Probabilistic Robotics Course

Projects

Giorgio Grisetti

grisetti@dis.uniroma1.it

Dept. of Computer Control and Management Engineering
Sapienza University of Rome

Rules

- Projects are individual
- The project has to be linked to a *private* git repo shared with the “staff”.

- **No admission at the exam without having choosen a project**

- For each project we provide:
  - ground truth solution
  - data of the problem

  Evaluation
  - The evaluation of a project consists in running the system on different data (but similar to the one used for testing)
  - Project accounts for ¼ of the final grade.
  - Special situations that require immediate registration of the exam (e.g. erasmus) will undergo a practical test (live) with the teachers.
  - **Projects should be hosted on a git repo. The history should be visible and meaningful and it will be taken into account for the evaluation**

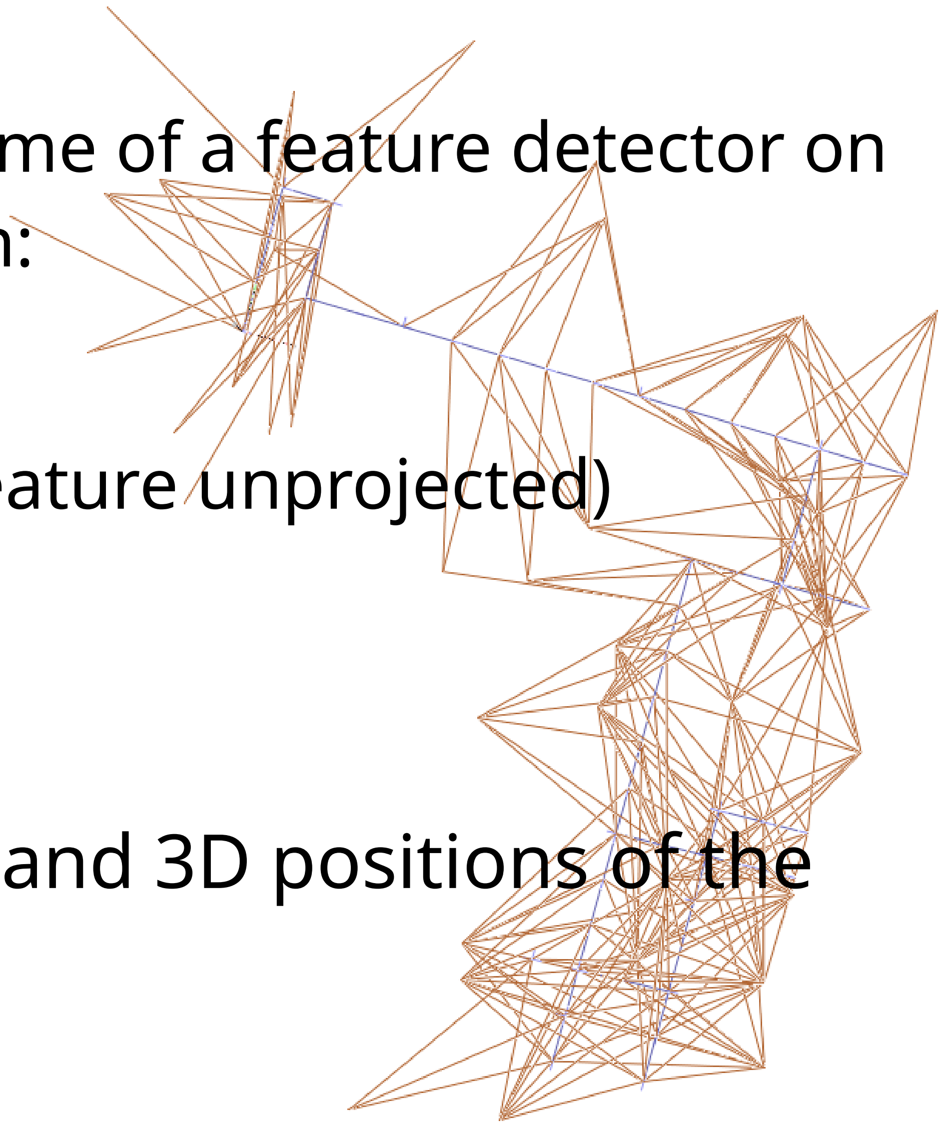
#1 Camera SFM (in 2 parts)

Input

- Files describing the outcome of a feature detector on an image. The files contain:
  - Camera id
  - Direction vector (img feature unprojected)
  - Image feature Id

