Vision Algorithms for Mobile Robotics (Mini-) Project

Yvain de Viragh
Marc Ochsner
Nico van Duijn

06.01.2016

1 Introduction

In the process of this course, we implemented several building blocks for a visual odometry pipeline during the exercises. As a mini-project, our task was to put the pieces together and implement a full pipeline. In doing so, we were using some of the work from the exercises as well as some MATLAB functions from the Computer Vision Toolbox. Our final pipeline was tested on three different datasets and showed good performance up to a scale factor.

2 Pipeline Structure

2.1 Initialization

2.2 Lukas-Kanade Tracker

2.3 Structure-Only Bundle Adjustment (Bonus feature)

In order to increase the accuracy of the pose estimation and to combat scale drift, we run structure-only bundle adjustment on all landmarks which have already been observed a certain number of times. While this should not be as effective as full bundle adjustment, it is computationally significantly cheaper.

Our approach is based on linear triangulation, but instead of only considering two point correspondences takes in account multiple ones. I.e., given n point correspondences we have the following system of equations:

$$\underbrace{\begin{bmatrix} \vec{p}_1^{\times} \boldsymbol{M}_1 \\ \vdots \\ \vec{p}_n^{\times} \boldsymbol{M}_n \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} {}_W\vec{r}_{WP} \\ 1 \end{bmatrix}}_{\vec{x}} = 0_{6\times 1} \quad \Rightarrow \quad \boldsymbol{A}\vec{x} = 0$$

where $\boldsymbol{M}_i$ denotes the projection matrix of frame/image i. This linear-least squares problem is solved same as in the case for 2 point correspondences. Crucial for this approach to work correctly is to not include any outliers. We therefore only take into account those observations where the reprojection error given the unadjusted landmark is sufficiently small.

3 Parameters and Tuning

4 Results and Discussion