Sensor fusion for flexible human-portable building-scale mapping

Sensor Fusion for Flexible Human-Portable Building-Scale Mapping

Maurice F. Fallon, Hordur Johannsson, Jonathan Brookshire, Seth Teller and John J. Leonard

Abstract— **This paper describes a system enabling rapid multi-floor indoor map building using a body-worn sensor system fusing information from RGB-D cameras, LIDAR, inertial, and barometric sensors. Our work is motivated by rapid response missions by emergency personnel, in which the capability for one or more people to rapidly map a complex indoor environment is essential for public safety. Human-portable mapping raises a number of challenges not encountered in typical robotic mapping applications including complex 6-DOF motion and the traversal of challenging trajectories including stairs or elevators. Our system achieves robust performance in these situations by exploiting state-of-the-art techniques for robust pose graph optimization and loop closure detection. It achieves real-time performance in indoor environments of moderate scale. Experimental results are demonstrated for human-portable mapping of several floors of a university building, demonstrating the system's ability to handle motion up and down stairs and to organize initially disconnected sets of submaps in a complex environment.**

I. Introduction

In this paper, we outline the systems and methodologies used to develop an interactive, multi-user, body-worn mapping system. The goal is to enable natural exploration and provide situational awareness of a multi-floor building, where GPS access is denied. The approach is motivated by the need for emergency responders, security and military personal to quickly develop knowledge about their environments.

Our prototype mapping system fuses visual, LIDAR, IMU and barometric sensor data from each user to generate robust maps of the environment. Although similar to a typical robotic Simultaneous Localization and Mapping (SLAM) task, the multi-user, body-worn system provides several unique challenges:

- The path from any single user may not be continuous (e.g., the user may travel up stairs or elevators).
- Sensors may not maintain a fixed pose relative to the plane of travel (e.g., LIDAR returns may not be parallel to the ground).
- Wheel odometry, typically used to estimate incremental translation, will not be available. All egomotion must, thus, be estimated from the other sensors.

Beyond these challenges, our mapping system faces similar issues associated with any multi-agent SLAM such as: efficiently combining map representations from different sources, the gradual increase in computation due to a growing SLAM problem, and communication constraints which limit the distribution of maps between users. We specifically address the issue where multiple users will explore regions which partially overlap.

In this work, disjoint maps from a single user (as floor levels are changed and revisited) and from multiple users combine to form a set of partially overlapping maps. We use visual information to propose similar locations and to infer overlapping regions. Using these inter-map constraints, maps can be aggregated in a single combined map.

In the following section, we review the background of personal localization and situational awareness systems and discuss our design prototype. Section III describes the hardware and, in Section IV, we give an overview of the algorithmic components of the mapping system. Given the challenging application we wish to support, we outline modules for estimating and correcting sensor tilt, floor traversals and user detection in Section V.

II. Background

Outdoor localization using GPS and Wi-Fi networks has become ubiquitous and is provided by many modern mobile phones. However, localization of individuals in indoor, unknown environments is more challenging. Our motivating example is a Biohazard Site Assessment team. In order to support such a scenario, we require a system which interacts minimally with the user and assumes no known infrastructure.

The approach developed by [1] envisages the rapid deployment of a RF sensor rig (containing 12-16 elements) surrounding a building of interest which is coupled with an user-mounted transmitter. The location of the user, typically a first responder, can then be triangulated to within approximately one meter and displayed graphically on a commander's console.

Foot-mounted IMU-based relative motion estimation methods have shown significant promise, including the work by [2]. This approach demonstrated remarkable performance (0.3% drift) and has been adapted to allow for continuous localization within a known, multi-floor map [3] using a particle filter.

The work described by [4] integrates step sensors, a barometer, a magnetometer and an IMU with a GPS sensor via EKF sensor fusion to estimate ego-motion with a very accurate global position estimate (assuming occasional GPS fixes).

