Large Scale Monocular Vision-only Mapping from a Fixed-Wing sUAS

Michael Warren, David McKinnon, Hu He, Arren Glover, Michael Shiel, Ben Upcroft

Abstract -

This paper presents the application of a monocular visual SLAM on a fixed-wing small Unmanned Aerial System (sUAS) capable of simultaneous estimation of aircraft pose and scene structure. We demonstrate the robustness of unconstrained vision alone in producing reliable pose estimates of a sUAS, at altitude. It is ultimately capable of online state estimation feedback for aircraft control and next-best-view estimation for complete map coverage without the use of additional sensors. We explore some of the challenges of visual SLAM from a sUAS including dealing with planar structure, distant scenes and noisy observations. The developed techniques are applied on vision data gathered from a fast-moving fixed-wing radio control aircraft flown over a 1×1km rural area at an altitude of 20-100m. We present both raw Structure from Motion results and a SLAM solution that includes FAB-MAP based loop-closures and graph-optimised pose. Timing information is also presented to demonstrate near online capabilities. We compare the accuracy of the 6-DOF pose estimates to an off-the-shelf GPS aided INS over a 1.7km trajectory. We also present output 3D reconstructions of the observed scene structure and texture that demonstrates future applications in autonomous monitoring and surveying.

1 INTRODUCTION

Low-flying small Unmanned Aerial Systems (sUAS), otherwise known as Unmanned Aerial Vehicles (UAVs), have received increasing interest in recent years as a potentially cost-effective method of mapping and monitoring large areas of terrain. In contrast to other methods of environment mapping such as high-flying aerial surveys using manned aircraft and satellite-based sensing, sUAS provide a number

Michael Warren, David McKinnon, He Hu, Arren Glover, Michael Shiel and Ben Upcroft
Queensland University of Technology, 2 George St, Brisbane, QLD, Australia e-mail: michael.warren, d.mckinnon, h.hu2, aj.glover, michael.shiel, ben.upcroft@qut.edu.au

of unique advantages in terms of reduced size, weight, infrastructure and cost. Additionally, they are often not subject to the restrictions of full-sized aircraft, meaning they can fly closer to the ground and in areas of potential sensitivity, increasing resolution and accuracy. Our interest is in using only visual sensors on the platform to estimate pose while simultaneously generating high resolution, high accuracy maps of vast areas in a single Euclidean frame with fast turnaround and minimal human interaction, facilitating accurate reconstructions of environments for research and commercial analysis.

Vision is rapidly becoming the sensor of choice in robotic pose estimation and has the ability to produce dense, 3D point clouds of the environment. These sensors are small, lightweight and have low-power requirements. Motivated by these properties and recent advances in visual Structure from Motion (SfM), loop closure detection and pose optimisation techniques, this paper presents a high-performance monocular visual Simultaneous Localisation and Mapping (SLAM) system. The pipeline is applied on data from a low-flying sUAS that determines 6-DOF aircraft pose (up to scale) and scene structure (Fig. 1) over large ($>$1km) trajectories.

While traditionally LIDAR and other laser based systems have been used in mapping from the air, including some autonomous applications [14], their bulk, cost and power requirements mean they are restricted by both platform size and flight time. In addition, many airborne mapping systems are dependent on Global Positioning System (GPS), Differential GPS (DGPS) and inertial measurement, often in a filtering framework, for accurate vehicle pose estimation. However, their deficiencies (such as multi-pathing, lock-on failure, sensor drift) and heavy dependence on external infrastructure are well known [27].

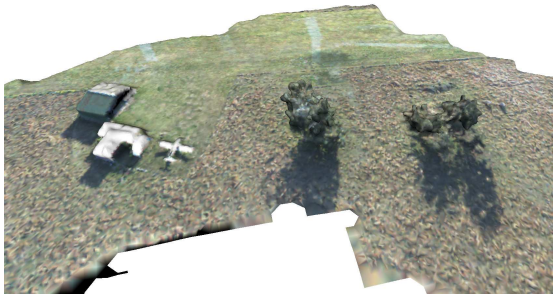

Fig. 1 A dense 3D mesh of rural farmland computed from sequential poses of visual SLAM from the air.

