

# Mini-project: A visual odometry pipeline!

Juan Bermeo<sup>1</sup> and Tom Paye<sup>2</sup>

<sup>1</sup>juandiego.bermeoortiz@uzh.ch, LEGI: 20-754-951

<sup>2</sup>Address of second author

## MILESTONES

### 1. Initialization Module

- (a) Define/Specify the logic or argument to pick specific initialization frames for each dataset.
- (b) Identify features in both frames with one of the following techniques:
  - Corner detection: Harris, Shi-Tomasi, FastCorner
  - Sift Features
  - **Bonus:** Lift
- (c) Match the keypoints between frames to find keypoint correspondences between different views (between frames or between multiple cameras).
  - Choose a type of distance/similarity metric
  - Choose to do matching with either pairwise comparison or KLT
- (d) Estimate the point cloud of the landmarks.
- (e) Estimate relative pose of the next frame by calculating the fundamental matrix.
  - Use 8 point ransac (estimateFundamentalMatrix)
  - **Bonus:** 5 point ransac
  - **Bonus:** 1 point ransac + 5 point ransac
- (f) Refine the estimated pose with Bundle Correction.

### 2. Continuous VO Module

- (a) Create functions that iterates over all subsequent states, always using the keypoints in frame  $i$  and compare/match them to those of frame  $i + 1$  to estimate the pose of frame  $i + 1$ . (practically following the same pipeline as initialization).
- (b) Persist the the estimated poses at each step.
- (c) Choose error metrics and create function that calculates global and local error in pose estimation:
  - Figure out how to obtain the ground truth of poses at the local and global level
  - Create function that computes the metric for both
- (d) **Bonus:** Use some method to correct global drift.
- (e) **Visualizing results:**
- (f) Create interface that runs the pipeline
  - Display current image, plot's current and past features and the distance between them (plots matched features).
  - Plot trajectory and landmarks of the last 20 frames.
  - Plot # of tracked landmarks in the last 20 frames
  - Plots full trajectory
  - **Bonus:** Plots trajectory over satellite map
- (g) As the script runs, stitch the resulting matched features into a video (screencast)

## **IMPLEMENTATION: OVERVIEW**

How we expanded or deviated from outlined pipeline

List of things delivered

## **IMPLEMENTATION: SUCCINCT RECOUNT OF PROGRESS**

Setting of milestones

Achieving milestones

Interesting problems encountered and their solution

## **IMPLEMENTATION: RESULTS**

Qualitative evaluation of screencasts

### **Qualitative evaluation of our VO system**

1. Interesting things that went right
2. Interesting things that went right
3. What might explain or be the solution to the things that went wrong
4. Behaviour of additional things we implemented Note that having implemented an additional feature which degrades the quality of your VO pipeline will be accepted and valued as long as i) the implemented feature is properly motivated and described, and ii) an analysis showing the effect of your additional feature is provided in the report. If the additional feature degrades the quality, provide both a screencast with additional feature enabled and additional feature disabled.