2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Visual Path Following on a Manifold in Unstructured Three-Dimensional Terrain

Paul Furgale* and Tim Barfoot†

University of Toronto Institute for Aerospace Studies
Toronto, Ontario, Canada, M3H 5T6
*paul.furgale@utoronto.ca, †tim.barfoot@utoronto.ca

***Abstract*— This paper describes the design and testing of a technique to enable long-range autonomous navigation using a stereo camera as the only sensor. During a learning phase, the rover is piloted along a route capturing stereo images. The images are processed into a manifold map of topologically-connected submaps that may be used for localization during an autonomous repeat traverse. Path following in non-planar terrain is handled by moving from localization in three dimensions, to path following in two dimensions using a local ground plane associated with each submap. The use of small submaps decouples the computational complexity of route repeating from the length of the path. We validate the algorithm by demonstrating its performance on a difficult three-dimensional route. Using this technique, a rover may autonomously traverse a multi-kilometer route in unstructured, three-dimensional terrain, without an accurate global reconstruction.**

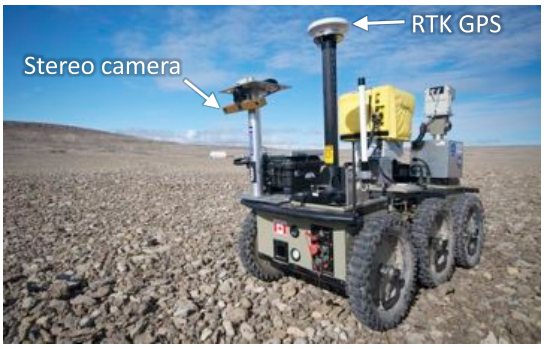

Fig. 1. The six-wheeled rover platform used in our visual path-following experiments. The stereo camera used for localization and mapping was a Point Grey Research Bumblebee XB3.

I. Introduction

In environments lacking a Global Positioning System (GPS) or equivalent, long-range autonomous navigation for rovers becomes a very difficult problem. Relative localization systems based on some combination of visual, inertial, and odometric sensing have become increasingly accurate. However, regardless of the level of accuracy, the error in the position estimate for any of these methods will grow without bound as the rover travels, unless periodic global corrections are made.

Manifold mapping has been used to enable autonomous robot mapping and localization in planar environments [1], [2], [3]. Embedding a robot's path in a higher-dimensional space allows for mapping without internal inconsistencies resulting from drift in motion estimates. Extending this idea to three-dimensional mapping results in a hybrid topological/metric representation [4].

We have developed a complete system for long-range, autonomous operation of a mobile robot (Figure 1) in outdoor, unstructured environments. This is achieved using *only* a stereo camera for sensing, a teach-and-repeat operational strategy, and a manifold map. During a learning phase—the *teach pass*—the rover is piloted over the desired route (either manually or using some external autonomous system), while the localization system builds a series of overlapping submaps. These submaps are then used for localization during the autonomous traversal phase—the *repeat pass*. The contribution of this paper is to outline a method suitable for long-range navigation in unstructured, three-dimensional terrain. Furthermore, we evaluate this method on a route with significant non-planar camera motion. While [5] evaluates the three-dimensional localization system in detail, this paper describes the hybrid topological/metric path representation we developed to bring path following on a manifold out of the plane and into unstructured, three-dimensional environments.

II. Related Works

In an early paper on vision-based map building, Brooks [6] outlined some basic principles for robotic mapping:

- The world is inherently three-dimensional. Localization and mapping should reflect this.
- Uncertainty in sensing will lead to maps that are globally inconsistent. However, to enable robot autonomy, maps only need to be locally consistent.

To deal with this, he proposed a map composed of freespace primitives in a graph. Similar in concept, [2] designed and implemented a multi-agent system that represented the robot's map as a manifold embedded in a higher-dimensional space. Manifold mapping changes the way a map represents the world. A map becomes topological in the sense that it defines a sequence of connected spaces, but the spaces in the map may have a many-to-one correspondence with the world. This topology is represented by dividing the map into a graph of submaps [7], [2], [3], or using a continuous relative representation [4]. Incremental errors that would cause inconsistencies in a purely metric map

disappear within the manifold representation. As a result, loop-closing decisions may be delayed [2] and loops may be closed in constant time, regardless of the size of the map [4]. Manifold mapping removes the constraint that maps be globally consistent, but in order to be useful for localization, the neighborhood around the robot must still appear locally Euclidean.