FootSLAM [5], [6] utilizes similar input data and a Dynamic Bayesian Network to produce a high level topology of the path followed by a user within a SLAM framework. At this stage of our platform development, we have excluded the

The authors are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA mfallon,hordurj,jbrooksh,teller,jleonard@mit.edu

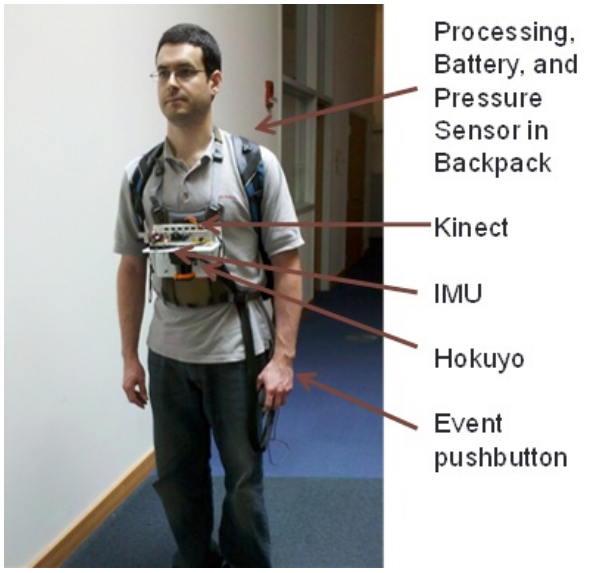


Fig. 1. A user carrying our prototype human-portable mapping system

foot-mounted IMU for experimental and practical reasons, although these compelling demonstrations motivate its usage.

Each of these systems estimate the relative motion of the user; however, in our envisaged applications, the layout and structure of the environment is of primary importance. Perhaps the platform most similar to ours is that developed by [7], [8]. This man-portable platform contains several LIDAR sensors as well as lateral cameras. The system uses three orthogonal LIDARs to estimate pitch and roll and uses LIDAR scan matching, aided by visual odometry, to build maps. While the visualizations produced by this approach are convincing, this large rig would be unappealing given our first responder application domain.

Additionally, [9] demonstrates a hand-portable system which carries out incremental 2-D scan matching using LIDAR, having estimated the pitch and roll of the unit using an IMU.

Finally [10] developed a light-weight cane-mounted LIDAR-IMU device which enables localization for the blind within a prior building model. This approach requires a pedometer and the limited sensing capability would restrict its utility.

III. Hardware

Our prototype system, shown in Figure 1, consists of a vest-mounted sensor suite and an electronics backpack. The sensor vest consists of a Microsoft Kinect RGB-D sensor, Microstrain 3DM-GX3-25 IMU, and Hokuyo UTM-30LX LIDAR. The electronics backpack includes a laptop, 12V battery (for the Kinect and Hokuyo) and a barometric pressure sensor. The rig is naturally constrained to be within about 10 degrees of horizontal at all times, which is important for successful LIDAR-based mapping (see Section V-B). An event pushbutton, currently a typical computer mouse, allows the user to 'tag' interesting locations on the map. Additionally the map can be transferred wirelessly to a handset for the user to visualize the current map *in situ*. Regardless of these interactions, the core map system is intended to be as passive as possible.

Given the dangerous nature of the end user scenario, modification of the HazMat suit is generally impossible. In fact, the suits are typically destroyed after use to avoid contamination. In the future, we envisage that the final device will be a hand-held unit, similar in size to a miner's lamp or discretely installed on the shoulder of the user.

IV. Core Mapping System Overview

As mentioned previously, we aim to produce a metric map for each separate floor with weak topological constraints representing vertical transitions between floors (namely staircases and elevators). It is important that this map be built on-line by the laptop worn by the user and shared with other members of the user's team.

SLAM has been a core research focus within the robotics community for many years. Developing from the modern probabilistic framework outlined in [11], researchers initially tackled the problem using the Extended Kalman Filter (EKF) to jointly estimate the robot's pose and environment map. Using this approach it was observed that the combined state vector of this system would grow linearly and the associated covariance matrix quadratically with time [12].

While particle filter approaches [13] have been considered to address complex data associations, recently smoothing approaches have become dominant [14]. These approaches recognize the sparse structure of the information form of the covariance matrix to develop methods for efficient and consistent optimization of the full trajectory and map.

Large scale, highly accurate LIDAR-based SLAM maps can now be generated in real-time — typically using wheel odometry. A core component of our SLAM system is accurate incremental scan-matching; we employ the multi-level resolution approach described by [15], [16]. Efficient pose-graph optimization, such as g2o [17] and iSAM [18], allow position error and uncertainty accumulated during exploration to be removed by closing loops. These components provide the core of our mapping algorithm.

Leveraging this basic SLAM expertise, we have made specific design decisions and algorithm adaptations to allow for flexible and robust body-worn mapping without placing restrictions on the motion of the user.

