Vision Aided Navigation 2024
Project Report

Over the semester we implemented a system that estimates the trajectory of the vehicle from a video captured with an onboard stereo camera.

The system uses a suite of algorithms in several stages to achieve this goal. In the final project report we present the different parts of the system, explore their properties and the quality of the estimation.

Present in your own words the different stages of the system. Summarize the ideas and algorithms as you comprehend them. The discussion should include:

1. Introduction and overview: provide a brief background and explain why the problem is important.

2. Code: How is the code organized – What functions are used to implement the main stages of the process. The code should be readable and well documented.
Supply the specific locations (function name, file name, line number) for the following stages:

- Triangulation
- RanSAC
- PnP trajectory calculation
- DataBase definition
- Adding a frame to the data base
- Single Bundle creation
- The relative transformation (including covariance) extraction from the bundle result
- PoseGraph building
- Loop closure detection and factor creation

3. Performance Analysis (חקר ביצועים): A detailed quantitative and qualitative analysis of the different stages of the system.
Where relevant, describe what you did to improve weak points, what was the change you implemented and what impact it had.

- Consider using log scale where appropriate (e.g., Covariance size graph, track length histogram)
- Where appropriate overly data on the same figure (e.g. for easy comparison)
- Make comparable graphs in the same scale

4. Discussion and Conclusions: Summarize the report and provide some criticism: identify weak points, unrealistic assumptions or aspects that could be improved. Suggest how you would improve the system and what are good directions for future research.
If you researched or implemented anything extra above what was required, describe what you did, the motivation and results.

- Limit the report to under 20 pages
- All pages and figures should be numbered
- The exact location of the code that created each figure should be specified next to it (file and line number)
- Every axis should have units labeled
- Make an effort to have the figures present the information clearly: Use a relevant scale, zoom in to the important part (e.g. drop the irrelevant part of the data), use colors, overlay effectively etc.

Performance Analysis

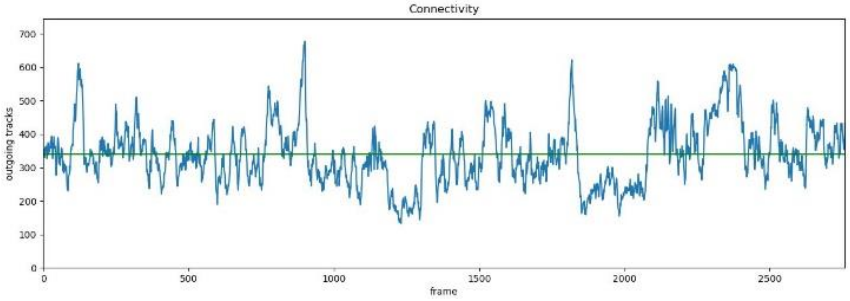
Present any graph, figure or statistic that demonstrates the performance of the different stages of the system as well as the system as a whole.
For each graph, provide an analysis of the information presented in the graph, highlight what the graph shows and explain what aspect (positive or negative) it demonstrates.
Where appropriate combine different data in the same graph. Crop the data such that the result is presented in a meaningful manner. **Each figure should be numbered.**
Present the following tracking statistics:

- Total number of tracks
- Number of frames
- Mean track length
- Mean number of frame links
- A graph of the number of matches per frame
- A graph of the percentage of inliers per frame

Add any graph / figure that you think presents the information and demonstrates the ideas you want to convey. A list of suggested graphs includes:

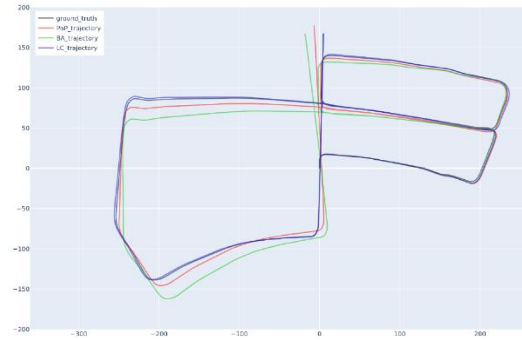
> **Note**: While including all the suggested graphs is not mandatory, it should include at least a substantial subset of them, and enough to convey the main points. Any additional graph that you think sheds light on some aspect or adds clarity is welcome