To see where this constraint expresses itself in the SLAM problem we examine the structure of the basic SLAM equations. The SLAM problem is formulated probabilistically as the task of estimating the joint posterior density of the map, $\mathbf{m}$, and vehicle state at time k, $\mathbf{v}_k$, given all previous measurements, $\mathbf{z}_{0:k}$, control inputs, $\mathbf{u}_{0:k}$, and prior knowledge, $\mathbf{x}_0$ [8]:

$$p(\mathbf{x}_k, \mathbf{m} | \mathbf{z}_{0:k}, \mathbf{u}_{0:k}, \mathbf{x}_0) \tag{1}$$

Most solutions to this problem involve computing $p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{m})$, the likelihood of the measurement vector, $\mathbf{z}_k$, given the current state and map estimates. The likelihood is then expressed using an observation model, $\mathbf{h}(\cdot)$, such that

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{m}) + \mathbf{v}_k, \tag{2}$$

where $\mathbf{v}_k$ is observation noise. The properties of (2) determine the form of the constraint. Most navigation sensors discern something about the geometry in the robot's local neighborhood and, for a map to be useful, the neighborhood must appear Euclidean to the sensor suite. Any deviation must be small enough to hide in $\mathbf{v}_k$. This is the motivation behind the adaptive window selection in [4], and the choice of submap size in [3]. If this constraint is satisfied, the map is still useful for localization, even if the global reconstruction is very inaccurate.

Visual teach-and-repeat navigation systems have been built on this very concept, using topologically-connected keyframes and a path-tracking system that attempts to drive the robot to the same viewpoints along the path. These algorithms may be classified as *appearance-based*, which correlate large portions of the input image with the keyframes [9][10], and *feature-based*, which track sparse feature points found in the keyframes [11][12][13]. Appearance-based systems generally require planar camera motion, so only feature-based algorithms are suitable for localization in three-dimensional environments. Royer et al. [11] use a global bundle adjustment to create a consistent map whereas similar path-following performance is reported by Šegvić et al. [12] who use only local reconstructions and a simple visual servoing scheme. These results suggest that large-scale autonomy is possible without a globally-consistent reconstruction. However, none of the teach-and-repeat algorithms published to date have been tested in unstructured, highly three-dimensional environments.

III. System Description

We have developed a complete system for mapping and localization using a stereo camera as the only sensor. The major processing blocks of our system are depicted in Figure 2.

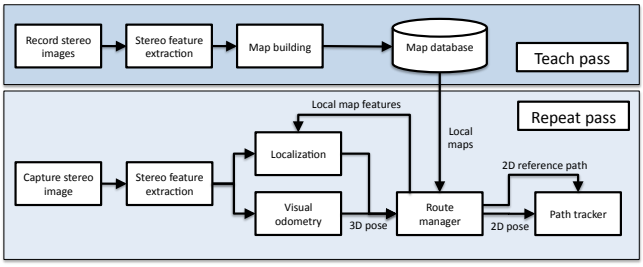

Fig. 2. An overview of the major processing blocks in our system.

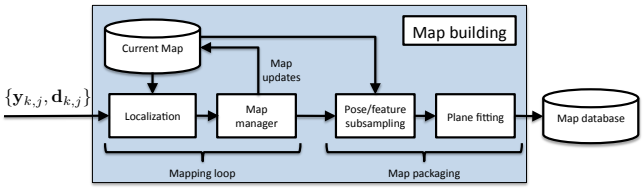

Fig. 3. An overview of the mapping process.

