Accurate and Computationally-inexpensive Recovery of Ego-Motion using Optical Flow and Range Flow with Extended Temporal Support

Graeme A. Jones
dircweb.kingston.ac.uk/graeme/

Digital Imaging Research Centre
School of Computing and Information Systems, Kingston University,
Penrhyn Road, Kingston upon Thames, UK, KT1 2EE

Introduction

Recovering the ego-motion of a moving camera within a static scene supports many applications in robotics and computer vision. The presented work is motivated by *pre-vis* applications in the film industry; specifically the ability to render digital assets into the scene during production in real-time. A low-cost commodity depth camera can be easily mounted on and calibrated to a high quality production cameras and used to extract changes in sensor pose from the induced motion of the rigid scene. This work explores the effectiveness of the computationally efficient *range flow* technique to generate this real time pose information directly from the depth stream of a Kinect sensor.

A number of challenges within the approach are addressed. First, an iterative version of the *small rotations* motion estimator is developed to ensure the most accurate inter-frame estimates. Second, the substantial issue of *drift* is addressed - the accumulated error between true and estimated sensor pose as motion estimates are temporally integrated. *Anchor frames* which enjoy significant overlap with subsequent frames are stored and used to provide additional temporal range flow constraint within the estimation process. Where there are loops in the data sequence, it is advantageous to select anchors from previously seen parts of the scene. Finally, in some scene configurations, there is insufficient constraint from the depth images. We exploit the availability of registered intensity images to further constrain the sensor motion using the *optical flow* framework.

Analogous to optical flow, *range flow* is a per-pixel constraint on the 3D displacement of an imaged 3D point given its local spatio-temporal depth derivatives. These must be combined across a region or an image to provide sufficient constraint to extract 3D motion[1, 2].

The Optical Flow and Range Flow Constraints

The *constant brightness equation* relates how the luminance at a 3D scene point is captured in temporally separated intensity images. A 3D point $\mathbf{X}$ is imaged at pixel position $\mathbf{x} = (x, y)^T$ in the intensity map I_t. This point undergoes a 3D motion $\Delta\mathbf{X} = (\Delta X, \Delta Y, \Delta Z)^T$ which results in an image motion $\Delta\mathbf{x}$ between frames t and τ and reprojects with the same intensity at the new image location $\mathbf{x} + \Delta\mathbf{x}$. Thus the *optical flow* constraint is defined as

$$I_\tau(\mathbf{x} + \Delta\mathbf{x}) = I_t(\mathbf{x}) \tag{1}$$

Analogous to the *constant brightness equation*, the following range constraint relates how a 3D point is captured in temporally separated depth images. A 3D point $\mathbf{X}$ (measured in the depth camera's coordinate system) is captured at pixel position $\mathbf{x} = (x, y)^T$ in the depth map Z_t. This point undergoes a 3D motion $\Delta\mathbf{X}$ which results, first, in an image motion $\Delta\mathbf{x}$ between frames t and τ, and second, in a change of the depth ΔZ of the 3D point captured at this new image location $\mathbf{x} + \Delta\mathbf{x}$. Thus the *range flow* constraint is formulated as

$$Z_\tau(\mathbf{x} + \Delta\mathbf{x}) = Z_t(\mathbf{x}) + \Delta Z \tag{2}$$

Motion Model

Scene displacement is induced by the movement of the camera whose rotation ω and translation $\mathbf{t}$ motion parameters can be modelled using the useful linear *small rotation* formulation when the movement between consecutive frames is relatively small. Here the 3D displacement $\Delta\mathbf{X} = (\Delta X, \Delta Y, \Delta Z)^T$ of a 3D scene point $\mathbf{X} = (X, Y, Z)^T$ is given by

$$\Delta\mathbf{X} = \omega \times \mathbf{X} + \mathbf{t} = \begin{bmatrix} Z\omega_y - Y\omega_z + t_x \\ X\omega_z - Z\omega_x + t_y \\ Y\omega_x - X\omega_y + t_z \end{bmatrix} = M(\mathbf{X})\mathbf{a} \tag{3}$$

where $\mathbf{a}$ is a concatenation of the motion parameters $\mathbf{a} = (\omega, \mathbf{t})^T$.

Minimising drift using Anchor Frames

Simply integrating between-frame motion estimates over time will inevitably result in *drift i.e.* the accumulated error between true and estimated sensor pose. To minimise this, additional temporal constraint can be included. Specifically we introduce the concept of an *anchor frame*. In addition to recovering a parameter update for the motion between the current frame and the previous frame, the same update is also constrained by the motion between the current frame and its anchor. A depth frame is an anchor to all subsequent frames with which it retains a significant degree of overlap. Once the amount of overlap falls below a threshold, the last frame is promoted as the next anchor. Updates to the motion are now computed from two sources of range flow constraint.

When there are loops in the sequence, it would be advantageous to select anchors from previously seen data rather than using the last frame. Such constraint from early frames makes a significant impact on the degree of drift. To this end, a list of all anchor frames is maintained. When a new anchor is required, this list is searched for the earliest anchor overlapping the current frame. However, a consequence of this approach, is the linear growth in storage requirements for these anchors and in the computational cost of searching through these anchors as the length of the video sequence grows.

Evaluation

The recently published TUM RGB-D Benchmark [3] is used to evaluate the motion estimator. This resource provides Kinect depth and registered RGB sequences with synchronized ground truth of the sensor pose for extensive set of sequences of varying lengths and differing levels of challenge such as large visual velocities or large dominant planar surfaces. Our study uses nine Freiburg1 (FR1) sequences in which the Kinect sensor is moved within a typical indoor environment. The resource also provides an evaluation tool that computes the root mean square error (RSME) between an estimated trajectory and the associated ground truth once these have been aligned. Specifically we use the translation and rotation RMSE measures, T-RMSE and R-RMSE respectively, and add the maximum value of the *absolute trajectory error* (MATE) which identifies the maximum sensor positional error anywhere along the estimated trajectory.

Conclusion

Pose accuracy can reasonably be judged as commensurate with SLAM approaches (as represented by the RGB-D SLAM system) but available at a fraction of the computational cost. Indeed a real-time implementation is running on an old Samsung P460 Notebook. Given the low computational cost of generating reasonably accurate pose estimates, the presented approach could usefully bootstrap more computationally expensive techniques.

[1] John L. Barron and Hagen Spies. “The Fusion of Image and Range Flow”). In *Proceedings of the 10th International Workshop on Theoretical Foundations of Computer Vision: Multi-Image Analysis*, pages 171–189, London, UK, UK, 2001.

[2] M. Harville, A. Rahimi, T. Darrell, G. Gordon, and J. Woodfill. “3D Pose Tracking with Linear Depth and Brightness Constraints”. In *International Conference on Computer Vision*, pages 206–213, 1999.

[3] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. “A Benchmark for the Evaluation of RGB-D SLAM Systems". In *Int. Conf. on Intelligent Robot Systems*, October 2012.