Gyroscope Assisted Scalable Visual Simultaneous Localization and Mapping

Benzun Wisely Babu
Worcester Polytechnic Institute
Worcester, MA 01609
Email: bpwiselybabu@wpi.edu

Dr. James Duckworth
Worcester Polytechnic Institute
Worcester, MA 01609
Email: rjduck@wpi.edu

Dr. David Cyganski
Worcester Polytechnic Institute
Worcester, MA 01609
Email: cyganski@wpi.edu

***Abstract*—This paper describes the development and evaluation of an indoor localization algorithm using *Visual Simultaneous Localization and Mapping (VSLAM)* aided by gyroscope sensor information. Indoor environments pose several challenges which could cause a vision only system to fail due to tracking errors. Investigation revealed significant feature loss in a vision only system when traversing plain walls, windows and staircases. However, the addition of a gyroscope helps in handling such difficult conditions by providing additional rotational information. A portable system consisting of an *Inertial Measurement Unit (IMU)* and a stereo camera has been developed for indoor mapping. The images and gyroscope rates acquired by the system are stored and post-processed using a new *Gyroscope Assisted Scalable Visual Simultaneous Localization and Mapping Algorithm (GA-ScaViSLAM)*. The algorithm has been evaluated for data-sets collected at Atwater Kent building, Worcester Polytechnic Institute. This algorithm was found to be more robust in comparison to the vision only system. The Ga-ScaViSLAM was found to have an error (rms) of 0.6 m in the indoor environments tested over a total path length of 77m.**

I. INTRODUCTION

Navigation in an unknown indoor environment requires reliable localization. Unlike outdoor environments, there is limited assistance from external agents such as GPS which provide information about the absolute position of the system. *Simultaneous Localization and Mapping (SLAM)* was developed to address the issue of localization in an environment where pre-cached maps and absolute pose information are not available.

Vision based SLAM (VSLAM) has gained popularity due to the availability of inexpensive cameras and low cost high speed processors. Most VSLAM algorithms rely on repeated observation of distinct features to determine the camera pose. The stereo disparity observed using nearby features provides information about translation of the camera while the disparity due to distant features provides information about the orientation of the camera.

However, there can be loss of features in indoor environments due to changing lighting conditions and textureless scenes. This is prominent while navigating through scenes containing plain walls, windows and stairs. This feature loss often leads to a tracking failure which degrades the VSLAM performance. The effect is also amplified while the system is taking a turn in indoor corridors due to sudden changes in both the sparsity and trajectory speed of the available features.

Sensors that augment vision can be used to assist in overcoming this tracking failure. Vision based algorithms are good at handling changes over long duration but fail to correctly observe sudden rotation rates. On the other hand, a gyroscope measures accurate angular rates over short period about its axis but drifts considerably over a longer period. Hence, a combination of vision and gyroscope should complement each other and provide more robust localization including handling turns inside the building.

In this paper, we present the *Gyroscope Assisted Visual Scalabe Localization and Mapping (GA-ScaViSLAM)* algorithm which improves robustness and accuracy of the existing scalable visual simultaneous localization and mapping algorithm ScaViSLAM[1]. A huge improvement in localization is achieved with the addition of a gyroscope. The algorithm is tailored for indoor operation and is capable of localization at stairs, narrow corridors, and other common indoor conditions that can cause problems while reducing long term orientation drifts.

The following section gives a brief overview of the existing techniques. A more complete description of the challenges encountered in indoor SLAM is presented in section III. The system design developed for the combined vision and gyroscope system is presented in section IV. Next, the approach developed to integrate the stereo system with the gyroscope from an algorithm point of view is discussed in Section V. It is followed by Section VI which discusses the experimentation carried out to validate the system performance. Finally Section VII presents a summary of the results.

II. BACKGROUND

Filtering and optimization are two main approaches to visual SLAM. The filtering approach performs estimation of the system location and map using a non-linear filter such as an Extended Kalman Filter (EKF) [2][3] while an optimization approach uses techniques such as Bundle Adjustment (BA) [4] or Iterative Closest Point (ICP) [5] to solve for the state of the map and the camera.

Civera et al. [6] demonstrated visual SLAM using a monocular camera for augmented reality applications. It used an extended Kalman filter to track the features in the map. Since the number of states of the filter increases with the number of features tracked, it does not scale well to large environments.

Furthermore it does not have the ability to perform loop closure.

Bundle adjustment is a concept used in photogrammetry to estimate the ego motion and position of features in a small scene [7][8]. It can be generalized as a non-linear optimization approach. In recent years there has been a shift towards optimization based techniques for solving SLAM. Agrawal [9], Lu and Milios [10] demonstrated representing SLAM as a pose graph optimization problem is efficient and robust to solve loop closure.