We demonstrate the ability of vision alone to generate pose capable of rivalling and ultimately complementing other sensors (GPS, INS etc.) in the airborne scenario for use in online state estimation feedback. We achieve this by careful implementation of algorithms for feature detection, pose estimation and feature triangulation, both in terms of speed and accuracy. In addition, our algorithm detects visual loop closures using openFAB-MAP [8] and applies these constraints in the pose-graph optimiser HOG-Man [9] to generate a refined pose and scene structure estimate. We show timing results demonstrating near online operation of the system and present a

pose comparison to a GPS aided INS system as ground-truth. In this paper we refer to visual pose estimation (or visual odometry) as the recovery of aircraft pose from visual SfM techniques in addition to simultaneous estimation of scene structure.

The rest of this paper is outlined as follows: Section 1.1 comprehensively reviews the literature on visual pose estimation and scene reconstruction on the ground and in the air. Section 2 describes our SLAM algorithm for pose and structure estimation. Section 3 describes the robotic platform and a collected dataset. Finally, Section 4 shows the results of the algorithm in generating pose on the gathered dataset. We compare results between raw SfM and a pose-graph optimised SLAM solution and compare both outputs to a GPS aided INS pose estimate. Additionally, an output mosaic and 3D reconstruction generated from recovered scene structure are demonstrated to indicate additional uses for the data.

1.1 Prior Work

Visual pose estimation without additional input has been demonstrated to great effect on the ground using iterative SfM techniques, both in iterative SfM based pose estimation [22, 28] and with the inclusion of loop-closure in the greater SLAM problem [17, 20]. Additionally, visual SfM has demonstrated highly accurate reconstruction of environments [23]. Such results demonstrate the suitability of vision to large scale pose estimation and mapping tasks.

In flying applications, vision has been used in a wide variety of scenarios [14]. It has received significant interest in small-scale online pose estimation tasks, particularly in quadcopter applications [2], but has often made assumptions about the environment such as texture [4] or geometry [7] to assist the estimate. Outdoors, vision has been used within a number of filtered algorithms to produce high quality pose estimates [3] and to generate both qualitative [5] and ground-truthed reconstructions of large scenes [16].

Iterative vision-only pose estimation, however, has only been used on small scale ($<$20m) airborne tasks, on relatively slow-moving craft such as airships and helicopters, and received little quantitative analysis. It has been shown in simulation [24] and small outdoor tasks [5], but with only qualitative assessments of accuracy.

A number of solutions exist that perform large scale visual mapping from aircraft [25] and sUAS [11] but these are characterised by their batch, strictly offline methods using photogrammetry techniques for image registration. Such methods are not suited to iterative online pose estimation and from a field and services point of view cannot be used to generate 3D maps or estimates in online time, meaning that autonomous decision-making cannot be performed. As a result, tasks such as ensuring full-coverage or view point path planning (next-best-view) cannot be achieved in a single flight, causing extended operational time and costs. In contrast, this paper details an approach that could be deployed as a SLAM pipeline for real-time visual pose estimation of a sUAS.

2 Methodology

Visual SLAM from a fixed wing aircraft at altitudes greater than 20-30m presents unique algorithmic challenges (particularly for SfM techniques), which has limited attempts at large-scale visual SLAM in this scenario. Firstly, the highly distant features impact accurate scene triangulation for small inter-camera baselines and introduce planarity issues for monocular cameras. Secondly, fast motion means feature tracks are fleeting and have only a short lifetime. Ultimately, the airborne scenario requires extreme robustness in the SfM algorithm to reliably estimate pose. This is dependent on reliable feature detection and tracking in addition to accurate triangulation and removal of noisy scene points. We have addressed these issues to demonstrate a visual SLAM pipeline for online aerial scenarios. This pipeline can be separated into:

- Pose and structure initialisation,
- An SfM approach for iteratively estimating camera pose and 3D structure of the observed scene,
- OpenFAB-MAP based loop-closure detection and,
- Pose-graph optimisation to generate a final SLAM estimate of pose and structure.

We additionally describe some algorithmic differences to the current literature. Finally, we generate 3D meshes from the optimised pose and scene structure as a demonstration of the quality of the final estimate.

2.1 Pose Initialisation

In order to set up the iterative SfM algorithm, an initial estimate of pose and scene is required. Initial pose is setup by computing the esssential matrix $\mathbf{E}_{1\leftrightarrow 2}$ using 5 matched features between the first and second images [21]. From this we fix the initial camera $\mathbf{P_1} = \mathbf{K}[\mathbf{I}|\mathbf{0}]$ at the origin of the global reference frame and extract the relative pose of the second camera as $\mathbf{P_2} = \mathbf{K}[\mathbf{R}|\mathbf{t}]$. The essential matrix is computed inside a MLESAC routine to eliminate poor initialisations and find the best subset of features for a good essential matrix.

