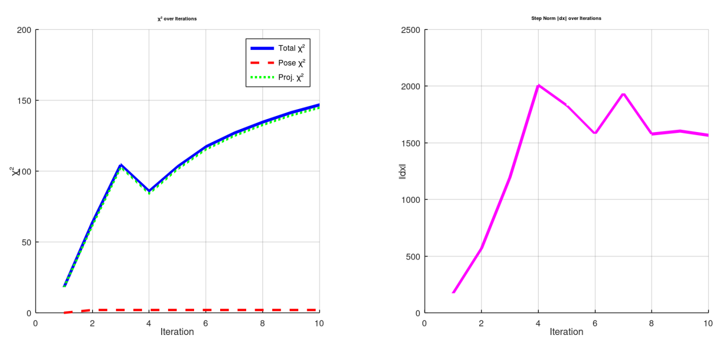


Figure 7: Convergence diagnostics for bundle adjustment after 10 iterations. **Left:** Total χ^2 error and its components (pose, projection) over iterations. The increasing trend indicates divergence rather than convergence. **Right:** Step norm $\|\Delta x\|$ per iteration remains large, suggesting unstable or poor optimization behaviour.

1.7 Results and Observations

I observe that:

- The average reprojection error is high across many landmarks.
- Only 3 out of hundreds of landmarks have reprojection error < 100 pixels.
- The optimization is numerically unstable and divergent.

I hypothesize that the persistent issues in optimization are due to one or more of the following factors:

- The triangulation is too poor to be used as an initialisation to the optimisation
- An error has been unknowingly introduced to the given bundle adjustment code
- At some stage there could be a mismatch of data association
- Attempting to perform bundle adjustment on the entire trajectory is the wrong approach and a windowed approach should be tried.