Klein and Murray [11] introduced a technique called *Parallel Tracking and Mapping (PTAM)* that separated SLAM into two separate threads. In one of the threads, the camera motion is estimated while the other thread handles optimization to determine the feature positions. They were able to achieve real-time performance by using a subset of the captured image frames called keyframes. PTAM provides good mapping over a small area but is not capable of handling large areas.

Konolige and Agrawal [12] used frame matching to demonstrate a visual SLAM technique that performs optimization over the keyframe poses. It demonstrates the ability to perform loop closure with a reduced skeleton set of the frames. They store only the relative non-linear constraint between the keyframes thus allowing them to track within a larger area with less error.

Strasdat et al. [1] introduced Scalable Visual SLAM (ScaViSLAM) aimed at providing a solution that can be used for both large scale environments and small scale objects. It merged frame based SLAM that performs well over a long range with bundle adjustment which is capable of providing precise localization and mapping over a small range. ScaViSLAM performs a combined optimization over two windows. However, it fails in situations where there is tracking failures due to conditions which often arise in the indoor environment.

Leutenegger et al. [13] presented a close integration of optimization based SLAM with an Inertial Measurement Unit (IMU). They were able to demonstrate real time mapping with tight coupling between the IMU and the camera is available. They had a larger number of internal states to optimize and these demonstrations were limited to wide corridors with straight paths in which feature sparsity and trajectory speeds are not a dominant issue.

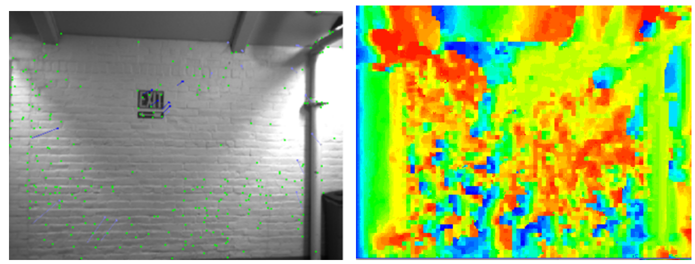
Chameleon [14] uses stereo camera measurement fused with an Inertial Measurement Unit for navigation and mapping. They use an Extended Kalman filter based SLAM. They show successful navigation and mapping but do not to perform loop closure. In situations where there is considerable feature loss the system drifts.

III. Indoor SLAM - Potential Problems

Indoor localization using vision is plagued by conditions where the loss of features can lead to catastrophic failure of the VSLAM algorithm. Indoor imaging conditions are highly dependent on the layout and lighting within the building. A visual SLAM solution capable of handling indoor localization needs to overcome three practical challenges presented below.

A. Plain Walls

Presence of texture in scenes is important for computer vision algorithms for feature identification and tracking. Plain walls and repeating brick structures (Fig. 1a) are common indoor conditions lacking texture. Lack of distinct texture results in poor correspondence and wrong disparity being calculated (Fig. 1b). This prevents accurate localization and mapping.



(a) Wall near a turn (b) Disparity image of wall

Fig. 1: A noisy disparity image is generated for a plain wall by a stereo disparity algorithm. Ideally the disparity image should have appeared as a uniform linear gradient.

B. Windows

Non-uniform brightness levels violate the constant intensity assumption that feature detection and tracking algorithms rely on. Sudden changes in lighting near windows affect the brightness of the images and causes clipping at either white or black intensity extremes. Thus based on the exposure time, either the indoor environment (Fig. 2a) or the scene outside the window (Fig. 2b) is prominent. This sudden change in the imaging condition can lead to loss of features being tracked and introduce error into the system.

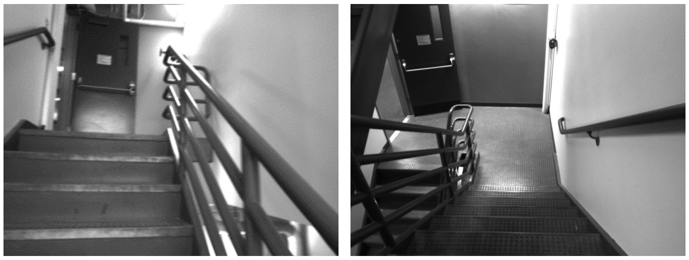
(a) Exposure tuned to indoor lighting (b) Exposure tuned to outdoor lighting

Fig. 2: The dramatic change of intensity levels of features and resulting clipping observed based on the camera exposure.

