Mini-project of Vision Algorithm for Mobile Robots

Video address at Youtube

https://www.youtube.com/playlist?list=PLFy5hlHdq7W6l5G42UbXeH3ER4VQeBkfE

Pipeline and implementation

- Set bootstrap frames with appropriate baseline
- Use Harris detector to get keypoints features in the first two frame
- Estimate essential matrix and decompose E to get the unique pose by disambiguating from 4 possible poses
- **Obtain the initiate landmarks by triangulation (with Bundle Adjustment)**

Here to begin iteration:

- For a new image, use KLT tracking to get the corresponding keypoints in the new image
- Use ransacLocalization() to obtain the pose of the current camera
- Use KLT to track kpt in the candidate kpt set, process triangulate check in each tracked candidates in the current frame
- **Triangulate the new points and landmarks from the candidate set** (for who pass the angle test and re-projection test)
- Update the state (image, kpts, landmarks, kpts not matched, valid tracked kpts, camera poses)

Functions:

The main functions of the project is listed below. They are used from MATLAB box or implementation of our exercises:

- **detectHarrisFeatures**: a series of functions that establish correspondences of keypoints for bootstrapping
- **vision.PointTracker**: used to track keypoints across images implemented with Lukas Kaneda Tomasi tracker
- **estimateFundamentalMatrix**: returns the estimated Fundamental Matrix together with the inliers used to compute it
- **decomposeEssentialMatrix & disambiguateRelativePose**: returns the relative rotation and translation between camera poses given the Essential Matrix, the camera parameters and the keypoints in each image
- **ransacLocalization**: returns the orientation and location of a calibrated camera in world coordinates given the 2D-3D point correspondences and the camera parameters. It uses the P3P algorithm to estimate the pose, while eliminating outliers using the RANSAC algorithm
- **triangulate**: returns the 3D points given 2D point correspondences and the camera projection matrices, refine the triangulated landmarks with BA (levenberg-marquardt)

Summarize

- how your implementation deviates or expands upon the recommendations written below

At the begin, I use simply DLT for RANSAC localization, and I make sure that it's inappropriate in this task, so I turn to P3P with adaptive iteration.

For a long time, I don't know how to use the 'PointTracker', so I use the KLP robust version that I write in the exercise, but I found it's slow and not good.

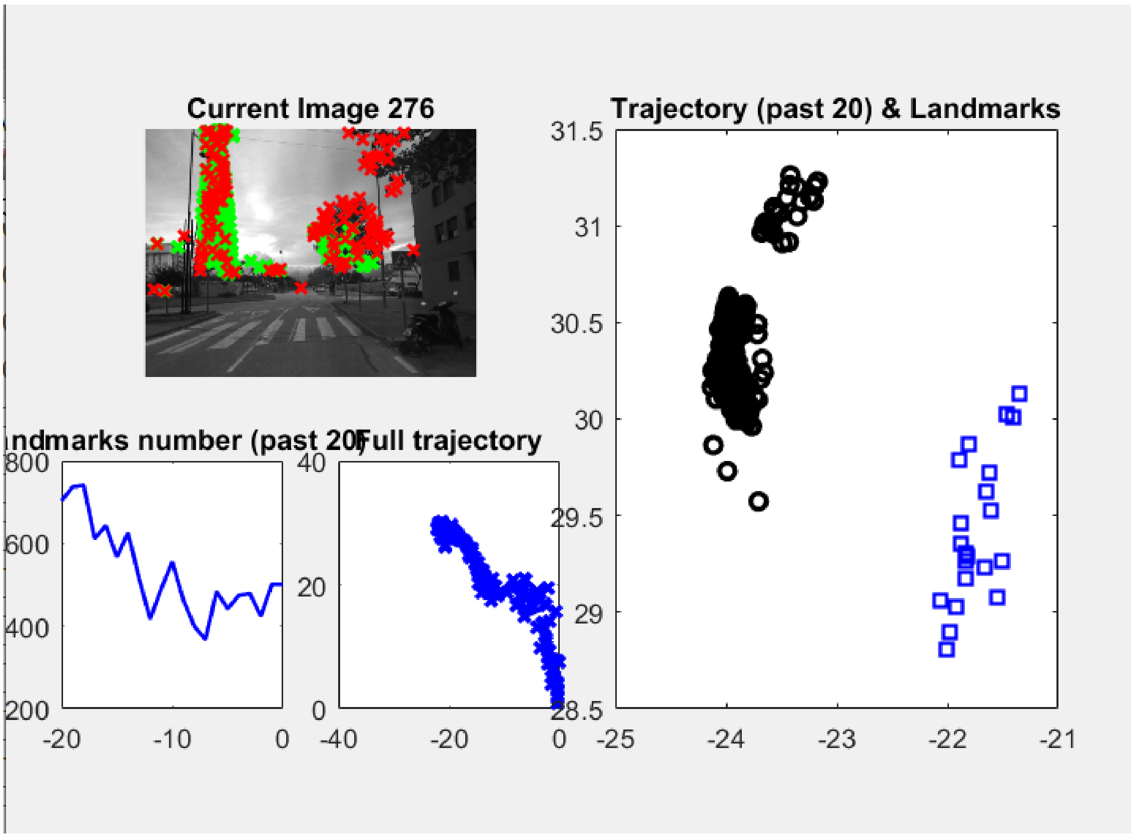
- Milestones set and achieved

  Milestones:

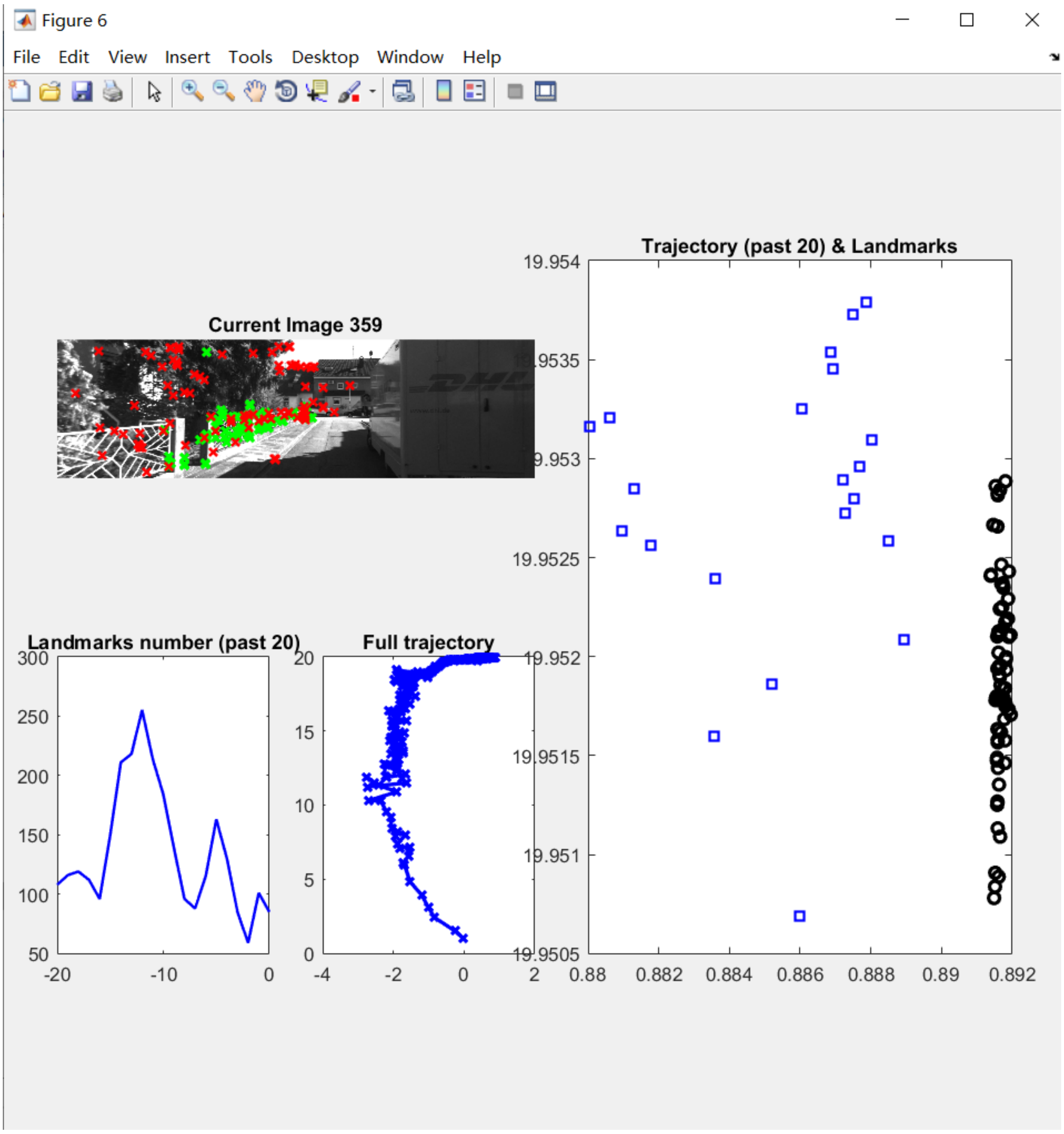
  1. invoking Harris detector and get keypoints from both images in the bootstrap period and corresponding matches.
  2. successfully calculate R | t from the bootstrap images and triangulate the first generation of landmarks
  3. successfully run a iteration of continues VO
  4. obtain a relative 'acceptable' full trajectory
- Problems encountered and solution

Results and screencasts for each dataset

- Malaga



- KITTI



Author Contributions

- Zilong Deng - zilong.deng@uzh.ch

  Part of functions implementation, combination of building blocks, debug

- Xuanang Lei - xualei@student.ethz.ch

  Part of functions implementation, writing reports