Bundle Adjustment

The aim of this project is to find camera calibrations and 3d point position describing a scene given observations of the 3d points in each camera.

We have N cameras and M points. We have the following information:

- Rough initial estimates of each cameras calibration.
- Rough initial estimates of each points 3d position.
- 2D observations of points in each camera. That is: for each camera a list of `(point_id, point_projected_into_camera)` tuples.

The objective is to find camera calibration (extrinsics and intrinsics for all N cameras) and 3d point positions such that, for every observation, the 3d point projects to the correct 2d point.

With correct calibrations and projected points should match exactly with the observations (the dateset has been generated synthetically).

We have provided you with three files:

1. `projection-check.json`: This file contains the calibration and observations of one camera and can be used to make sure that your camera projection model is correct. These 3D points are not the same as in the following files.

2. `noisy-calibrations-and-world-points.json`: This file contains the actual calibrations and 3D points that need adjusting. The 2D observations are accurate but we have added noise to the 3D world points and the calibrations (rotation, translation and focal length). A correct solution should yield very low reprojection errors.
3. `noisy-calibrations-and-accurate-world-points.json`: This file has accurate 2D to 3D correspondences with an initial noisy calibration. To be used only with the alternative route described below.

Bundle Adjustment - Google Drive

https://drive.google.com/drive/folders/1u6EWAxTFzxmkcdZkK…

File Structure

```
{ "cameras": { "{cameraId}": { "calib": { "extrinsics": [], // 3x4 row
major OpenCV style extrinsics matrix. "intrinsics": [], // 3x3 row
major OpenCV style intrinsics matrix. "size": [] // [width, height] of
image. }, "observations": { "{pointId}": [], // [x, y] in image co-
ordinates. ... } }, ... }, "worldPoints": { "{pointId}": [], // [x, y,
z] in world co-ordinates. ... } }
```

Camera Model

The camera model is an opencv style pinhole projected camera. There are no distortion coefficients. The aspect ratio is 1 that is: `(focalLengthX = focalLengthY)`. As mentioned above the provided intrinsic values are correct in `noisy-calibrations-and-world-points.json` except for focal length.

Submission

1. Your optimised calibrations and world points in the same file format as `noisy-calibrations-and-world-points.json` in a file called `answer.json`.
2. Your source code.
3. A brief note about what you have done! Please let us know if you have used AI to help your submission.

Extensions/Alternatives

We assign this project to a range of candidates. There are adjustments to make the task easier or harder. Use your judgement to pick a route that works with the time/skills you have.

Extensions

1. Don't use the 3D world points! You can get initial guesses using triangulation with the noisy calibrations.
2. Test your solution with noise/outliers in the 2d points.
3. Some sort of visualisation.
4. Solve the problem using neither the initial calibrations or initial world points (i.e. only use the `observations` field from the json).

Alternative route

An alternative task which might be easier to start on is to: refine approximate calibrations of the cameras cameras to the fixed ground truth 3d points (a straightforward calibration refinement with 3d-2d point correspondences) using `noisy-calibration-with-accurate-points.json`.

You could then move onto bundle adjustment if you have time.

General Guidance

We will assess your submission factoring in how much third party library code you have used. For two submissions implementing the same features less library code for the core part of the problem would be preferred. For many candidates using an existing non-linear least squares library would be appropriate. For ancillary tasks tasks like parsing json/loading images/test frameworks we suggest you do use libraries.

Including some notes/documentation about your solution and the approach you took would be useful. To help candidates with limited spare time we attempt to take into account the time spent on the assignment when assessing it - some notes of how you approach the problem and where time was spent will help us make a good judgement here.

Notes

- `[0, 0]` is the top left of the image and `[width, height]` is the bottom right of the image.
- Feel free to use existing frameworks/tooling but you will get bonus points for doing things from scratch.