Malte Schwarzkopf, St John's College, ms705@cam.ac.uk

Proteus – Project Progress Report – Week 13

Summary

Initial fundamental matrix computation code in place, feature copying implemented, node moving supported by the GUI, OpenGL-based model viewer widget added.

What I intended to do

- Finish the last bits of the edge snapping implementation
- Get started on code to compute the fundamental matrix and perform the first steps of the structure-from-motion process
- If time permits, try to experimentally integrate the KLT tracker for proper feature tracking between frames (carried over from before)
- Read up on more theory behind the fundamental matrix and camera matrix computations
- Feature copying and linear interpolation between frames are still missing

What I ended up doing

- Finished off edge snapping as planned; leaving further improvements for later
- Implemented the fundamental matrix computation using the TooN library, but realised it doesn't support access to components of the SVD after computing it.
- Interfaced the code with QuantLib, re-implemented the fundamental matrix computation using the QuantLib API
- Identified and fixed a bug with the fundamental matrix computation if there are 8 or less annotated features in a frame
- Implemented a facility to copy annotations between frames (using copy constructors and the overloaded assignment operator)
- Added a 3D model viewer based on the Qt OpenGL viewport widget, supporting rotation around all axes; initial reading about generating OpenGL content
- Celebrated Christmas :-)

Where I am in the timetable

As before, I am somewhat behind on the time table – worryingly, the core features (the structure-from-motion process) seem to go a lot more slowly than anticipated. Some delays happened due to the Christmas holidays, but I attempted to not take too much time away from project work. The change of linear algebra library also contributed an unexpected delay (see below). The current state is that the (non-normalised) fundamental matrix computation is working except for a bug that leads to the signs on the first and second column of the resulting F matrix being swapped.

Problems/Issues encountered

The initially used linear algebra library, TooN, turned out to not support access to component matrices of the SVD, hence it had to be replaced by QuantLib (which, unfortunately, is rather huge, and makes the memory footprint of the application significantly worse). I spent quite a significant amount of time comparing a model implementation in Matlab to my implementation – part of the reason why this took so long was the fact that I had to learn to understand the Matlab syntax before I could make head and tail of the code.

What I intend to do

- Continue implementation of the fundamental matrix computation, investigate the sign error on the first and second columns
- Add normalisation for fundamental matrix
- Investigate how to implement camera matrix computation
- If time permits, try to experimentally integrate the KLT tracker for proper feature tracking between frames (carried over from before)
- Read the chapter on camera matrix computation and point extraction (triangulation) in Hartley & Zisserman

Further remarks

None.