C. Staircase

Large depth variations in a local image neighborhood combined with significant camera motion prevents successful tracking. Staircases have features which lie on a plane nearly perpendicular to the imaging plane (Fig. 3). As this plane approaches the scene, the distribution of the features gets stretched or features disappear between video frames.

Feature tracking algorithms such as cross correlation perform poorly as they depend on the feature distribution remaining approximately constant.



(a) Facing a staircase. (b) moving down a staircase

Fig. 3: The neighborhood of features near staircase changes dramatically as we climb up or down the stairs.

The challenges described are more catastrophic when encountered near turns where there is a sudden scene change (high feature velocity) which compounds the feature loss. These challenges are fundamental to image tracking in the indoor environment and can be overcome with additional properly integrated sensors.

Since a gyroscope is capable of providing additional observation during a turn, an Inertial Measurement Unit (IMU) can be added to complement the vision based system.

IV. System Design

The system we designed consists of a custom hardware created using a *Bumblebee XB3* camera attached to a *NavChip* Inertial Measurement Unit (Fig. 4). A *Next Unit of Computing* processing unit, consisting of a solid state storage device is used to collect data from the IMU and the stereo camera. The processor communicates with the image trigger circuit using a serial port to set the desired trigger pulse duration. The stereo camera is configured to perform image capture on trigger pulses. The stereo images are captured using the narrow base-length (12cm) camera pair. The narrow stereo camera pair allows better disparity at a short range. The images are captured at 640x480 resolution and transmitted to the computer using a *Firewire 800* connection.

A. Stero Camera Model

The stereo camera observations are modeled as a three dimensional vector where the first two components u_l, v_l represent the pixel location as observed by the left camera. The third component u_r is the row measurement in the right camera frame.

$$\hat{z_b}(\boldsymbol{T_i}, \boldsymbol{X}) = \begin{pmatrix} u_l \\ y_l \\ u_r \end{pmatrix} = \begin{pmatrix} f_x \frac{y_1}{y_3} + p_1 \\ f_y \frac{y_2}{y_3} + p_2 \\ f_x \frac{y_1 - b}{y_3} + p_1 \end{pmatrix} \text{ where } y := T_i.X_k \tag{1}$$

In equation 1, the focal length (f_x, f_y) and the camera center (p_1, p_2) is obtained using camera calibration. X_k represents a feature observed in the world coordinate frame.

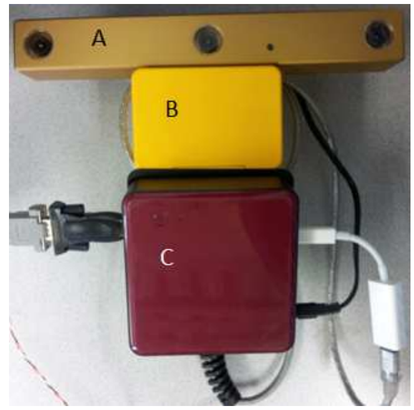


Fig. 4: The handheld system used to capture the data for the system. A - Bumblebee XB3 , B - NavChip IMU, C - Intel Next Unit of computing

B. Gyroscope Model

The gyroscope rates ($\dot{\boldsymbol{\theta}}$) are modeled as three dimensional rotation about its axis. If $(\theta_x, \theta_y, \theta_z)$ describe the rotation about the three axis (x, y, z), 2 describes the gyroscope rate observed in the camera frame. The rotation rates are assumed to be corrupted by the bias noise (n_b).

$$\dot{\boldsymbol{\theta}}_{\boldsymbol{c}} = q2r(Q_c^{imu}) * \begin{pmatrix} \dot{\theta}_x + n_{b_x} \\ \dot{\theta}_y + n_{b_y} \\ \dot{\theta}_z + n_{b_z} \end{pmatrix} \tag{2}$$

The quaternion (Q_c^{imu}) describes the rotation between the IMU and the camera frame. The function *q2r()* converts a quaternion to rotational matrix.

C. Coordinate frames

There are three main coordinate frames in the system (Fig. 5):

1) World frame (F_w) - The base frame with respect to which all the elements in the environment are defined.
2) Camera frame (F_c) - The frame with respect to the left camera center. All the features observed by the camera are represented in this coordinate frame initially.
3) IMU frame (F_i) - The frame with respect to which the IMU measurements are observed initially.