- Connectivity: For each frame, the number of tracks **outgoing** to the next frame (the number of tracks on the frame with links also in the next frame)

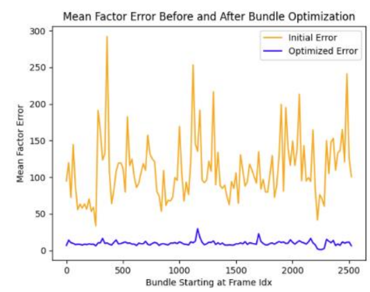


- Track length histogram

- The trajectory – bird eye view of PnP estimation, Bundle estimation and Pose Graph (with loop closure) estimation, including the ground truth for comparison.



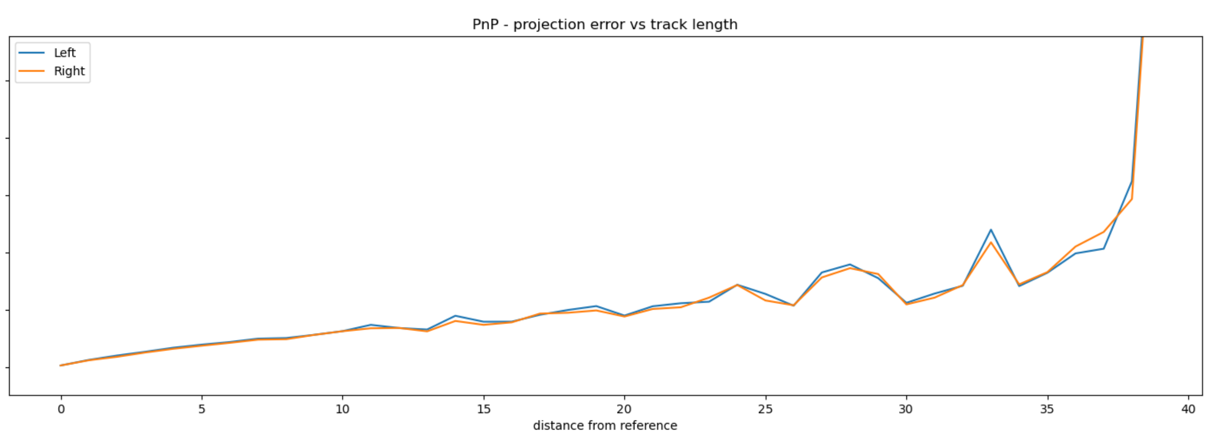
- Optimization error – Keyframes on the x axis and mean factor error on the y axis.
  The graph presents a line for the mean factor graph error (total error / #factors) of each bundle window (the window starting at that keyframe)
  One line for the error before optimization (initial error) and one for the error after optimization.



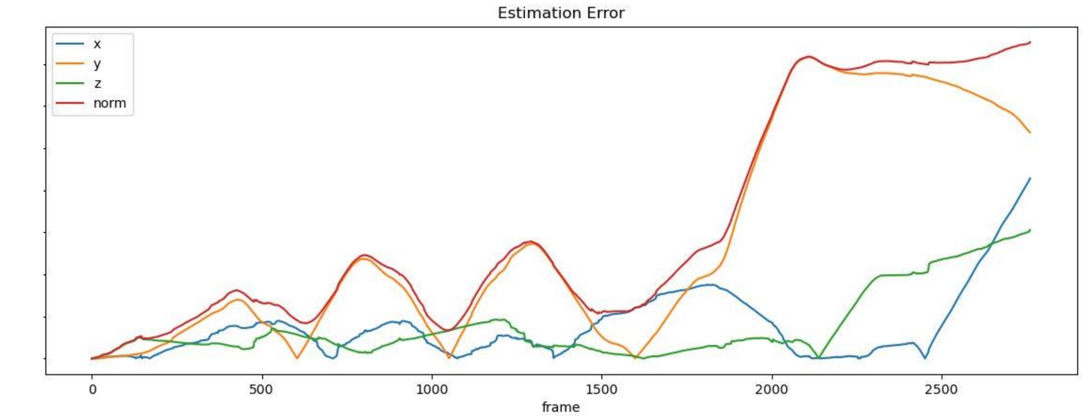
- Same as above, but for the median projection error of the bundle window (before and after optimization)

- Median (or any other meaningful statistic) projection error of the different track links as a function of distance from the reference frame (1st frame for Bundle, triangulation frame for PnP)
  - for PnP estimation
  - for Bundle estimation

  Note that for the required information can take a while to collect for all the tracks.
  A graph of a representative subset of the tracks can be a good approximation - Explain how you chose this subset.