V. Multiple Floor and User Mapping

Within the typical SLAM paradigm, a single robot explores its environment and extends its map while remaining continuously localized. By actively planning its exploration, the robot can avoid becoming lost or disconnected from a previously explored location. In contrast, when the user tilts the sensors aggressively, travels in elevators, or traverses staircases, numerous disconnected and unorganized maps will be generated. For example, the experiment discussed in Section VI consists of 18 floor traversals over the course of 31 minutes and results in floor maps greater than 100m

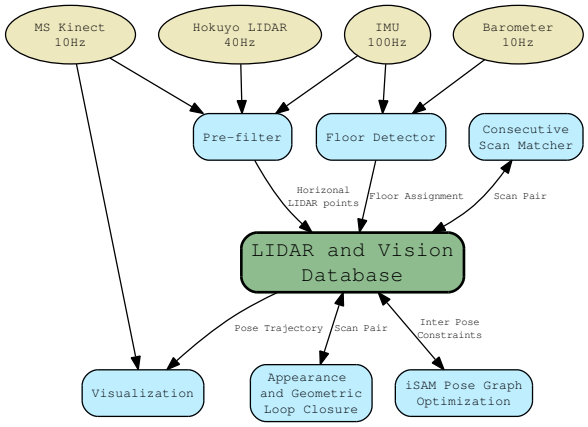


Fig. 2. The components of our software system. Lemon colored components represent raw streams of sensor measurements. Blue components represent the filters, detectors and classifiers which build the maps. These interact with a core database containing the filtered LIDAR and the RGB SURF features.

across. Our approach aims to piece these maps together to form a complete overview of the user's exploration.

A. LIDAR Odometry and Visual Loop Closure

The major software components of the system are illustrated in Figure 2.

As the user explores, consecutive scan-matching is used to infer relative motion. On a regular basis (2m or 30 degrees), a new node is added to a SLAM graph representing the combined incremental motion. In this way a single linear graphical model is formed with no conflicting constraints.

Additionally, we search for loop-closures using two back-end modules. The first module searches for matches between scans located geometrically close to one another (according the current SLAM graph) based on the following geometric conditions:

1) The scans are more than 30 seconds separated.
2) The scan footprints overlap by 50 percent.
3) The proportion of returns matching exceeds 70%.
4) The scans contain more than one *non-parallel wall exceeding 2m*.

These proposals are tested using the scan-matcher mentioned in Section IV. While the Hokuyo LIDAR sensor has a range of 30m and a field of view of 270°, our current prototype obscures the sides of the scan; as a result, we have found that long corridors within our test location lack sufficient features to maintain accurate scan-matching. For this reason, recurring failure of geometric condition (4) was also used to initiate new submaps in these locations.

As mentioned above, numerous disconnected but overlapping maps will be created via multiple staircase traversals and multiple users. So as to organize and reconnect the maps, we maintain a visual database of locations previously visited using the DBow library [19] with SURF features [20] extracted from the Kinect's RGB camera.

Each visual location corresponds to a pose from the SLAM graphical model (and hence a matching LIDAR pose). This visual bag-of-words application implements the hierarchical vocabulary tree described by [21] and an inverted file structure which allows for quick queries of matching pairs of images. So as to allow efficient combination of previously disconnected maps into a single combined map we utilize the concept of an 'Anchor Node' introduced by [22].

In this approach, an auxiliary anchor node is added to each individual pose graph during exploration. When an encounter or visual loop closure is detected, the corresponding LIDAR scans are scan-matched (initialized using zero pose offset) to produce an inter-submap constraint. This approach is successful because SURF features are reproducible only within a 50 degree viewing angle [20]. The constraint between the two pose graphs is then expressed through the two anchor nodes which allows the SLAM optimization engine to avoid re-parameterizing each submap into a single global coordinate frame at that instant. An example of a successful and failed loop-closure are demonstrated in Figure 3.

(a) Geometric proposal with 130° orientation difference - correctly accepted.

(b) Appearance proposal with a high scan-matching score - correctly rejected as only a single line feature is present.

Fig. 3. Two challenging loop-closure scenarios.

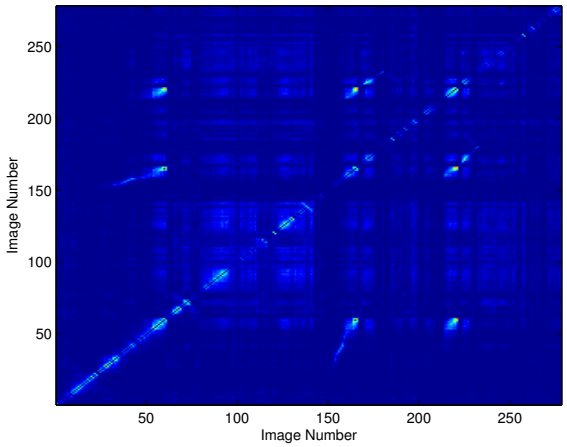


Fig. 4. The similarity matrix generated by exploring a particular location 3 times — as evidenced by the activity in off-diagonal elements of the matrix.