D. Calibration

A one time calibration procedure is need to determine the transformation between the IMU and the camera. The transformation T_{imu}^c consist of a rotation quaternion Q_{imu}^c and a displacement vector d_{imu}^c. Since we are only interested in the gyroscope rates, the rotation quaternion between the IMU and the camera (Q_{imu}^c) is found using the calibration technique demonstrated in [15]. Also every time the system is started, the gyroscope bias is estimated by keeping the system stationary for a 500 frame capture (approx 50 seconds).

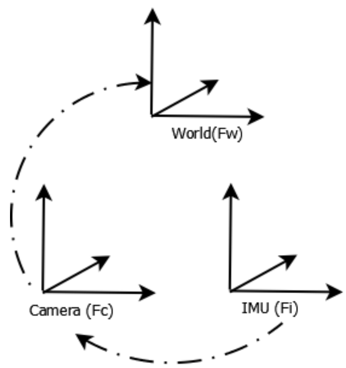


Fig. 5: Coordinate Frames

V. Integrating Gyroscope

Stereo images are captured using the narrow base-length stereo camera pair while the rotational rates are captured using the 200Hz (f_{imu}) gyroscope. To synchronize the camera with the IMU, a trigger is generated for every n gyroscope observations as described in Section IV. This trigger is used to initiate a stereo image capture at rate f_t (Eq.3).

$$Image\,capture\,rate(f_t) = \frac{200}{n}Hz \tag{3}$$

In indoor environments, automatic exposure helps to capture scenes with varying lighting conditions better but this can lead to loss of IMU trigger in scenes with low lighting conditions. Since the frame rate is determined by the IMU trigger rate (Eq.3), the camera exposure time (t_{exp}) needs to be below the IMU trigger period (Eq.4) to ensure that an image frame is generated for every IMU trigger.

$$Exposure\,Time : t_{exp} < \frac{1}{f_t} \tag{4}$$

The images and the gyroscope observations are captured using independent driver threads and stored for offline processing (Fig. 6). This ensures that there is no loss of data from the sensors.

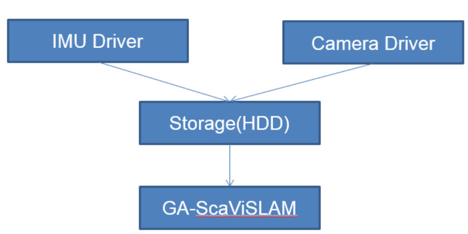


Fig. 6: The data flow in the system

A. Threading Architecture

Ga-ScaViSLAM uses a similar threading architecture as proposed in ScaViSLAM[1] (Fig. 7). The algorithm is defined in two logical blocks: the front end where the features are detected and initialized and a back end where non-linear optimization is carried out to perform mapping.

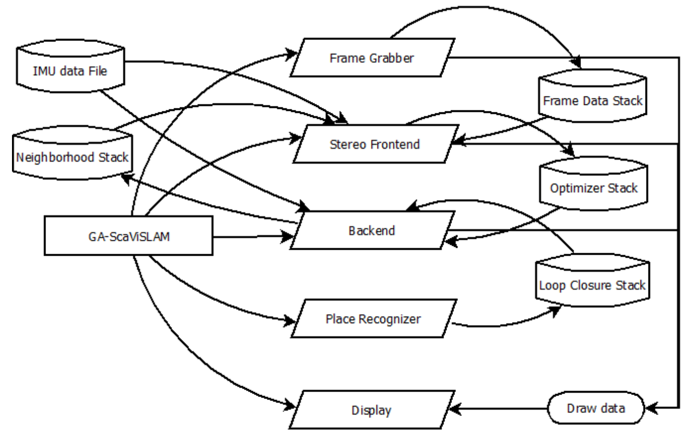


Fig. 7: The threading architecture used in Ga-ScaViSLAM

The algorithm launches multiple separate threads for near real time processing. The frame grabber thread loads the stereo image pair into the frame data stack. The stereo front end uses the frame data from the stack and calculates the disparity images using semi-global block matching. The back end thread performs double window optimization [1]. A place recognizer thread is used to find loop closure and incorporate the loop constraints into the optimization. Finally a display thread is used to provide visual feedback. The gyroscope rates are incorporated both in the front end and the back end of the stereo SLAM system.