Further, it is well known that a potential ambiguity exists in the pose generated from an essential matrix estimated from observing planar scenes [26]. The configuration of the airborne scenario often reflects this due to the distance of the scene and flat terrain. To avoid degenerate initialisations, we implement a test for degeneracy based on structure. As described in [5], a degenerate essential matrix will result in an unnatural spread in depth of a reconstructed scene. We use a similar algorithm as the structural degeneracy test. We first find the subset of points $\mathbf{X}$ with depth $\mathbf{Z}$ greater than their median depth $\tilde{\mathbf{Z}}$ in the coordinate frame of the origin camera:

$$\mathbf{Z_l} = \left\{\mathbf{Z} : \mathbf{Z} > \tilde{\mathbf{Z}}\right\} \tag{1}$$

We then find the mean of the depths of this subset, $\mathbf{Z_l}$, and divide it by the original median to generate the heuristic h:

$$h = \frac{\bar{\mathbf{Z}}_\mathbf{l}}{\tilde{\mathbf{Z}}} \tag{2}$$

This heuristic is then evaluated on a strict condition, where if $h > 1.2$, the initialisation is rejected and a new essential matrix computation is performed. We find that a significant number of initialisations are degenerate when applied to airborne data, requiring up to five repeats of the initialisation step. Once an initial camera pair is accepted, observed 3D structure is triangulated directly from the pair and their matched features.

2.2 Structure from Motion

Following a correctly initialised camera pair and 3D structure, our algorithm then follows an SfM routine to iteratively generate camera pose and scene structure from incoming frames. We split the algorithm into four main components:

1. Feature detection, matching and tracking
2. Motion update
3. Structure update
4. Sliding window bundle-adjustment

Additionally, we include openFABMAP based place recognition as a final step in the loop. The aim of our pose estimation and scene reconstruction task is to only use visual features. We do not consider motion models, filters or any additional sensors such as an IMU or GPS to aid the solution. Ultimately, however, this pipeline would be used inside a redundant framework that includes these sensors.

2.2.1 Feature Detection, Matching and Tracking

SIFT [18] features are detected in the image according to a bucketing scheme (400 equally spaced buckets per frame) to improve the spread of features, similar to that in [20]. This avoids grouping high density features in highly salient regions to help improve the pose constraint and more reliably track features throughout the image. We use a GPU implementation of SIFT detection and matching to approach an online time pose-update step.

We place requirements on descriptor matching that is stricter than other implementations to ensure that feature matches are accurate and tracks are generated only for the most salient features. We use SIFT as this has proven the most reliable in this scenario for both inter-frame and wide baseline matching, in part due to its 128-dimension descriptors. This is in contrast to the generally faster and more widely used SURF descriptor often used in ground applications where upright descriptors

(64-dimensions) are often acceptable. The dot product is used as the metric of a match between two descriptors instead of the more common Euclidean distance.

2.2.2 Motion Update

Using feature matches between the new and previous frames that have well initialised scene structure, the new camera pose is extracted using calibrated 3-point pose estimation [10], and uses a fourth point to disambiguate the 3 generated pose solutions. This is again performed inside a MLESAC estimator to generate the best possible camera location.

2.2.3 Structure Update

After a new camera pose is estimated, new scene points $\mathbf{X}$ that meet the minimum track length (four sequential views) are computed using a least-squares triangulation. At each update step, additional observations of a point are used to recompute a least-squares triangulation from all views.

In this algorithm, a strict upper limit is placed on the reprojection error of any scene point. A scene point with a reprojection error $e_r > 0.4$ pixels in any image is discarded from the estimate. This actively removes any scene point that is not accurately triangulated at the extreme depths indicative of this scenario, reducing the number of active tracks. As a consequence of these strict feature tracking routines we compensate by detecting and matching a high number of features per frame. We find that the culling routine actively removes more than 90% of features in each image, and only 30-40 are actively tracked frame-to-frame.

2.2.4 Bundle Adjustment

After each motion and structure update, a bundle-adjustment nonlinear optimisation is performed on the last five camera poses and observed scene. We use the analytical derivatives in the Jacobian calculation to improve optimisation speed and accuracy.