When submaps have been successfully merged, the geometric loop-closure algorithm mentioned above is repeated to further refine the map. While this step is not as critical as the visual appearance-driven loop closings, given the complex topologies described in the experimental section, it is useful to densify connections between otherwise loosely connected submaps.

Our approach intentionally does not tightly integrate visual information with the LIDAR and, as a result, the intrinsic calibration of the the camera and extrinsic calibration of the various sensors are not required and can be time varying. This approach was taken so that the design prototype could be flexible and deformable.

B. Man-portable Considerations

In the following sections we will describe a series of algorithms to detect unusable data, floor transitions and user interactions. These algorithms acted as a pre-filter between the raw incoming data and the SLAM system.

1) Tilt-correction: In a typical robotics application, the LIDAR sensor is mounted entirely horizontal on a robot moving horizontally. However on the pedestrian's vest, the sensor is subject to pitch and roll due to the user's gait which can quickly corrupt the map.

While one could consider the dependence on a single approximately horizontal LIDAR as being a weak point of this system, our experiments have shown that a user with a natural walking gait collects laser scans suitable for processing.

To correct for pitch and roll, we use an estimate of the direction of gravity from the IMU to estimate the pitch and roll of the sensor using the manufacturer-supplied compensation filter. The LIDAR returns are projected on to a horizontal plane before being inserted into the map. Additionally, by using an estimate of the height of the sensor, floor penetrating returns can also be discarded entirely. This height was typically known in advance but could easily be learned from the RGB-D sensor data. Beyond a pitch of 10 degree limits, the maximum range of the LIDAR before ground penetration is about 5 meters (at a typical waist height) which is too short for stable scan-matching.

C. Detecting Staircase Transitions

Detection of floor transitions is a primary capability required to enable free user motion within a large building. Although we considered detecting the vertical component of incremental visual odometry (aligned using the IMU's attitude estimate), this approach was not used here. Stairwells are often unlit or featureless, and users tend to turn sharply on intermediate landings causing frequent motion blur in the Kinect RGB images.

To detect the floor transitions, we instead used a basic barometric pressure sensor (Gulf Coast Data Concepts P/N B1100-1). Figure 5 shows the raw results from the pressure monitor (blue) as the user traveled up and down six flights of stairs. We found that the readings were consistent over several hours, but varied over several days.

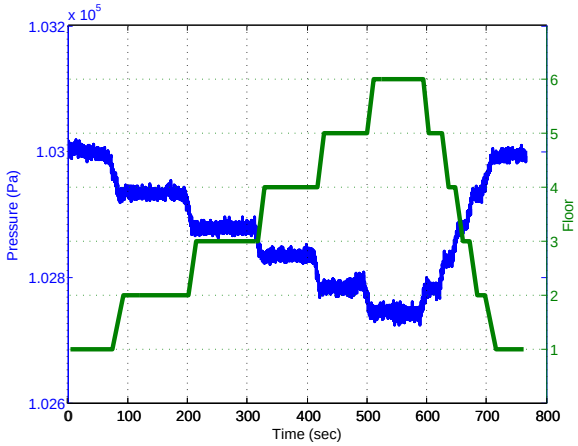


Fig. 5. The raw pressure readings (blue) are processed using a Gaussian mixture model to detect floors (green).

In order to accurately detect floors, we use a Gaussian mixture model as presented in Algorithm 1. The raw pressure reading, p_{new} is filtered by `LPF` to remove high frequency noise. Since the change in floors is relatively slow, we use a low pass filter with cutoff at 0.1Hz. Incoming samples are then accumulated into a 10 second buffer, p_{hist}, via `Append`. When the variance exceeds a threshold, $thresh1$, we consider the user to be in a floor-transition state (i.e., going up/down the stairs).

When a floor-transition ends, `BestFloor` determines whether the new readings are similar to a floor we have already visited. Since each Gaussian corresponds to a floor, the most similar floor is determined by finding the minimum Mahalanobis distance between the mean of p_{hist} and each Gaussian in the mixture, GM. If $thresh2$ is exceeded, a new Gaussian model, GM_{new} is created for the new floor via `NewGM`. When walking on the floor steadily, `UpdateGM` updates the mean and covariance associated with the current floor in GM.