B. Gyroscope in Front end

The objective of the front end is to detect features and perform an initial estimate of the pose of the camera. The features from a frame are extracted using a grid based FAST [16] algorithm. The pose of a keyframe is determined by the dense tracking algorithm described below. The pose estimation in the front end is used as an initial estimate in the back end convex optimizer. Since a convex optimizer only produces the local minima and not the global minima the initial estimate needs to be close to the actual pose. In an indoor environment the estimation of orientation by a vision only system is poor due to the lack of distant features. In Ga-ScaViSLAM (Fig. 8) the gyroscope is used to assist the front end in determining the orientation. This orientation quaternion between two frames (k,k+n) can be calculated using equation 5.

$$Q^k_{k+n} = Q^{-1}_{k+n} * \prod_{i=k}^{k+n} Q_i \tag{5}$$

1) Dense Tracker: The dense tracker [17] uses intensity consistency to estimate the transform between the frames. All the points in the images are used to produce a estimate of the transformation between two frames. It uses the Levenberg Marquardt (LM) algorithm to find the transformation that minimizes the back projection error of the points in the scene. The gyroscope orientation estimate is used to initialize the LM algorithm, thus making the dense tracker more robust and preventing it from getting stuck in local minimum.

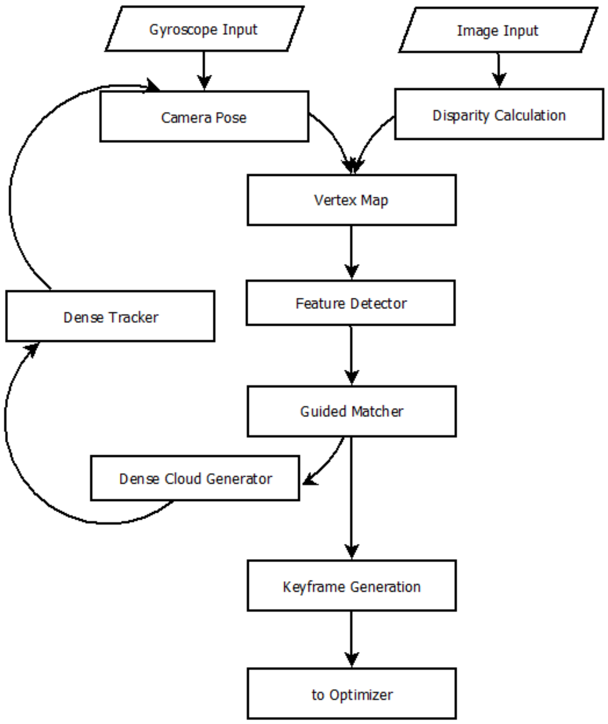


Fig. 8: The front end in Ga-ScaViSLAM

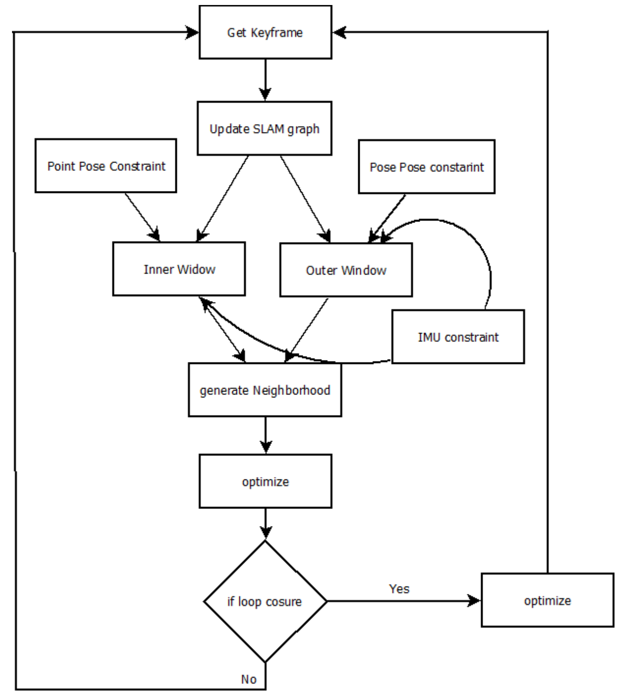


Fig. 9: The back end in Ga-ScaViSLAM

2) Keyframe Generation: A better accuracy can be achieved by tracking a greater number of features in the environment [18]. In narrow corridors and corners a large numbers of features go out of field of view very quickly. But, only a subset of the total frames observed, these are called keyframes. They are used in the back end for optimization. To capture as many features as possible during turns, Ga-ScaViSLAM increases the number of keyframes generated during turns to prevent the back end from failing.

C. Gyroscope in Back end

The back end (Fig. 9) is responsible for generating localization information based on the motion from the beginning of the operation of the system. The back end uses a graph that maintains the constraints and relationships between the keyframes sent by the front end. The keyframes based on the history are subdivided into inner and outer window. All optimization is carried out using the g2o [19] optimization package.

1) Inner Window: In the inner window the optimization function tries to minimize the feature re-projection error combined with the orientation error between the gyroscope and the visual system.