Output

- Position of each camera and 3D positions of the triangulated points



#1a Camera SFM

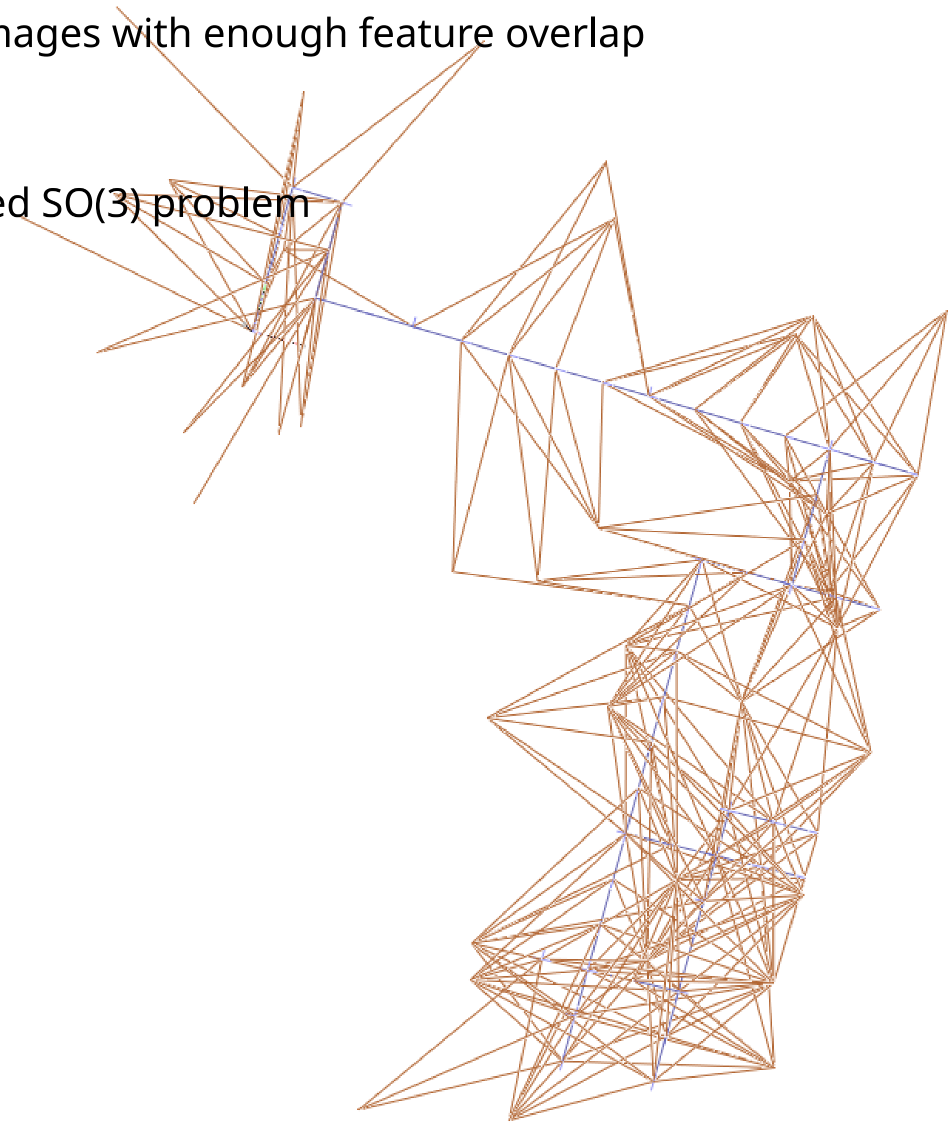
Initialization and rotation synchronization

How

- Extract Essential Matrices between pairs of images with enough feature overlap
- Identify the inliers in each image pair
- Compute the global rotations solving a relaxed SO(3) problem

Output

- For each matching pair:
  - Id cam 1
  - Id cam 2
  - Index of matching point in 1^{st} image
  - Index of matching point in 2^{nd} image
  - R, and t (up to a scale)
- For each image (but the 1^{st} which is at origin)
  - Absolute Rotation
- Error values (rotation)



#1b Camera SFM

Solve the global translation problem by blocking the global orientations and seeking for the translations that (up to a scale) satisfy the pairwise epipolar constraints