The algorithm is executed once per pressure reading. If m pressure samples are maintained for each Gaussian in the mixture, the algorithm's complexity is $O(m)$, dominated by the calculation of the mean and covariance of the Gaussian.

The results of the algorithm on a sample data set are shown in Figure 5. Notice that the algorithm correctly determines floor transitions on the way up the stairs (times 0-550) and then correctly identifies the previously visited floor on the way down the stairs (times 550-800).

D. Detecting Elevator Transitions

We also explored the detection of elevator transitions. While the barometer could again be used for this purpose, when a user is stationary in an elevator there is a recognizable and consistent acceleration profile observed in the Z-component of the Microstrain IMU's accelerometer as illustrated in Figure 6. By integrating this measurement we can more accurately quantify the magnitude of the vertical transitions as the user is typically stationary for the duration.

Algorithm 1: EstimateFloor $(p_{new}, floor, GM, p_{hist})$

inputs : p_{new} raw pressure
$floor$ current floor estimate
GM Gaussian mixtures
p_{hist} 10 second buffer of pressure data
outputs: $floor$ floor estimate

$p_{filt} \leftarrow \texttt{LPF}\,(p_{new}, 0.1Hz)$
$p_{hist} \leftarrow \texttt{Append}\,(p_{hist}, p_{filt})$
$\sigma_p \leftarrow \texttt{cov}\,(p_{hist})$
$onStairs \leftarrow \sigma_p > thresh1$
if $state == ON\_FLOOR \wedge onStairs$ **then**
  // walking into stairwell
  $state \leftarrow ON\_STAIRS$
  $floor \leftarrow \emptyset$
if $state == ON\_FLOOR \wedge \neg onStairs$ **then**
  // on same floor, update the GM
  $\texttt{UpdateGM}\,(p_{filt}, GM[floor])$
if $state == ON\_STAIRS \wedge onStairs$ **then**
  // still in stairwell
  $state \leftarrow ON\_STAIRS$
if $state == ON\_STAIRS \wedge \neg onStairs$ **then**
  // walking out of stairwell
  $state \leftarrow ON\_FLOOR$
  $[floor_{new}, logProb] \leftarrow \texttt{BestFloor}\,(p_{hist}, GM)$
  if $logProb < thresh2$ **then**
    $[floor_{new}, GM_{new}] \leftarrow \texttt{NewGM}\,(p_{hist})$
    $GM \leftarrow \{GM, GM_{new}\}$
  else
    $\texttt{UpdateGM}\,(p_{filt}, GM[floor])$
  $floor \leftarrow floor_{new}$

Using a method similar to that described in Section V-C a unique classification of the number and relative height of specific floors is possible.

In future work, we aim to use this height information to create full 3-D constraints between different floor maps and to correctly align the maps with one another.

E. Integration with SLAM system

For the elevator transitions, we biased our thresholds towards over-estimating the number of floor transitions so that none were missed — even if this occasionally proposed a spurious submap. Additionally, floor transition detectors were typically used only to indicate the transitions but not to infer the specific floor. For the staircase transitions, for example, the algorithm does not necessarily provide floor numbers corresponding to the physical ordering of floors. Rather, it provides a unique number for each floor; this is what is needed to associate maps as the user travels the building.

For our practical experiments, we maintained a single instance of SLAM for the duration of operation — with each floor map retained in memory. While this raised the possibility of incorrect loop closure between floors, say between Floor 2 and Floor 3, we did not experience this issue in practice. Nonetheless, robust loop-closure and introspection is a major outstanding issue, we discuss this in Section VII.

VI. Experimental Results

As a demonstrative example of performance, we carried out an extensive multi-floor traversal in which the user continuously explored three floors of our building, each being more than 100m in the longest dimension. The results of this experiment are presented in Figure 8.

By traversing staircases a total of 18 times, 19 submaps were generated. These submaps were gradually connected to one another when overlapping areas of the map were discovered. By the end of the mission three complete maps representing each floor were created.

For a couple of very short submaps insufficient evidence was available to make connection to the main map — usually because the particular submap was very short in duration (sometimes less than a minute). These maps are not illustrated in the figures and remained disconnected from their relevant collective floor map. If subsequent information would become available later these maps could then be connected to the main map.