2) Outer Window: In the outer window a pose-pose constraint is enforced. Hence the optimization function tries to minimize the predicted transformation with the transformation observed from different keyframes in the path travelled by the system. Additionally, the orientation constraint as defined by the rotational change observed by the gyroscope between the keyframe is added as a constraint to the outer window.

3) Objective Function: After adding the constraints from the inner and the outer window, the objective function is represented by the equation 6. The first summation minimizes the re-projection error, the second summation minimizes the relative pose between the two frames where $v_{ij} := \log_{SE(3)}(T_j^i * T_i * T_j^{-1})$. finally the third summation minimizes the error between the quaternion rotation given by the IMU and the observed quaternion rotation between two keyframes.

$$\chi^2 = \sum_{z_{ik}} (\boldsymbol{z_{ik}} - \boldsymbol{\hat{z}}(\boldsymbol{T_i}, \boldsymbol{X_k}))^2 + \sum_{T_{ij}} v_{ij} \Lambda_p v_{ij} + \sum_{Q_{ij}} \exp(Q_{ij} * Q_{ji}^{imu}) \Lambda_g \exp(Q_{ij} * Q_{ji}^{imu}) \quad (6)$$

4) Loop Closure: In the back end, loop closure using place recognition proposed by Strasdat et al. [1], is used. If loop closure occurs, an additional constraint is added to the SLAM graph. Again optimization is carried out to improve the localization accuracy.

VI. Experimentation and Results

All data-sets were collected using the hand-held device shown previously in Fig. 4. A stereo image was generated at every 20^{th} IMU observation ($n = 20$). The image capture rate was 10Hz.

The data-sets were captured inside the Atwater Kent building. The experiment was conducted over three floors with varying lighting conditions. The first and the second floor have wide corridors with windows while the third floor has narrow corridors without windows. The stairs between the floors are moderately lit and have plain walls.

In each of the experiments an initial idle time of around 500 frames was used to determine the bias for the gyroscope calibration. We compare the performance of the vision only ScaViSLAM system [1] with Ga-ScaViSLAM.

A. Ground Floor Data-set

This data-set was collected in the ground floor of Atwater Kent building. Fig. 10a shows the general path traveled overlayed on the floor plan. The data-set is characterized by wide well-lit corridors. A total distance of approximately 80m was traveled during the test.

The localization result overlayed on the floor plan is presented in the Fig. 10b. In the vision only system, the turns are estimated poorly. Also there is a break in tracking due to change in lighting near the second turn where windows are located. Loop closure occurs at the end, but the propagated correction is not able to compensate for the tracking loss that occurred.

Fig. 10c shows the path generated by Ga-ScaViSLAM. We observe a more accurate path, that accurately tracks turns and performs a successful loop closure.

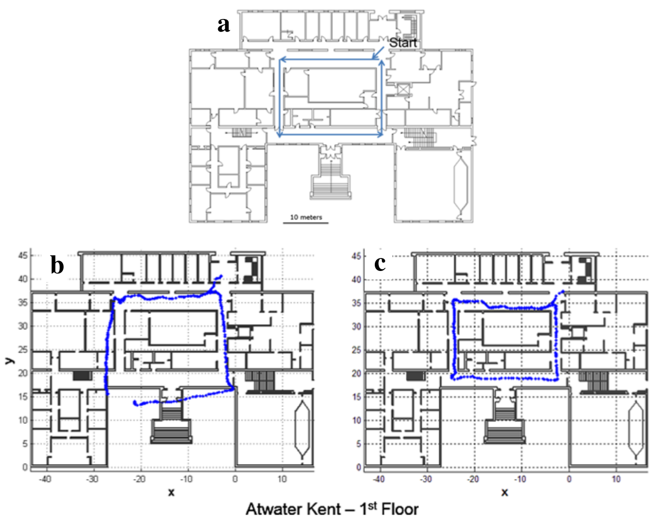


Fig. 10: The path traveled in Atwater Kent ground floor (a). The path generated by vision only system is overlayed on the ground floor plan (b). The path generated by Ga-ScaViSLAM overlayed on the ground floor plan (c).

B. Two Floors Data-set

To perform a test in a larger environment, we start at the second floor corridor, climb up the stairs to the third floor, make a loop on the third floor and return back to the second floor climbing down the stairs to complete the loop. Fig. 11 shows the path travelled (approx 160m). In addition to being longer than the previous data-set it also has varying lighting conditions, stairs and narrow corridors.

