Visual Odometry Using Commodity Optical Flow

Jason Campbell[†‡], Rahul Sukthankar[†‡], Illah Nourbakhsh[‡]

[†] Intel Research Pittsburgh
417 South Craig Street, Suite 300
Pittsburgh, PA 15213
jason.d.campbell@intel.com, rahul.sukthankar@intel.com

[‡] Carnegie Mellon University, The Robotics Institute
5000 Forbes Ave
Pittsburgh, PA 15213
jasoncam@ri.cmu.edu, rahuls@ri.cmu.edu, illah@ri.cmu.edu

Abstract

A wide variety of techniques for visual navigation using robot-mounted cameras have been described over the past several decades, yet adoption of optical flow navigation techniques has been slow. This demo illustrates what visual navigation has to offer: robust hazard detection (including precipices and obstacles), high-accuracy open-loop odometry, and stable closed-loop motion control implemented via an optical flow based visual odometry system. This work is based on 1) open source vision code, 2) common computing hardware, and 3) inexpensive, consumer-quality cameras, and as such should be accessible to many robot builders.

Demo Overview

Optical flow field and camera ego-motion estimation have been the subject of much research for over 30 years, but this research has seen limited use. For many years this could be attributed to the high computational cost of the known techniques, but modern PCs and embedded systems have been sufficiently powerful to enable real-time optical flow analyses for several years now. Other potential reasons why optical navigation has not seen wider application may include the mathematical and coding complexity of implementing a robust vision system, and a lack of understanding about the high quality of information available via the technique.

This demonstration is designed to explore (and give participants a chance to explore) the practicality and potentially high quality of an optical flow navigation system based on 1) readily available open source vision code, 2) common computing hardware, and 3) consumer-quality cameras. The capabilities of this visual navigation system include robust hazard detection (precipices and obstacles), high-accuracy open-loop "visual odometry", and stable closed-loop motion control. As part of the demonstration, participants will arrange a variety of hazards (obstructions, precipices) for a tabletop mobile robot equipped with a USB webcam as its only sensor. The robot will show robust detection of any hazards put in its way and, in the absence of a hazard, demonstrate robotic positioning accuracy to better than 1% of distances traveled and better than 5 degrees in orientation, in spite of any (accidental or experimenter-induced) wheel slip. Participants may also vary parameters in the vision system (in particular, frame rate and resolution) and observe the effect on overall system performance in order to gain a better appreciation of the design-time tradeoffs required by different environments, velocities, and types of movement.

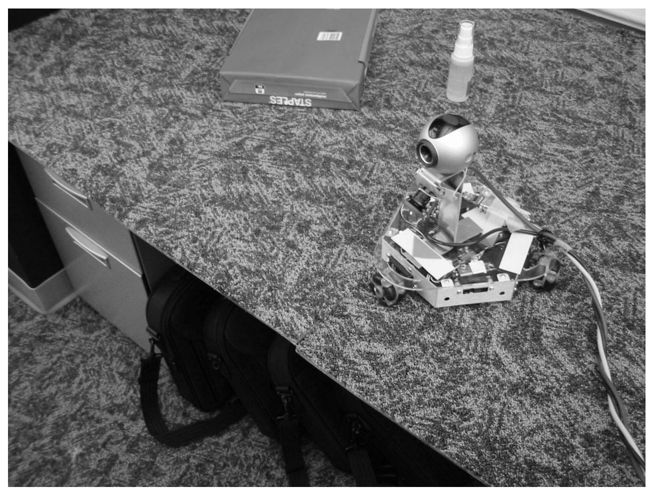

Figure 1: Robot poised near a drop-off to illustrate precipice detection. The visual navigation system demonstrated can reliably detect a cliff even when similar colors and textures appear at both the bottom and the top of the precipice (such as in this photo). The robot pictured has two cameras, but only the lower one, pointing roughly 30° below horizontal, is in use.