The SfM routine is then continuously repeated in a loop such that new poses are computed, new structure initialised and the estimate optimised via bundle adjustment to provide an updated and refined estimate in a sequential manner.

2.2.5 Place Recognition

After each pose update, openFAB-MAP generates loop closure hypotheses between the current and all previous images in the trajectory. The feature codebook and Chow-Liu Tree used by the algorithm are precomputed offline from seperate airborne vision data. In comparison to the SURF detector used in the original FAB-MAP [6], our algorithm uses the STAR detector (based on CenSurE [1]). This alternative detector produces more reliable loop-closure results on airborne data, where scenes have few unique features and are very self similar.

If openFABMAP determines a location probability for a frame greater than 99%, features are matched (similarly to Sec. 2.2.1) between the current and other images at that location. However, a minimum difference of 1000 frames is required to avoid naive matching against spatially close frames. If the ratio of matched feature inliers to the number of features in the current frame is greater than 15% the match is considered a positive loop closure and recorded for use in subsequent pose optimisation (Sec. 2.4).

2.3 Frame Striding

In contrast to other methods that often use a key-framing approach [17, 24] to discard images with small inter-frame movements, we use a frame-striding technique to actively skip images in the input stream. In the airborne scenario our algorithm uses a basis stride length of three frames.

By processing only every third frame, speed is significantly improved and frames where relative motion is small are actively avoided. In situations where the pose estimate between frames fails due to frame drops or rapid rolling/pitching of the aircraft, a recursive fallback is implemented to generate the next pose. When a failure to generate a pose estimate between frames i and $i+3$ is detected, a pose estimate between frames i and $i+2$ is attempted and so on until a reliable pose estimate is found, then returns to a three frame stride.

2.4 Pose Optimisation

The pose estimates computed from SfM and the constraints imposed by the detected loop closures can then be represented as a pose-graph and subsequently optimised using HOG-Man. All camera poses generated by the pose estimation routine are represented as nodes, with edges applied between sequentially adjacent poses. The loop-closure hypotheses generated by openFAB-MAP are used to apply additional edge constraints in the graph.

Similar to the method described in Sec. 2.2.2, a pose at time j matched to a 'base' pose i is re-computed from the structure observed by the camera at pose i. Any false-positive matches generated by openFAB-MAP are discarded at this point as they will not meet the required geometry test when generating a new camera pose.

In the pose-graph the loop-closure edge is generated by computing the 3D homography between the base camera and recomputed camera. These nodes and edge constraints are then input to the graph optimiser and processed in a sequential method to generate the optimised camera poses. As our graph only considers poses, we need to recover scene points from the optimised poses. All scene points are retriangulated via least-squares based on their original projections while ensuring that all meet the new epipolar constraints generated from the camera poses.

2.5 3D Scene Reconstruction

As a demonstration of the quality of the optimised solution, we generate a 3D reconstruction from the imagery and final pose estimate using a methodology described in [19]. Dense depth maps are generated to create oriented 3D points in a single consistent Euclidean space. A Poisson Surface estimation [12] is performed from this set of oriented points to generate a reconstruction of the environment and textured by projecting the coloured image data to the surface. In comparison to other reconstruction work which creates meshes from optimally selected, high resolution views, our method generates meshes from sequential, relatively low resolution images in addition to estimates of aircraft pose.

3 Experimental Setup

The flight platform is a 1/3 scale Piper Cub with a wingspan of 3.6m and fuselage length of 2.3m (Fig 2). It is capable of speeds of 30 to 110km/h with a maximum payload of 6kg.

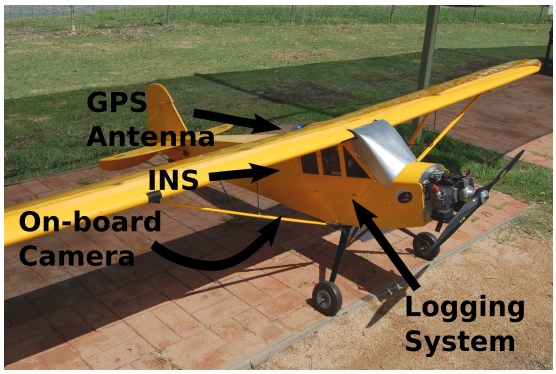

Fig. 2 The experimental platform, showing the location of logging system, camera, INS and GPS antenna.