A. Multi-Session Mapping combining a Human and a Robot

As described above, the mapping and remapping of a particular area becomes equivalent to connecting independent pose trajectories to one another. Figure 9 illustrates the map in Figure 8 subsequently extended by the addition of information collected by a (teleoperated) robot. The human-created map is colored orange and the robot's contributions are in purple. This particular map contains 17 inter-user loop closures found first by visual loop-proposal means but later further connected using geometric proposals.

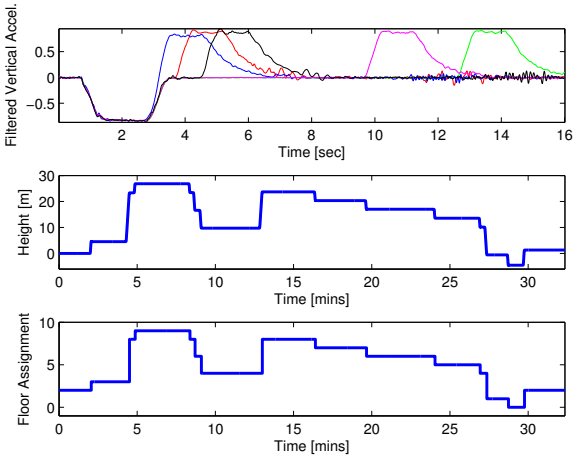
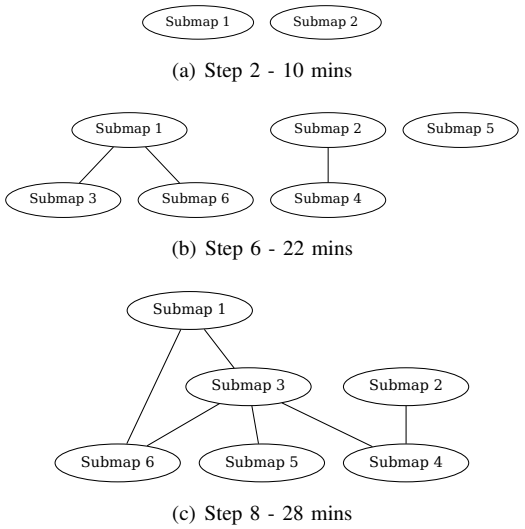


Fig. 6. Top: Filtered IMU Z-component for 5 elevator rides. Middle: Height can be estimated by integrating the elapsed time between the characteristic accelerations and decelerations. Bottom: Using this approach we successfully estimated the floor transitions across a 35 minute experiment across 10 floors - beginning and ending on Floor 2 of MIT's Stata Center.



(a) Step 2 - 10 mins

(b) Step 6 - 22 mins

(c) Step 8 - 28 mins

Fig. 7. Evolution of the submap topology for Floor 3. As time evolves different (and non-consecutive) submaps are connected to one another until all submaps form a single heavily-connected and unified map.

In our future work, we will extend our approach to support mixed teams of robots and humans simultaneously exploring and sharing maps (wirelessly) when encounters occur. This area of research opens interesting trade offs between the distribution of map information and communication constraints.

VII. Conclusion

In this paper we present the algorithms and systems developed to enable natural exploration and map-building of multi-floor building by a person carrying a multi-sensor rig.

Our robust system demonstrates real-time, stable operation in a challenging application for explorations of considerable scale. However, for a practical system to scale limitlessly, consideration must be given to 1) the continuously growing pose graph and 2) addition of false loop-closures. In the case of the former, sparsification of the pose graph by determining the informational contribution of individual scans, as suggested by [23], seems promising. This approach suggests a SLAM system which scales with area explored rather than time of exploration.

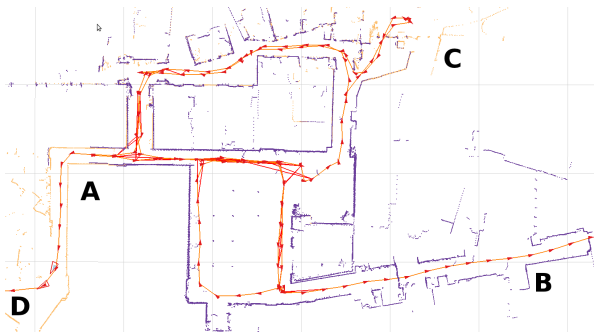


Fig. 9. A map combining two separate explorations — one from a human system (C to D, orange) and a second from a robot (A to B, purple).

As the size of exploration increases, false loop-closures become inevitable. Introspection on the low-level SLAM system — completely removing a small set of patently incorrect constraints — seems necessary. An interesting approach proposed in [24] achieves a robustness to false loop-closures by disabling those with inordinate optimization cost.

