Resource-Aware Large-Scale Cooperative 3D Mapping from Multiple Cell Phones

Chao X. Guo, Ryan C. DuToit, Kourosh Sartipi, Georgios Georgiou, Ruipeng Li, John O’Leary, Esha D. Nerurkar, Joel A. Hesch and Stergios I. Roumeliotis

Introduction: A robust and tractable solution to the large-scale 3D mapping problem has numerous useful applications such as human (or robot) indoor navigation, augmented reality, and search and rescue. Visual and inertial measurements have been used to create 3D maps in GPS-denied environments, with emphasis on how to efficiently process a single dataset. In many practical applications, however, a mobile device (cell phone or wearable computer) may not have sufficient resources to record and process a dataset collected over a building-size area. In this work, we address the problem of cooperative mapping (CM) using datasets collected by multiple users at different times, with unknown relative starting poses. Specifically, our proposed algorithm seeks to find an efficient batch least-squares (BLS) solution that can utilize all the available visual and inertial measurements to achieve the best accuracy, and judiciously selects common landmark measurements (landmarks observed by multiple users) so as to gain processing speed. In particular, the main contributions of this work are:

• Instead of processing all the datasets together as in BLS, we formulate CM as a equivalent constrained optimization problem, where the cost function is the summation of cost terms from each dataset, and the constraints relate landmarks observed by multiple users. This formulation allows trading estimation accuracy for computational cost by selecting a subset of commonly-observed landmarks.

• The proposed algorithm is modular (i.e., maps or submaps can be added or removed), lends itself to parallel implementation, and is able to leverage each individual user’s intermediate mapping results to reduce the processing cost.

• We provide a robust method for determining an initial estimate for the relative transformation between all users.

Algorithm Overview: The objective of this work is to find a BLS solution over all users’ trajectories and maps. Our algorithm can be divided into three main steps:

1) Obtain a BLS solution for each individual user’s trajectory and map independently, using measurements from only their dataset.

2) Generate an initial estimate of the users’ relative poses, using their visual measurements to common landmarks.

3) Find the optimal BLS solution of all users’ trajectories and maps utilizing all available sensor data, and either a subset, or the entire set, of constraints that arise from commonly-observed landmarks.

Experimental Results: To validate our algorithm, we collected four datasets in a 2,000 m$^2$ building. These datasets comprised gray-scale images and inertial measurements collected with a mobile device, spanning a total distance of approximately 2 km. Harris and KLT were used to generate consecutive image feature correspondences, while inter/intra-dataset loop closures were found using feature descriptors (ORB, FREAK, or SIFT) and a vocabulary tree (VT).

Fig. 1 shows all four trajectories. To further validate our approach, we compared the estimated dimensions of the building with those in the blueprints, and found our estimation error to be approximately 0.5%. Finally, by dropping constraints imposed by a subset of the commonly-observed landmarks, we were able to increase speed by a factor of three, while introducing only 13 cm of RMS pose error. See http://onionmaps.info for an interactive visualization.

We leverage the CM results for online re-localization. Specifically, a mobile application queries the VT used in the CM with the current image, to determine feature matches with the mapped images. Once the feature correspondences are found, and given their 3D positions from the CM, we employ PnP to localize the user within a building, allowing for a Google Maps similar interface for indoor navigation. Given sufficient feature matches and geometry, we are able to localize the user within 0.5 m for over 95% of queries.