The aircraft includes an off-the-shelf mini-ITX computer system running an Intel Atom processor (1.6GHz), with two 64GB solid-state drives in a RAID0 configuration. The sensor payload consists of a IEEE1394B colour Point Grey Flea 2 camera. The camera is downward facing towards the terrain in the fuselage of the platform, behind the engine and logging system (Fig. 2). A 6mm lens is used with a field of view of approximately $42° \times 32°$. The camera is calibrated before flight using a checkboard pattern and a modified version of the RADDOC Calibration toolbox [15].

3.1 Dataset

Data was collected over a 90 second portion of flight, at an altitude of 20-100m and a speed of $\sim 20m/s$. Bayer encoded colour images are logged at a resolution of 1280×960 pixels at 30Hz. Shutter time for each frame was set at 8.5µS to counteract motion blur. The area was rural farmland with relatively few trees, animals and buildings. Some difficulties in the dataset include rapid lighting transitions, and frame drops occur at semi regular intervals due to buffer overflows leading to difficulties in feature tracking. An XSens MTi-G INS/GPS system is used as the ground truth measurement system on the aircraft, with a manufacturer claimed positional accuracy of 2.5m CEP. Size and weight restrictions prevent the use of more accurate DGPS systems, however, the MTi-G itself provides a reasonably accurate estimate of pose over broad scales. The MTi-G unit is rigidly attached to the onboard camera, while the GPS receiver is installed directly above the camera. GPS, unfiltered IMU data and filtered INS pose were recorded at 120Hz from the XSens MTi-G.

4 Results

The algorithm was performed offline on the collected images to generate 879 camera poses. The dataset consisted of 2670 frames. Some key parameters for the processing include a stride length of 3 frames, 400 feature buckets, 10 features per bucket and a sliding window bundle adjustment of 5 frames.

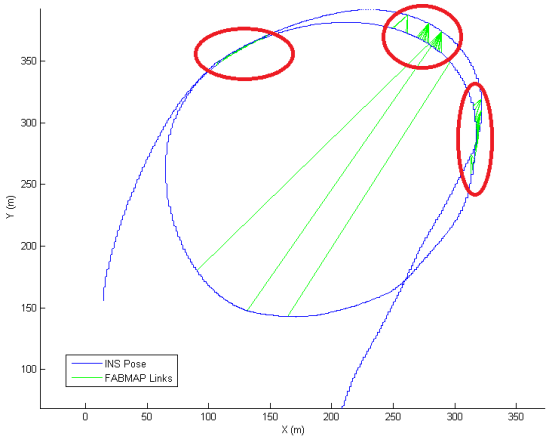

Fig. 3 Loop-closure events, highlighted in red, with probability $p > 0.99$ detected by openFAB-MAP overlaid on the ground truth GPS/INS pose. Some expected link locations are not observed due to differences in camera orientation at similar translational poses.

OpenFAB-MAP produced 91 loop closure events with $p > 0.99$, as seen in figure 3. Of these, 71 passed a minimum feature inlier count of 15% and the MLESAC camera resectioning routine, successfully removing all false positive events identified by openFABMAP, and hence used to generate an additional edge constraint.

The generated edges and poses were used by HOG-Man to produce an optimised SLAM estimate over the 879 poses.

The monocular pose results for both the raw SfM and optimised pose estimates were then converted to a metric scale by calculating the ratio of distances between two spatially distant ground truth poses and their corresponding reconstructed poses. This scale ratio is then applied via a homography to the reconstructed poses to achieve metricity. Both the raw and optimised poses are then registered to the ground truth in all 6 degrees of freedom [13] on the first 30 camera poses.

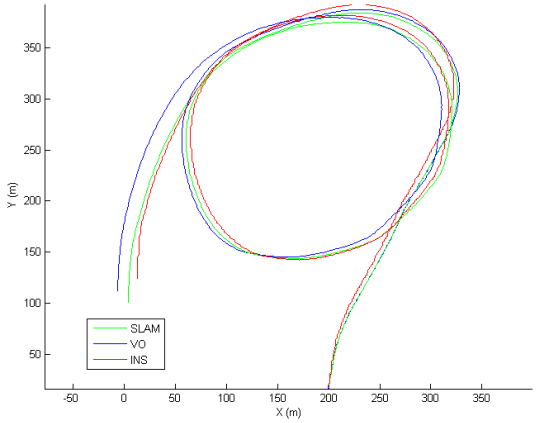

Fig. 4 Diagram in X, Y, showing SfM (VO) only path (blue), SLAM path (green) and INS path (red).