A number of other avenues of future work remain. As mentioned above, integration of the foot-mounted IMU system mentioned in Section II would be a promising area of future work — particularly when map segments cannot be definitively reconnected with the main map. Even though in this work only the color information from the RGB-D camera is used a natural extension is to use the depth information and a more visually-driven approach — similar to the authors RGB-D localization work [25].

Acknowledgments

This work was partially sponsored by MIT Lincoln Laboratory under Air Force Contract FA8721-05-C-0002, in collaboration with MIT Lincoln Laboratory Technical Staff members Anthony Lapadula and Sean Winkler. This work was also partially supported by the Office of Naval Research under research grants N00014-10-1-0936, N00014-11-1-0688 and N00014-12-10020. Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

References

[1] A. Cavanaugh, M. Lowe, D. Cyganski, and R. Duckworth, "Bayesian fusion algorithm for precision personnel location in indoor environments," in *Institute of Navigation International Technical Meeting*, January 2011.

[2] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *Computer Graphics and Applications, IEEE*, vol. 25, pp. 38 –46, Nov. 2005.

[3] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proceedings of the 10th International Conference on Ubiquitous Computing*, (New York, NY, USA), pp. 114–123, ACM, 2008.

[4] S. Moafipoor, D. A. Grejner-Brzezinska, and C. K. Toth, "Multi-sensor personal navigator supported by adaptive knowledge based system: Performance assessment," in *IEEE/ION Position, Location and Navigation Symposium*, May 2008.

[5] P. Robertson, M. Angermann, and B. Krach, "Simultaneous localization and mapping for pedestrians using only foot-mounted inertial sensors," in *Proceedings of the 11th International Conference on Ubiquitous Computing*, vol. 30, September 2009.

[6] M. Angermann and P. Robertson, "Footslam: Pedestrian simultaneous localization and mapping without exteroceptive sensorshitchhiking on human perception and cognition," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1840–1848, 2012.

[7] T. Liu, M. Carlberg, G. Chen, J. Chen, J. Kua, and A. Zakhor, "Indoor localization and visualization using a human-operated backpack system," in *International Conference on Indoor Positioning and Indoor Navigation*, September 2010.

[8] N. Naikal, J. Kua, G. Chen, and A. Zakhor, "Image augmented laser scan matching for indoor dead reckoning," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, October 2009.

[9] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf, "A flexible and scalable slam system with full 3D motion estimation," in *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, November 2011.

[10] J. A. Hesch and S. Roumeliotis, "Design and analysis of a portable indoor localization aid for the visually impaired," *Intl. J. of Robotics Research*, vol. 29, pp. 1400–1415, 2010.

[11] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, pp. 167–193, Springer Verlag, 1990.

[12] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (SLAM): Part I," *Robotics & Automation Magazine*, vol. 13, pp. 99 –110, Jun 2006.

[13] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *National Conf. on Artificial Intelligence*, (Edmonton, Canada), AAAI, 2002.

[14] F. Dellaert, "Square Root SAM: Simultaneous location and mapping via square root information smoothing," in *Robotics: Science and Systems (RSS)*, (Cambridge, MA), 2005.

[15] E. Olson, "Real-time correlative scan matching," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, (Kobe, Japan), pp. 4387–4393, Jun 2009.

[16] A. Bachrach, S. Prentice, R. He, and N. Roy, "Range - robust autonomous navigation in GPS-denied environments," *J. of Field Robotics*, vol. 28, pp. 644–666, September 2011.

[17] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2011.

[18] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robotics*, vol. 24, pp. 1365–1378, Dec 2008.

[19] C. Cadena, D. Gálvez, F. Ramos, J. Tardós, and J. Neira, "Robust place recognition with stereo cameras," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, (Taipei, Taiwan), October 2010.