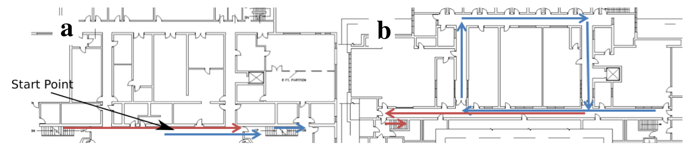


Fig. 11: The path travelled in the (a) second floor, (b) third floor.

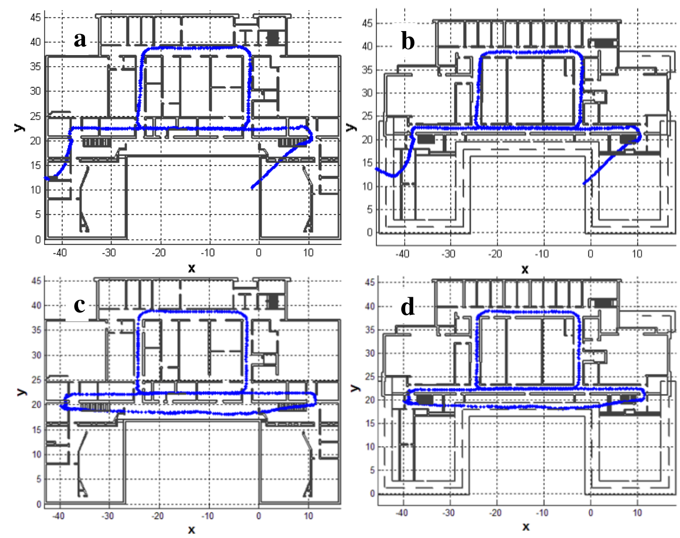


Fig. 12: The path generated by vision only system overlayed on the second floor (a). The path generated by vision only system overlayed on the third floor (b). The path generated by Ga-ScaViSLAM overlayed on the second floor (c). The path generated by Ga-ScaViSLAM overlayed on the third floor (d).

Fig. 12a, 12b shows the vision only system fails completely while traversing stairs and only completes loop closure on the third floor. It fails to recognize the loop closure that occurs in the second floor.

Fig. 12c, 12d shows the path generated by Ga-ScaViSLAM. It does not have the same failures and is able to achieve both vertical and horizontal loop closure.

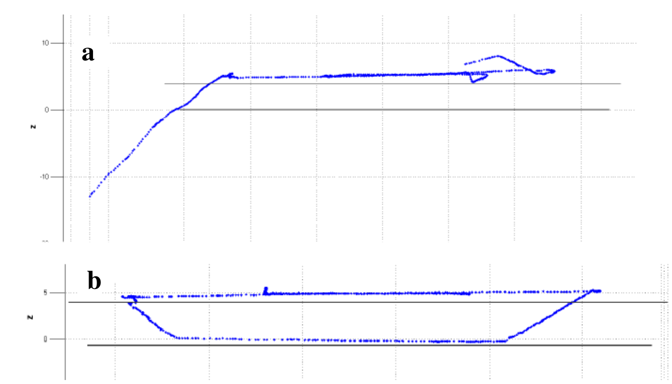


Fig. 13: The elevation result in the vision only system(a) and the elevation result in the Ga-ScaViSLAM(b).

Fig. 13 shows the result of the path generated by both the systems while climbing stairs. We observe that the vision only system has huge drifts while Ga-ScaViSLAM is able to traverse the stairs without much loss of accuracy.

C. Third Floor Data-set

Finally to evaluate the position accuracy, this dataset was collected with ground truth data points that were surveyed on the floor. Fig. 14 shows the path traveled while collecting this data-set. This data-set has narrow poorly lit corridors with texture-less walls.

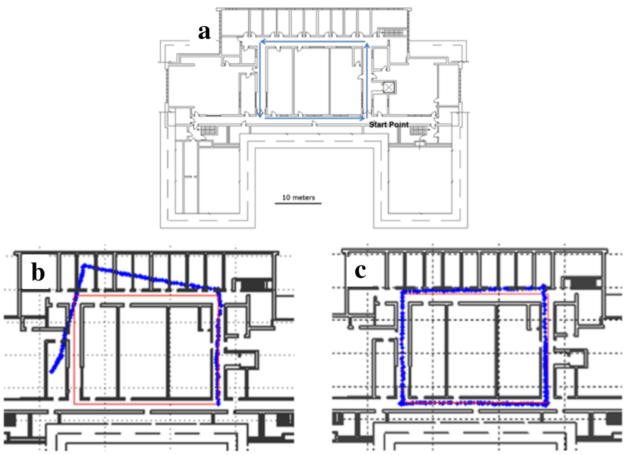


Fig. 14: Red - ground truth, Blue - path generated. The path travelled in the 3rd floor(a). The path generated by vision only system overlayed on the ground truth(b). The path generated by the Ga-ScaViSLAM overlayed on the ground truth(c)

Fig. 14b Using the vision only system, the algorithm fails at the third turn where there is a texture-less wall. This break in the algorithm is due to failure in tracking as discussed in the section III. The path traversed by the vision only system had an RMS error of 1.85m. The RMS error for the traversed path was calculated for the system after aligning the map.