Related Work

This is by no means the first robot to use visual sensing to control motion! As long ago as 1976, Hans Moravec and Donald Gennery used feature tracking algorithms to perform visual servoing/course correction on the Stanford Cart (Gennery and Moravec 1976). Since then many research robots have explored more sophisticated forms of visual sensing. One recent example has been the CMU

Autonomous Helicopter Project, which uses an on-board visual navigation system based on optical flow techniques (Amidi 1996).

Early work on visual navigation focused on the extraction of an optical flow field from a time-sequence of images. For a survey of such algorithms, see (Barron et al. 1994) which remains the definitive comparison study in the area. This study highlighted the gradient-based image matching technique proposed in (Lucas and Kanade 1981) as effective across both synthetic and real-world image sequences, a conclusion other researchers have reached less formally as they have chosen to base further work on this algorithm. Recently, an efficient form of the Lucas-Kanade technique (Bouguet 1999) has become widely available as part of the OpenCV computer vision library[1] (Bradski 2000). OpenCV also incorporates image acquisition and processing functions suitable for commonly available cameras and so offers a relatively complete package for those wishing to apply the Lucas-Kanade technique. This demo system uses the Lucas-Kanade functions in OpenCV to capture video frames and extract the optical flow field from each pair of frames.

Once an optical flow field has been obtained for a sequence of images, a further analysis is required to estimate camera ego-motion corresponding to the flow field. From a research perspective this problem typically has been viewed as just one component of a more general problem termed “structure-from-motion” or SFM. Ego-motion estimation and SFM are challenging because they seek to recover 3-D information from a 2-D projection, and to do so an algorithm must treat multiple observations of movement in the 2-D images as one or more sets of simultaneous equations. The resulting sets of equations are often both highly over-determined and subject to ill conditioned inputs. Statistical methods such as least-median-of-squares or RANSAC have been proposed to help screen out outliers and segment flow fields (Bab-Hadiashar and Suter 1996). An additional complication arises in that some solution techniques may fail when presented with flow fields corresponding to the common case of 2-D rotational and translational movement along a flat floor, or when the tracked points all lie in a single plane (Torr et al. 1998). The visual odometry system demonstrated adopts several simpler techniques (calculating median feature displacements and evaluating the consistency of feature vectors over time) which allow quick but surprisingly accurate estimates to be made of incremental and total distance traveled / angle turned / floor geometry by assuming the robot is traveling in only two dimensions over a predominantly flat floor.

[1] http://www.intel.com/research/mrl/research/opencv/

References

Amidi, O. 1996. An Autonomous Vision-Guided Helicopter. Ph.D. Thesis, Electrical & Computer Engineering Department, Carnegie Mellon University.

Barron, J. L., Fleet, D. J., and Beauchemin, S. S. 1994. Performance of Optical Flow Techniques. The International Journal of Computer Vision 12(1):43–77.

Bab-Hadiashar A., Suter D. 1996. Robust Optic Flow Estimation Using Least Median of Squares. In Proceedings of the IEEE International Conference on Image Processing, 513–516.

Bouguet, Jean-Yves 1999. Pyramidal Implementation of the Lucas-Kanade Feature Tracker, OpenCV Documentation, Microprocessor Research Labs, Intel Corporation.

Bradski, G. 2000. Programmer's tool chest: The OpenCV library. *Dr. Dobbs Journal*, November 2000.

Gennery, D. and Moravec, H. 1976. Cart Progress Report, Stanford University (unpublished). http://cart.frc.ri.cmu.edu/users/hpm/project.archive/robot.papers/1976/nasa1.txt

Lucas, B.D., and Kanade, T. 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. In Proceedings of 7th International Joint Conference on Artificial Intelligence, 674–679.

Torr, P.H.S., Zisserman, A. and Maybank, S. 1995. Robust Detection of Degenerate Configurations for the Fundamental Matrix. In Proceedings of the Fifth International Conference on Computer Vision, 1037–1042.