Project Report

Visual Odometry Pipeline

Marius Grimm, Amadeus Oertel, Toni Rosiñol

Titus Cieslewski, Henri Rebecq
Adviser

Prof. Dr. Davide Scaramuzza
Professor

Robotics and Perception Group
University of Zurich & Swiss Federal Institute of Technology Zurich (ETH)

2017-01

Universität Zürich[UZH]

ROBOTICS & PERCEPTION GROUP

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Contents

List of Figures

Abstract

very short and juicy summary of topic and main points

Figure 1: 2D-2D correspondences with KLT.

1 Introduction

2 Initialisation

2.1 Monocular Initialisation

The monocular initialisation is a key module in the Visual-Odometry pipeline. It is ordered in the follwoing way:

1. Find the 2D-2D correspondences between the the chosen first two images
2. Apply the 8-point Algorithm to estimate the Fundamental matrix combined with running RANSAC.
3. Check the validity of the estimated Fundamental matrix by either comparing the reprojection error or the distance to the epipolar line.
4. RANSAC return a set of inlier of the current 2D-2D correspondences.
5. With the new set of inlier we can re-evaluate the final Essential Matrix E estimate.
6. The Essential Matrix E can then be decomposed into two rotation and two translation hypotheses for the pose of the first camera frame. This gives in total a set of four camera motion possibilities.
7. These hypotheses have to be disambiguated to choose the right camera rotation and translation.

We have implemented two different approaches for finding the 2D-2D correspondences:

1. Exercise implementation of Harris descriptor and detector.
2. Implementation of the Kanade-Lucas Tracker (KLT).

An examplary output of the 2D-2D module can be seen in Figure **??**. The correspondences between two images are indicated.

Due to efficiency reasons we decided to use the epipolar line distance as a measure to validate the estimated Fundamental Matrices. In theory the reprojection error would have yielded better results (sacrificing efficiency); however, in our case the differences were neglectable.

3 Continuous Operation

3.1 KLT

4 Bonus Features

4.1 Plotting

4.2 Automatic Selection of Frames for Initialisation

4.3 Relocalisation

4.4 Full Bundle Adjustment

4.5 Calibrated Smartphone Camera and Own Dataset

In order to assess the robustness of our VO pipeline, we decided to record our own dataset. For this endeavour, we printed a checkerboard to calibrate the camera of an smartphone. We then used the calibrated camera to record the dataset. In order to achieve good results for the calibration, we made sure that we followed the following procedure:

1) Set focus and exposure mode of the smartphone to fixed. 2) Ensure that the checkerboard is over a planar surface and presents no light reflections. 3) Take frames of the checkerboard from sufficiently different angles and distances.

Once the calibration dataset recorded, we proceeded to find the intrinsics of the camera. As a first approach we used the Matlab toolbox (REFERENCE TO THE TOOLBOX). Unfortunately, we could not get good calibration results. This was in part due to the fact that the interface requires you to click on the corners of the checkerboard in the image, and therefore limits the amount of images you can process in a given time. We also used OpenCV to retrieve the calibration matrix of the camera. Nonetheless, we found that the most user-friendly calibration tool was the one offered as a Matlab app named Camera Calibrator.

Using this last tool, we managed to retrieve approximately the same intrinsic parameters independently of the calibration images. Once calibrated, we processed the images to get a gray-scaled, resized and undistorted set of images. Then we fed the new dataset to our VO pipeline and checked the results with our ground truth. The ground truth was taken approximately with no special instrumentation other than a meter.

We encountered many problems while recording the dataset, to name a few: motion blur when moving, changes of illumination on the scene, maintaining a constant focus, transferring images, having enough features on the scene, etc.

In the end, we found that monocular initialisation was having difficulties to output a sufficient amount of 2D-3D correspondences. This led the continuous operation part to stop after two frames on average. The reprojection errors on the triangulated keypoints were of the order of magnitude of tens of pixels, depending on parameters and bootstrap_frames.

In the end, we learned about different tools available for camera calibration and about the difficulties of recording a satisfiable dataset for visual odometry. Or seen from another perspective, we learned about the limits of our VO pipeline.

Our calibration files and images of the dataset are provided under the folder smartphone.

4.6 Quantitative Analysis

4.6.1 Keypoint Tracking via Block Matching versus KLT

4.7 Monocular Initialisation Harris Detector via Matlab versus Exercise Harris Detector

5 Conclusion

Main conclusions and final remarks.

References

[1] J.-L. Blanco, F.-A. Moreno, and J. González-Jiménez. The málaga urban dataset: High-rate stereo and lidars in a realistic urban scenario. *International Journal of Robotics Research*, 33(2):207–214, 2014.

[2] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[3] D. Scaramuzza and F. Fraundorfer. Visual odometry: Part i - the first 30 years and fundamentals. *IEEE Robotics and Automation Magazine*, 18(4), 2011.

[4] D. Scaramuzza and F. Fraundorfer. Visual odometry: Part ii - matching, robustness, optimization, and applications. *IEEE Robotics and Automation Magazine*, 19(2), 2012.

A Appendix

Additional material such as long mathematical derivations.

B Examples

This appendix provides some additional hints and examples for the layout and style of the thesis. It is worthwhile to look at the source file `Examples.tex` for this appendix to understand how it was created.

B.1 Tables

Tables are left justified and the caption appears on top as seen in Table 1.

Table 1: Translations.

English	German
cell phone	Handy
Diet Coke	Coca Cola light

B.2 Figures

Figure 2 shows a simple figure with a single picture and Figure 3 shows a more complex figure containing subfigures.

Figure 2: IRIS logo.

(a) ETH logo (b) IRIS logo

Figure 3: Two pictures as part of a single figure through the magic of the subfigure package.

B.3 Units

The SIUnits package provides nice spacing for units as demonstrated in Table 2. Use of the package also makes it easy to change the style or even the unit text in the future.

Table 2: Spacing for units.

Output	**Command**
42m	`42m`
42 m	`\unit{42}{\metre}`
42 m	`42 m`

B.4 Miscellany

Capitalization. When referring to a named table (such as in the previous section), the word *table* is capitalized. The same is true for figures, chapters and sections.

Naming of structural elements. Refer to a `\section` in LaTeX as a chapter and call a `\subsection` section. (I don't like the way `\chapter`s are rendered in the report document class. Hence the suboptimal markup/naming correspondence.)

Bibliography. Use `bibtex` to make your life easier and to produce consistently formatted entries.

Contractions. Avoid contractions. For instance, use "do not" rather than "don't."

Captions. A brief version of a caption can be provided for the list of figures and tables as demonstrated with the caption of Figure 3. The mechanism can also be used to get rid of the final period of a caption in the lists.