The results of the SfM only (VO) and optimised (SLAM) pose estimates are shown in figures 4 and 5. The SfM only estimate clearly drifts, and has a final pose error of 40.6m. The SLAM pose has significantly reduced error due to the optimisation, with a final pose error at the end of the trajectory of 27.2m. The length of the entire set of poses is 1.70km, meaning a translational drift of approximately 1.6% over the length of trajectory. This is consistent with the accumulated error in other presented works in ground scenarios [28, 17]. We speculate that some of the error is due to scale drift observable towards the end of the trajectory of both the raw and optimised estimates in Figure 5.

We also compare the orientation produced by both estimates to ground truth, shown in Figure 6. From this, it is clear that the algorithm is capable of accurately estimating orientation, with a maximum error of approximately 10.3° from the SfM only pose estimates, and a significantly reduced maximum error of 5.7° in the optimised estimate. The slightly positive pitch visible in Figure 6 is a result of the slightly backward facing orientation of the camera and INS rig in the aircraft.

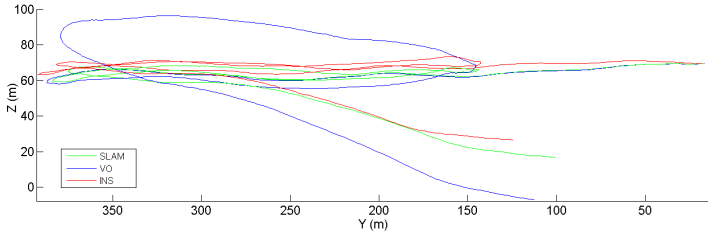

Fig. 5 Diagram in Y, Z, showing SfM (VO) only path (blue), SLAM path (green) and GPS/INS path (red).

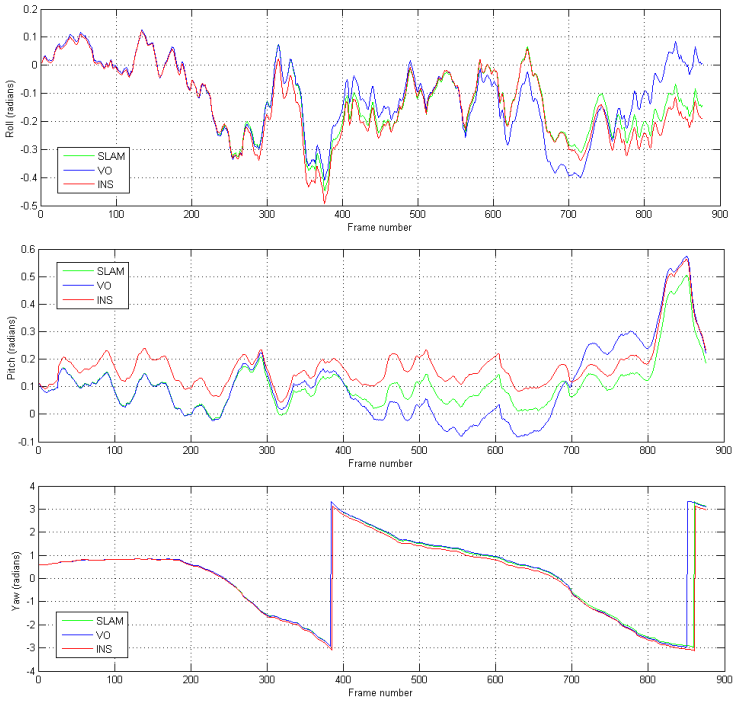

Fig. 6 Roll, pitch, yaw estimates for SfM (VO) only (blue), SLAM estimate (green) and GPS/INS (red), showing strong correlation.

4.1 Timing Results

The SfM algorithm, openFABMAP loop-closure detection and HOG-Man pose estimation were all performed using Windows 7 64-bit on an Intel Core i5 650 Processor at 3.2Ghz with NVIDIA Quadro 600 GPU and 16GB of system RAM. Aggressive memory management in the SfM algorithm meant that total memory consumption at the end of the sequence was 2.04 GB. In performing timing tests page-outs and

disk-writes were not included in the time estimates. From Table 1 it can be seen that the SfM algorithm is capable of performing at just over 3.1Hz if considered as a single frame stride (where every frame is processed). If we consider the 3 frame stride of this algorithm, the effective computed frame rate increases to just over 9.4Hz.