How

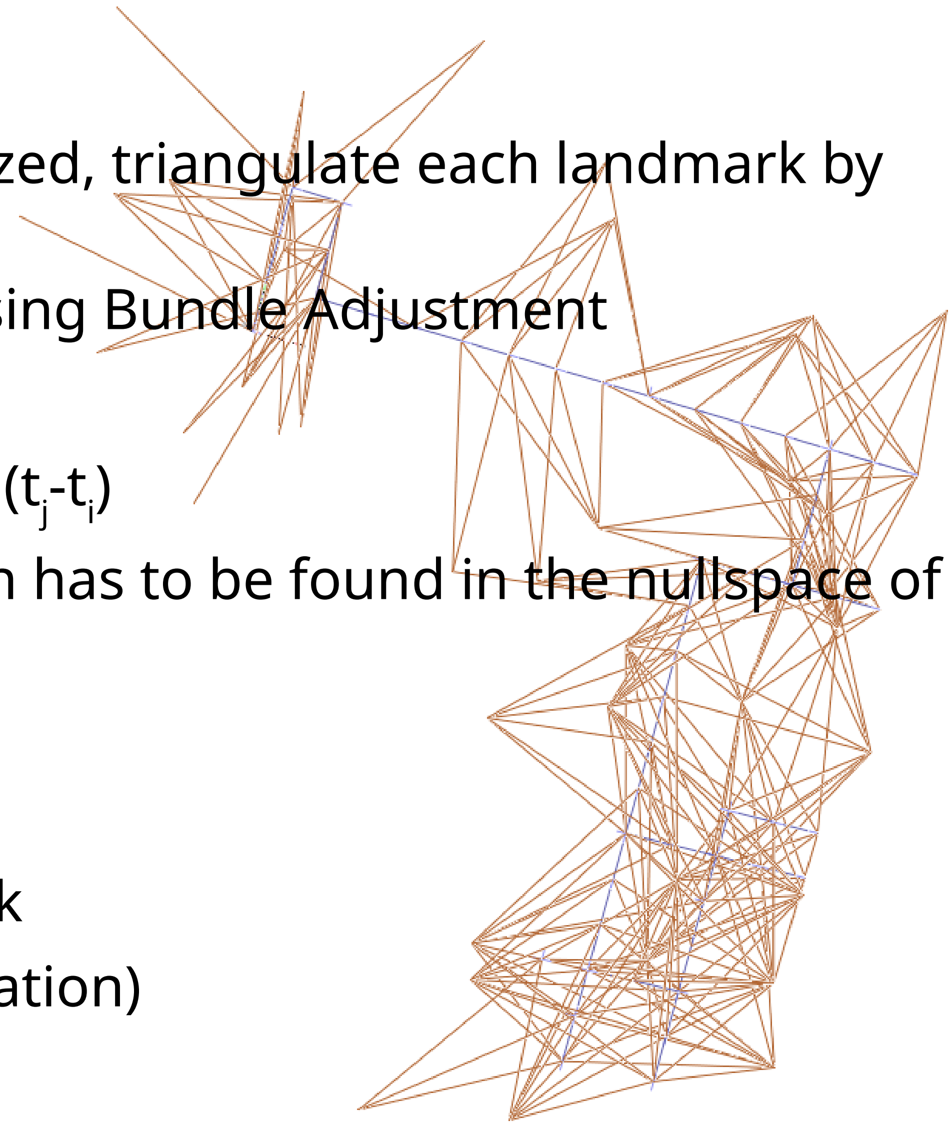
- Once the translations are initialized, triangulate each landmark by keeping the camera poses fixed
- Perform a final refinement by using Bundle Adjustment

Hints

- The epipolar constraint is $t_{ij} \times R^T_i (t_j - t_i)$
- In the error function the solution has to be found in the nullspace of H

Output

- Global position of each camera
- Global position of each landmark
- Error values (rotation and translation)



#2 Visual Odometry

Input

- Camera parameters
- Image sequence where each image is described by a sequence of keypoint-id pairs <u,v,id>

Output

- Trajectory (estimate vs gt)
- 3D points (estimate vs gt)
- Error values (rotation and translation)

How

1. Register the first pair using epipolar geometry
2. Triangulate the initial points and impose a scale
3. Incrementally add a new – triangulate and track
    a) Determine the position using Projective ICP
    b) Triangulate the missing points
    c) Update the camera pose

#3 Planar Monocular SLAM

Differential Drive equipped with a monocular camera

Input

- Integrated dead reckoning (wheeled odometry)
- Stream of point projections with “id”
- Camera Parameters
  - Extrinsics (pose of camera on robot)
  - Intrinsics (K)

Output

- Trajectory (estimate vs gt)
- 3D points (estimate vs gt)
- Error values (rotation and translation)

How

- Bootstrap the system by triangulating the initial set of points with the odometry guess
- Bundle Adjustment (total least squares) at the end

#4 Calibration of a (real) Robot

Calibrate both the kinematic parameters and the sensor positions of a front-rear tricycle-like robot

Input

- A file containing the encoder ticks of all encoders in the system:
  - absolute on the steer axis
  - incremental on the steering wheel
- The positions of the sensor w.r.t. an external tracking system

Output

- 2D position of the sensor w.r.t the mobile platform
- The kinematic Parameters:
  - Ksteer, Ktraction, SteerOffset, Baseline

#5 Multi-PICP localization

Determine the position of a multi (3) camera system observing a set of known 3D point landmarks.

Input

- Position of the landmark on each of the images at each frame(unknown associations)
- Global pose of the landmarks

Output

- Trajectory (estimate vs gt)

- **How**
  - Bootstrap the system by running RANSAC on the set of landmarks seen in the 1st frames
  - Track the position by running a variant of PICP that runs on all cameras.

For all projects...

Read carefully the README file inside your folder data. It contains all details to successfully complete and evaluate your project.

For any inquiries or doubts please send us an email.

How to get a project

Send an email asking for a project (name and number) to

- salem@diag.uniroma1.it
- derebotti@diag.uniroma1.it

1) Use as Subject: **[ProbRob][ProjAss]**
2) Write something that identifies you (at least your Student ID)
3) Wait for instructions