Incremental Light Bundle Adjustment

Vadim Indelman
indelman@cc.gatech.edu
Richard Roberts
richard.roberts@gatech.edu
Chris Beall
cbeall3@gatech.edu
Frank Dellaert
dellaert@cc.gatech.edu

College of Computing,
Georgia Institute of Technology,
Atlanta, GA 30332, USA

Fast and reliable bundle adjustment is essential in many applications such as mobile vision, augmented reality, and robotics. Two recent ideas to reduce the associated computational cost are structure-less SFM (structure from motion) [1, 5, 6, 7] and incremental smoothing [3, 4]. The former formulates the cost function in terms of multi-view constraints instead of re-projection errors, thereby eliminating the 3D structure from the optimization. The latter was developed in the SLAM (simultaneous localization and mapping) community and allows one to perform efficient incremental optimization, adaptively identifying the variables that need to be recomputed at each step.

In this paper we combine these two key ideas into a computationally efficient bundle adjustment method, and additionally introduce the use of three-view constraints to remedy commonly encountered degenerate camera motions.

The optimized cost function in light bundle adjustment (LBA) is defined, similarly to [5, 6], as

$$J_{LBA}(\hat{x}, p) \doteq \sum_{i=1}^{N_h} \|h_i(\hat{x}, p)\|^2_{\Sigma_i} \tag{1}$$

with $\hat{x}$ the estimated poses for all cameras, $\hat{p}$ all image observations across all views, Σ the measurement covariance, and where $\|a\|_\Sigma \doteq a^T\Sigma^{-1}a$ denotes the squared Mahalanobis distance. The parameter N_h represents the number of multi-view constraints h_i derived from the feature correspondences in the given sequence of views. Each constraint h_i is a function of several camera poses and the image observations in the corresponding images. The applied multi-view constraints are a combination of two- and three-view constraints [2], that, as opposed to using only two-view constraints, allow consistent motion estimation in a straight-line camera motion.

We formulate the optimization problem in terms of a factor graph, and incrementally update a directed junction tree which keeps track of the current best solution [3, 4]. The factor graph defines a factorization of the function $f(x)$ as

$$f(x) = \prod_\alpha f_\alpha(\mathcal{X}_\alpha), \tag{2}$$

where $\mathcal{X}_\alpha \subset x$ is the set of all camera poses x_j connected by an edge to factor f_α. Each factor f_α represents a single multi-view constraint between the appropriate views. A simple example of a factor graph using two- and three-view constraints is shown in Figure 1a.

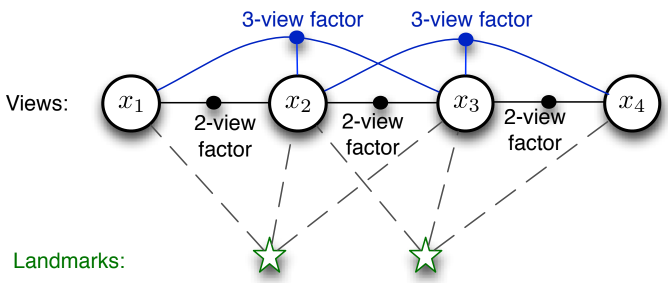
The optimization process corresponds to adjusting all the camera poses x to obtain a maximum a posteriori estimate

$$\hat{x} = \arg\max_x f(x) = \arg\min_x (-\log f(x)). \tag{3}$$

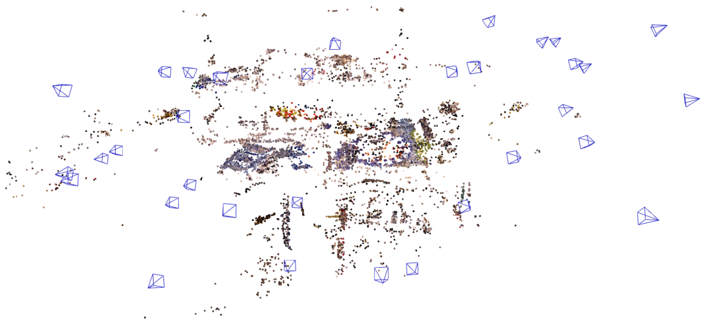
Assuming a Gaussian distribution, the above formulation is equivalent to a non-linear least-squares optimization of the cost function (1). Typically, only a small fraction of the camera poses are recalculated in each optimization step, leading to a significant computational gain. Although only the camera poses are optimized in LBA, if desired, all or some of the observed 3D points can be reconstructed after the optimization convergence.

We present a performance evaluation of incremental LBA (iLBA), i.e. applying incremental smoothing for optimizing the cost function J_{LBA}, using several datasets. Figure 1b shows the optimized camera poses and the reconstructed structure for one of the datasets. The structure reconstruction was performed based on the LBA-optimized camera poses.

Comparing iLBA to previous structure-less BA methods [1, 5, 6, 7] and to conventional bundle adjustment reveals significantly better timing performance and similar accuracy levels.



(a)



(b)

Figure 1: (a) A factor graph representation for a simple example of 4 cameras observing 2 landmarks. Two-view and three-view factors are added instead of projection factors. Landmark observations are denoted by dashed lines. (b) Optimized camera poses and reconstructed structure in the *cubicle* dataset.

[1] V. Indelman. Bundle adjustment without iterative structure estimation and its application to navigation. In *IEEE/ION Position Location and Navigation System (PLANS) Conference*, April 2012.

[2] V. Indelman, P. Gurfil, E. Rivlin, and H. Rotstein. Real-time vision-aided localization and navigation based on three-view geometry. *IEEE Trans. Aerosp. Electron. Syst.*, 48(3):2239–2259, July 2012.

[3] M. Kaess, V. Ila, R. Roberts, and F. Dellaert. The Bayes tree: An algorithmic foundation for probabilistic robot mapping. In *Intl. Workshop on the Algorithmic Foundations of Robotics*, Dec 2010.

[4] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Intl. J. of Robotics Research*, 31:217–236, Feb 2012.

[5] A. L. Rodríguez, P. E. López de Teruel, and A. Ruiz. Reduced epipolar cost for accelerated incremental sfm. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3097–3104, June 2011.

[6] A. L. Rodríguez, P. E. López de Teruel, and A. Ruiz. GEA optimization for live structureless motion estimation. In *First Intl. Workshop on Live Dense Reconstruction from Moving Cameras*, pages 715–718, 2011.

[7] R. Steffen, J.-M. Frahm, and W. Förstner. Relative bundle adjustment based on trifocal constraints. In *ECCV Workshop on Reconstruction and Modeling of Large-Scale 3D Virtual Environments*, 2010.