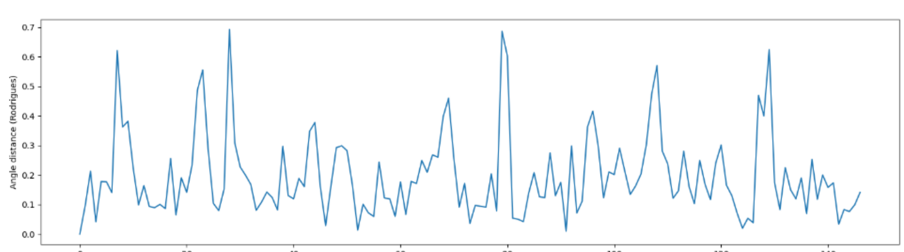
- Absolute PnP estimation error:
  - X axis error, Y axis error, Z axis error, Total location error norm (m)
  - Angle error (deg)



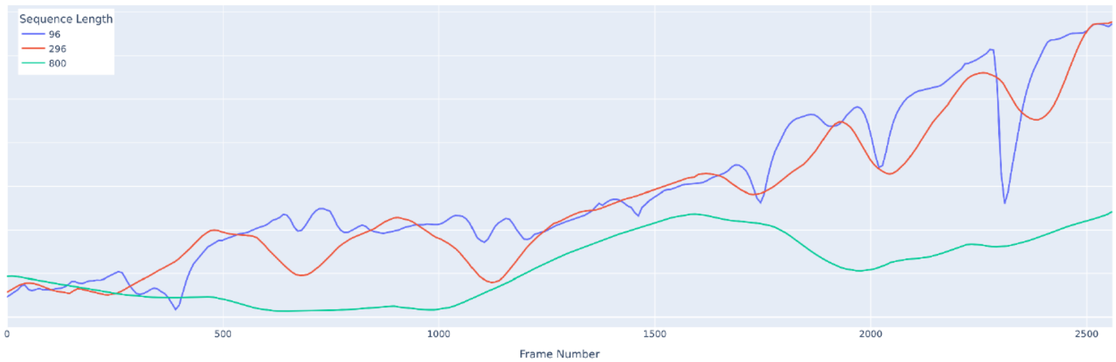
- Absolute Pose Graph (**without** loop closure) estimation error:
  - X axis error, Y axis error, Z axis error, Total location error norm (m)
  - Angle error (deg)

- Absolute Pose Graph (**with** loop closure) estimation error:
  - X axis error, Y axis error, Z axis error, Total location error norm (m)
  - Angle error (deg)

- Relative Error:
  The error of the relative pose estimation of every two consecutive keyframe compared to the ground truth relative pose between them, for both **Bundle** estimation and **PnP** estimation (for PnP this is the accumulation of the relative poses estimated between the two keyframes)
  See appendix
  - Total location error norm (m)
  - Angle error (deg)
  - Do you see any indication that leads you to suspect the accuracy of the ground truth poses?



- Relative **PnP** estimation error over sub-sections:
  The error of the relative pose estimation compared to the ground truth relative pose, evaluated on sequence lengths of (100, 400, 800).
  See appendix for definition
  - Total location error norm (measure as error%: $\frac{m}{m}$)
  - Angle error (measure as $\frac{deg}{m}$)
  - For each graph calculate the average error of all the sequences for total location norm and angle error (a single number for each)



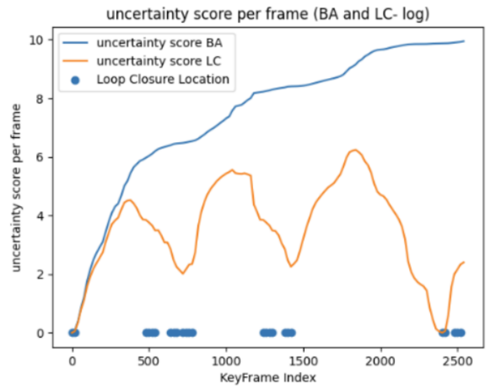
- Relative **Bundle** estimation error over sub-sections:
  The error of the relative pose estimation compared to the ground truth relative pose, evaluated on sequence lengths of (100, 400, 800). Choose closest keyframe if necessary
  See appendix for definition
  - Total location error norm (measure as error%: $\frac{m}{m}$)
  - Angle error (measure as $\frac{deg}{m}$)
  - For each graph calculate the average error of all the sequences for total location norm and angle error (a single number for each)