Fig. 14 shows the path generated with the Ga-ScaViSLAM. The surveyed points and path generated align very well indicating small error from the ground truth. After aligning the map, an RMS error of 0.6m is observed for the narrow indoor corridor loop with a total travelled path distance of 77m.

VII. Conclusion

The difficulties in indoor visual SLAM are presented. A portable hardware system for indoor mapping using a stereo system and gyroscope capable of overcoming the difficulties was developed. The Gyroscope assisted Scalable Visual Simultaneous Localization and Mapping(Ga-ScaViSLAM) algorithm combined images and gyroscope information to demonstrate robust indoor localization. The gyroscope information was successfully used at both feature extraction front end and optimization back end. Experimentation was carried out using real data collected from a multi-floor indoor environment. It has successfully demonstrated the ability to perform localization and mapping in building environments with stairs, windows and plain walls. Compared to a vision only system, the Ga-ScaViSLAM is more robust. Additionally, it also produces more accurate maps with a RMS error of only 0.6m in a narrow dimly lit corridor loop over a path distance of 77m.

Acknowledgment

We gratefully acknowledge the support of Airbus Defence and Space for this research, which has been performed under research contract ASV/LP/PR/SPA/2011-009.

References

[1] H. Strasdat, A. J. Davison, J. M. M. Montiel, and K. Konolige, "Double window optimisation for constant time visual slam." in *ICCV*. IEEE, 2011, pp. 2352–2359.

[2] S. Thrun, W. Burgard, D. Fox, H. Hexmoor, and M. Mataric, "A probabilistic approach to concurrent mapping and localization for mobile robots," in *Machine Learning*, 1998, pp. 29–53.

[3] S. Betge-Brezetz, P. Hebert, R. Chatila, and M. Devy, "Uncertain map making in natural environments," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 2, 1996, pp. 1048–1053 vol.2.

[4] A. Eudes, M. Lhuillier, S. Naudet-Collette, and M. Dhome, "Fast odometry integration in local bundle adjustment-based visual slam," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, Aug 2010, pp. 290–293.

[5] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM - 3D mapping outdoor environments: Research articles," *Journal of Field Robot.*, vol. 24, no. 8-9, pp. 699–722, Aug. 2007. [Online]. Available: http://dx.doi.org/10.1002/rob.v24:8/9

[6] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel, "1-point ransac for EKF-based structure from motion," in *International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 3498–3504.

[7] D. C. Brown, "A solution to the general problem of multiple station analytical stereo triangulation," Technical Report RCA-MTP Data Reduction, Patrick Airforce Base, Florida, Tech. Rep. 43 (or AFMTC TR 58-8), 1958.

[8] B. Triggs, P. Mclauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment a modern synthesis," in *Vision Algorithms: Theory and Practice, LNCS*. Springer Verlag, 2000, pp. 298–375.

[9] M. Agrawal, "A lie algebraic approach for consistent pose registration for general euclidean motion," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 2006, pp. 1891–1897.

[10] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, pp. 333–349, 1997.

[11] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, 2007, pp. 225–234.

[12] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.

[13] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial slam using nonlinear optimization," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[14] J. Rydell and E. Emilsson, "Chameleon: Visual-inertial indoor navigation," in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, April 2012, pp. 541–546.

[15] J. Lobo and J. Dias, "Relative pose calibration between visual and inertial sensors." *International Journal of Robotic Research*, vol. 26, no. 6, pp. 561–575, 2007.

[16] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking." in *IEEE International Conference on Computer Vision*, vol. 2, October 2005, pp. 1508–1511.

[17] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *Proceedings of the 2011 International Conference on Computer Vision*, ser. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 2320–2327. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2011.6126513

[18] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Editors choice article: Visual slam: Why filter?" *Image Vision Comput.*, vol. 30, no. 2, pp. 65–77, Feb. 2012. [Online]. Available: http://dx.doi.org/10.1016/j.imavis.2012.02.009

[19] R. Kmmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization." in *ICRA*. IEEE, 2011, pp. 3607–3613.