Table 1 Timing Results from Monocular SLAM algorithm

Process	Minimum (ms)	Average (ms)	Maximum (ms)	Note
Initialisation	725	-	-	Performed only once
Feature Detection	129	202	1294	
Feature Matching	65	77	196	
Pose Update	2	15	195	
Structure Update	1	9	62	
Bundle Adjustment	0	15	102	
Total SfM time per frame	197	318	1849	average fps: 3.14Hz, @ 3 frame stride: 9.43 Hz
openFABMAP	21	54	90	None
Loop Closure Matching	0	206	1857	Performed only on loop closure detection

From the computed poses and loop closure links, HOGMan produced an optimised result over the 879 poses in 2.1 seconds. While both the SfM algorithm and openFABMAP loop-closure detection were performed in a single thread, multi-threading the algorithm would lead to efficiency gains approaching online operation. We also anticipate that with strict memory management the algorithm is capable of performing similarly over much larger datasets.

4.2 Reconstruction

From the optimised pose estimates, 3D scene points were re-triangulated using their feature projections to reconstruct the optimised scene. In Figs. 1 and 7 we present reconstruction outputs generated from this optimised estimate. Figure 7 shows a 2D mosaic of the observed images projected to a ground plane estimated from the 3D scene features. This mosaic is compared to satellite imagery of the area for qualitative analysis. It should be noted that the mosaic is only computed from pose estimates of the camera and no explicit feature matching is performed to create the 2D reconstruction.

Figure 1 shows a subsection of the final 3D reconstruction. From this reconstruction 3D structure is readily apparent, showing buildings, a parked aircraft and trees on predominantly flat terrain. These results demonstrate the viability of our airborne SLAM algorithm in producing up-to-date, 2D and 3D textured maps of environments at high resolution with rapid turnaround.

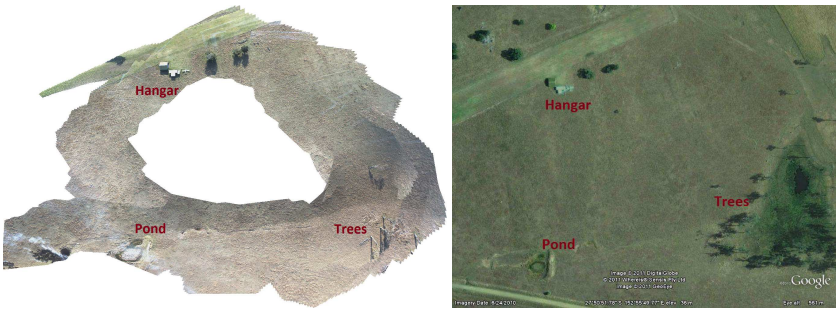

Fig. 7 Left: A densly reconstructed ground plane using only camera pose to inform map generation over 879 frames. Right: a comparison of the same area on Google Earth, showing qualitative accuracy of the final SLAM estimate.

5 CONCLUSIONS

We have successfully demonstrated that visual SLAM on a fixed-wing airborne robotic platform is capable of a high degree of accuracy without additional inputs. This demonstration shows capability for use in more complicated filtered algorithms and in conjunction with additional sensors in the air. In future we intend to apply this algorithm in real-time that will facilititate online navigation and mapping for airborne robotic vehicles. Additionally, we intend to demonstrate the algorithm using multi-camera rigs to increase accuracy, remove initialisation degeneracies and remove scale issues.

References

1. Agrawal, M., Konolige, K., Blas, M.: Censure: Center surround extremas for realtime feature detection and matching. International Conference on Computer Vision, The Proceedings of the pp. 102–115 (2008)
2. Blosch, M., Weiss, S., Scaramuzza, D., Siegwart, R.: Vision based mav navigation in unknown and unstructured environments. In: Robotics and Automation (ICRA), 2010 IEEE International Conference on, pp. 21–28. IEEE (2010)
3. Bryson, M., Johnson-Roberson, M., Sukkarieh, S.: Airborne smoothing and mapping using vision and inertial sensors. In: 2009 IEEE International Conference on Robotics and Automation, pp. 3143–3148. IEEE (2009)
4. Cherian, A., Andersh, J., Morellas, V., Papanikolopoulos, N., Mettler, B.: Autonomous altitude estimation of a UAV using a single onboard camera. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems pp. 3900–3905 (2009)
5. Clark, R., Lin, M., Taylor, C.: 3D environment capture from monocular video and inertial data. SPIE on Three-Dimensional Image Capture and Applications (2006)
6. Cummins, M., Newman, P.: FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. The International Journal of Robotics Research **27**(6), 647–665 (2008)
7. Eynard, D., Vasseur, P., Demonceaux, C., Frémont, V.: UAV altitude estimation by mixed stereoscopic vision. In: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, Iros, pp. 646–651. IEEE (2010)