- Number of matches per successful loop closure frame
- Inlier percentage per successful loop closure frame

- Uncertainty size vs keyframe – pose graph **without** loop closure:
  - Location Uncertainty
  - Angle Uncertainty

- Uncertainty size vs keyframe – pose graph **with** loop closure:
    - Location Uncertainty
    - Angle Uncertainty

  How did you measure uncertainty size?
  How did you isolate the different parts of the uncertainty?



Appendix: Relative Error

Many stages in our process estimate the relative pose between two poses. A suitable measure for the accuracy should also be relative. To check the error of a relative pose estimate we compare the ground truth relative pose between two poses c_a, c_b to the estimated relative pose.

Comparing the rotation difference is best done with the Rodrigues formula that represents rotation as a 3D vector. The direction of the vector is the axis of rotation, and the size of the vector is the magnitude of the rotation in Radians.
For the size of a rotation represented by rotation matrix R (for example the relative rotation between c_a and c_b, $R = R_a^T R_b$) in degrees:

```
rvec, _ = cv2.Rodrigues(R)
numpy.linalg.norm(rvec) * 180 / numpy.pi
```

A basic relative estimation error graph compares the relative pose estimation between two consecutive frames $\Delta C(c_i, c_{i+1})$ and the ground truth relative pose between them $\Delta C(gt_i, gt_{i+1})$. (note that we may want to compare consecutive keyframes and not consecutive frames, so we'll use $\Delta C(c_i, c_{i+k})$ with k the number of frames between the keyframes)

The relative graphs over long sub-sections of the trajectory should give an idea of the expected error per meter traveled and is the way the KITTI benchmark grades the submitted solutions. For a given sequence of frames (e.g., segment 51→150 of length 100) we want to calculate the error of the estimated translation 51→150 compared to the true translation 51→150 between pose 51 and pose 150, divided by the trajectory length.

For the location we'll have

$$\frac{|error(51,150)|_{loc}}{total\_distance(51,150)}$$

where $|error(51,150)|_{loc}$ is the location difference between the estimated movement from frame 51 to frame 150 to the real (ground truth) movement, and $total\_distance(51,150)$ is the total ground truth length of the trajectory traveled from frame 51 to frame 150.
In the same manner, for angle we'll have:

$$\frac{|error(51,150)|_{ang}}{total\_distance(51,150)}$$

See definitions below.
Note that the denominator is **not** the location difference between gt_{51} and gt_{150}, but the sum of **all** the distances between all the frames between gt_{51} and gt_{150}.
The denominator should be the same for both location and angle errors, while the numerator changes ($distance$ or $angle$).
We measure this error-per-meter for different lengths of sequences – e.g., for sequences of length 100 we measure on sequence 0→99 , 1→100 , 2→101 , … , $(n-99)$→n.

Definitions:
est_a is the estimation pose of frame a, gt_a the ground truth pose of frame a.
$\Delta C(a,b)$ the relative pose between pose a and pose b (pose b in the coordinate system of pose a)

$$est\_displacement(a,b) \doteq \Delta C(est_a, est_b)$$
$$gt\_displacement(a,b) \doteq \Delta C(gt_a, gt_b)$$
$$error(a,b) \doteq \Delta C(est\_displacement(a,b), gt\_displacement(a,b))$$
$$|pose|_{loc} \doteq size\ of\ the\ location\ part\ of\ the\ pose\ (in\ meters)$$
$$|pose|_{ang} \doteq size\ of\ the\ rotation\ part\ of\ the\ pose\ (in\ degrees)$$

(For rotation size use the Rodrigues representation)

$$total\_distance(a,b) \doteq \sum_{i=a}^{b-1} |\Delta C(gt_i, gt_{i+1})|_{loc}$$