OptFlow:
Matlab-Tools for the computation of optic flow

Claudius Strub, Jens Peter Lindemann

September 15, 2015

1 Overview

This is a short introduction on how to use the OF-toolbox. This toolbox computes the optic flow an agent (e.g. an animal or a robot) experiences when moving in a given environment on a given trajectory. Here the environment is defined by a set of 3D polygons and the trajectory describes a time series of positions and view orientations of the agent. The toolbox computes image velocities resulting from these definitions at sampling points on a spherical visual sensor.

The toolbox functions are grouped by prefixes into the following groups:

- OFCalc... Computation of optic flow data.
- OFMod... Creating models.
- OFDraw... Visualisation of data.
- OFGenerate... Generate sample points, or stats.
- OFSubroutine... No need to call these ones. They are used by other Methods.

1.1 You need

You need three data structures to start

1. a list of 3D-coordinates (vertices) and yaw, pitch, roll angels which describe your Trajectory (tra). This is an $Tx6$ matrix. Angles are in radian! example: tra=[0 0 0 0 0 0; 0.5 0.5 0 0.15 0 0.2];
2. a list of spherical coordinates which describe your SamplePoints (sp). This is generated from OFGenerateSp() (see help for parameter details).
3. a model which describes your world. There are a couple of methods starting with OFMod... which can be used to create one.

1.2 Creating a model

1.2.1 example

world = OFModCylinder('object', 2, 4, 10);
world = OFModTranslate(world, 4, 2, 0);
world = OFModJoin(world, OFModCube('EventHorizon', 10, 10, 10));

1.2.2 infos

The 3D center of all models is in [0 0 0].
When an model is labelled with 'EventHorizon', it is drawn transparent (initially used for a surrounding box). All other labelled models will be drawn half-transparent.

1.2.3 optional: Visualisation

- Models can be plotted with OFDrawPatch().
  example: OFDrawPatch(world, true);
- Or a model and trajectory can be plotted together with OFDrawTra().
  example: OFDrawTra(world, tra, 1);

1.3 compute OpticFlow

1.3.1 example

[hof, vof]=OFCalcOpticFlow(world, tra, sp);

1.3.2 infos

To compute the optic flow use OFCalcOpticFlow() To get good results, trajectory steps should be as small as possible. (Optic flow is computed with the Koenderink equation [1]) Rotations of more than $pi/4$ between two trajectory steps will lead to inaccurate results. Rotations of more than $pi/2$ will lead to wrong results!

1.3.3 optional: Visualisation

To get the optic flow at trajectory step 2:

- on an 3d sphere: OFDraw3dPlot().
  example: OFDraw3dPlot(tra, sp, hof, vof, 2);
- in an cylindrical coordinate system: OFDrawCylPlot().
  example: OFDraw3dPlot(sp, hof, vof, 2);

In the cylindric coordinate system the optic flow is shifted, so that the sample point [0 0] is in the middle (pi) now. So $pi/2$ is left and $3 * pi/2$ is right. 0 and $2 * pi$ are backwards.

References

[1] J. J. Koenderink and Andra J. van Doorn: Facts on Optic Flow. *Biological Cybernetics 56(4) pp. 247–254*, 1987