8. Glover, A.J., Maddern, W.P., Milford, M.J., Wyeth, G.F.: FAB-MAP + RatSLAM : Appearance-based SLAM for Multiple Times of Day. In: The IEEE International Conference on Robotics and Automation, May, pp. 3507–3512. IEEE, Alaska, USA (2010)
9. Grisetti, G., Kuemmerle, R., Stachniss, C., Frese, U., Hertzberg, C.: Hierarchical optimization on manifolds for online 2d and 3d mapping. In: Robotics and Automation (ICRA), 2010 IEEE International Conference on. IEEE, Anchorage, USA (2010)
10. Haralick, B., Lee, C., Ottenberg, K., Nölle, M.: Review and analysis of solutions of the three point perspective pose estimation problem. International Journal of Computer Vision **13**(3), 331–356 (1994)
11. Hiep, V., Keriven, R., Labatut, P., Pons, J.P.: Towards high-resolution large-scale multi-view stereo. In: Conference on Computer Vision and Pattern Recognition (CVPR). Miami, USA (2009)
12. Hoppe, H.: Poisson surface reconstruction and its applications. Proceedings of the 2008 ACM symposium on Solid and physical modeling - SPM '08 p. 10 (2008)
13. Horn, B.K.P.: Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America A **4**(4), 629 (1987)
14. Kanade, T., Amidi, O., Ke, Q.: Real-time and 3D vision for autonomous small and micro air vehicles. 2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601) pp. 1655–1662 Vol.2 (2004)
15. Kassir, A., Peynot, T.: Reliable Automatic Camera-Laser Calibration. In: Australasian Conference on Robotics and Automation. Brisbane, Australia (2010)
16. Kim, J., Sukkarieh, S.: Autonomous airborne navigation in unknown terrain environments. IEEE Transactions on Aerospace and Electronic Systems **40**(3), 1031–1045 (2004)
17. Konolige, K., Agrawal, M.: Frameslam: From bundle adjustment to real-time visual mapping. IEEE Transactions on Robotics **24**(5), 1066–1077 (2008)
18. Lowe, D.: Object recognition from local scale-invariant features. In: International Conference on Computer Vision, The Proceedings of the, vol. pages, pp. 1150–1157 (1999)
19. McKinnon, D., He, H., Upcroft, B., Smith, R.N.: Towards automated and in-situ, near-real time 3-D reconstruction of coral reef environments. In: Oceans, September (2011)
20. Mei, C., Sibley, G., Newman, P.: A Constant-Time Efficient Stereo SLAM System. Systems Engineering pp. 1–11 (2009)
21. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 756–777 (2004)
22. Nistér, D., Naroditsky, O., Bergen, J.: Visual odometry for ground vehicle applications. Journal of Field Robotics **23**(1), 3–20 (2006)
23. Pollefeys, M., Nistér, D., Frahm, J.M., Akbarzadeh, a., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H.: Detailed Real-Time Urban 3D Reconstruction from Video. International Journal of Computer Vision **78**(2-3), 143–167 (2007)
24. Strasdat, H., Montiel, J., Davison, A.: Scale drift-aware large scale monocular SLAM. In: Proceedings of Robotics: Science and Systems (RSS). Citeseer, Zaragoza, Spain (2010)
25. Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Machine Vision and Applications (2011)
26. Torr, P., Fitzgibbon, A., Zisserman, A.: The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. International Journal of Computer Vision **32**(1), 27–44 (1999)
27. Volpe, J.: Vulnerability assessment of the transportation infrastructure relying on the global positioning system. Transportation (2001)
28. Warren, M., McKinnon, D., He, H., Upcroft, B.: Unaided stereo vision based pose estimation. In: Australasian Conference on Robotics and Automation, December. Brisbane, Australia (2010)