[20] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

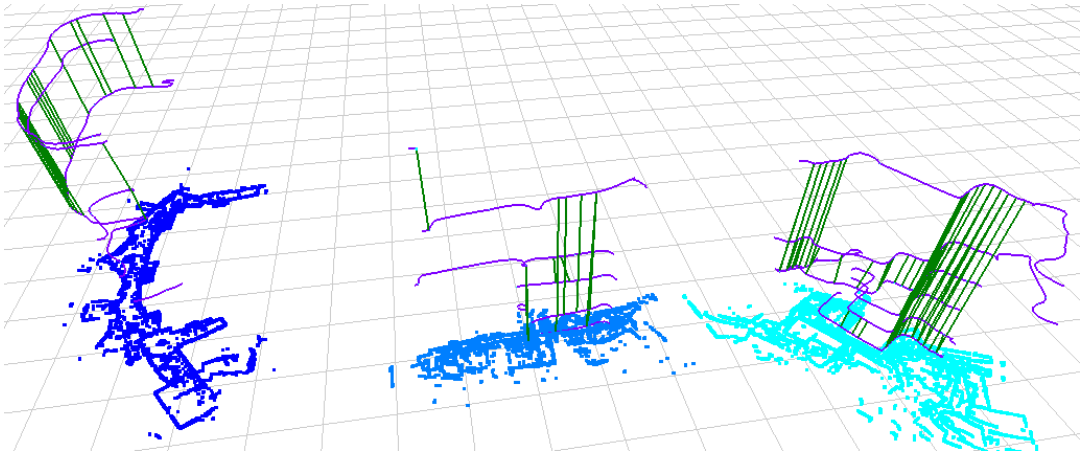
[21] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 2161–2168, 2006.

[22] B. Kim, M. Kaess, L. Fletcher, J. Leonard, A. Bachrach, N. Roy, and S. Teller, "Multiple relative pose graphs for robust cooperative mapping," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, (Anchorage, Alaska), pp. 3185–3192, May 2010.

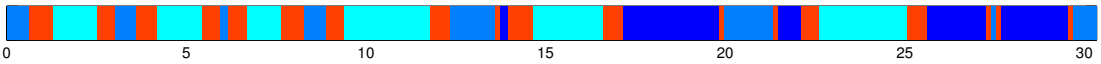
[23] H. Kretzschmar, C. Stachniss, and G. Grisetti, "Efficient information-theoretic graph pruning for graph-based SLAM with laser range finders," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Sep 2011.

[24] N. Sunderhauf and P. Protzel, "Towards a robust back-end for pose graph SLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2012.

[25] M. Fallon, H. Johannsson, and J. Leonard, "Efficient scene simulation for robust monte carlo localization using an RGB-D camera," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, (St. Paul, MN), May 2012.

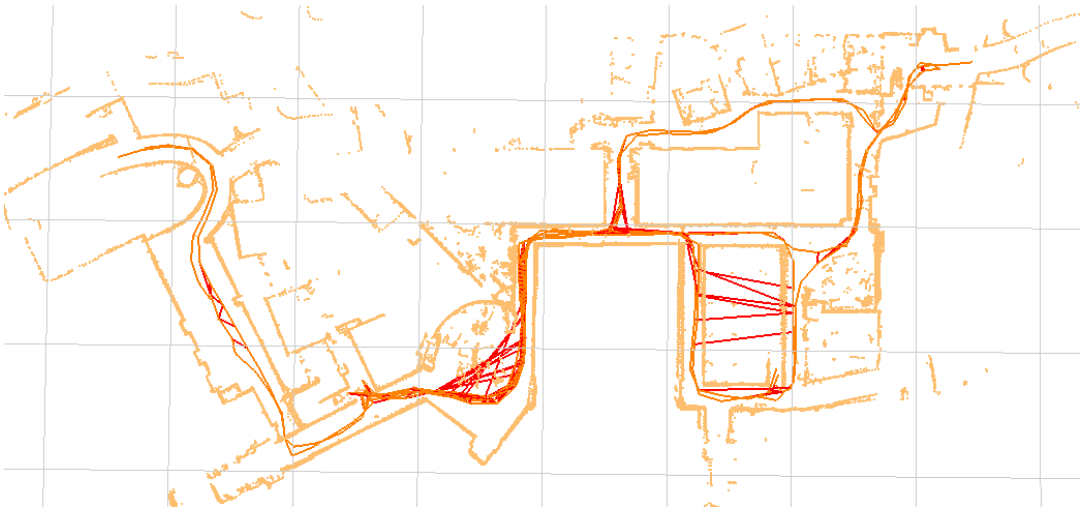


(a) Maps of 3 floors simultaneously built in one instance of exploration. This figure shows time represented as the upward axis. Approximately horizontal lines represent user motion during each floor traversal and vertical lines indicate confirmed loop closures. From left to right the floors illustrated are the 1st, 2nd and 3rd floors.



(b) Time evolution of the user's location during exploration, in minutes. Three blue colors represent individual floors and red staircase traversals.

(c) First floor - with 10 meter grid



(d) Third floor - with 30 meter grid.

Fig. 8. Continuous Man Portable Exploration over 3 floors with 18 staircase traversals and 19 submaps.