The map frame $\underset{\rightarrow}{\mathcal{F}}_m$ is the frame in which three-dimensional estimation occurs. We define $\underset{\rightarrow}{\mathcal{F}}_{c_k}$ to be a coordinate frame attached to the left camera of a stereo pair at time k. The attitude of the camera at this time may be described by $\mathbf{C}_{m,c_k}$, the rotation matrix that transforms vectors from $\underset{\rightarrow}{\mathcal{F}}_{c_k}$ to $\underset{\rightarrow}{\mathcal{F}}_m$. Similarly, we define the camera's position as $\rho_m^{c_k,m}$, a vector from the origin of $\underset{\rightarrow}{\mathcal{F}}_m$ to the origin of $\underset{\rightarrow}{\mathcal{F}}_{c_k}$ (denoted by the superscript), and expressed in $\underset{\rightarrow}{\mathcal{F}}_m$ (denoted by the subscript). Together, $\mathbf{C}_{m,c_k}$ and $\rho_m^{c_k,m}$ define the camera's *pose* in $\underset{\rightarrow}{\mathcal{F}}_m$.

The localization system used in this work is fully described and evaluated in [5]. Each stereo pair is processed using a sparse stereo pipeline based on our own implementation of the Speeded Up Robust Features (SURF) algorithm [14]. For each keypoint j found at time k, the pipeline returns the stereo image coordinates, $\mathbf{y}_{k,j}$, 64-dimensional SURF descriptor, $\mathbf{d}_{k,j}$, and the three-dimensional position of the feature with respect to $\underset{\rightarrow}{\mathcal{F}}_{c_k}$, $\mathbf{p}_{c_k}^{j,c_k}$. For every timestep k, the localization algorithm returns the pose of the camera in $\underset{\rightarrow}{\mathcal{F}}_m$.

A. Route Learning

The route learning process is shown in Figure 3. A localization loop incrementally builds a sequence of submaps, each composed of a reference path, three-dimensional feature points, and associated SURF descriptors. When the reference path reaches a maximum length (5 meters for all experiments), the submap is saved to disk and packaged for use in the repeat pass.

First, the poses in the reference path are subsampled to satisfy a minimum-spacing constraint. This smoothes the path and puts it in a format suitable for our path tracker. All experiments in this paper use a 0.5 m spacing. Features that were never tracked (i.e., seen by one frame but never seen by another) are removed from the map.

The subsampled reference poses give the path of the camera in $\underset{\rightarrow}{\mathcal{F}}_m$, but our path tracker controls the position

of the vehicle, not the camera. As the global reconstruction is inconsistent, we must determine a suitable frame in which to perform path following.

We define a frame, $\underrightarrow{\mathscr{F}}_{v_k}$ attached to the vehicle. At time k, we may compute the vehicle position using the rotation and translation between the camera and vehicle frames: $\mathbf{C}_{c_k,v_k}$ and $\rho_{v_k}^{c_k,v_k}$. The reference path of the vehicle, $\rho_m^{v_k,m}$, is

$$\rho_m^{v_k,m} := \left(\rho_m^{c_k,m} - \mathbf{C}_{m,c_k}\mathbf{C}_{c_k,v_k}\rho_{v_k}^{c_k,v_k}\right) \ , \tag{3}$$

and its attitude is

$$\mathbf{C}_{v_k,m} = \mathbf{C}_{c_k,v_k}^T\mathbf{C}_{c_k,m} \ . \tag{4}$$

The projection from three dimensions to two is determined by fitting a plane to the map's features. Each feature, $\mathbf{q}_m^{i,m}$, has passed an outlier rejection step in the localization loop; to the camera, the reconstruction is locally Euclidean. For each feature i, at position $\mathbf{q}_m^{i,m}$ in the map, we find d_i, the minimum distance between the feature and one of the vehicle reference poses:

$$d_i := \min_k \left\| \rho_m^{v_k,m} - \mathbf{q}_m^{i,m} \right\|$$

From this distance, we compute a weight, w_i, used in the plane fitting:

$$w_i = \begin{cases} \frac{1}{d_i+\sigma_p} & \text{if } d_i \le \tau_d \\ 0 & \text{otherwise} \end{cases}$$

The threshold, τ_d, ensures that distant features outside of the vehicle corridor are not used for the plane fit, and σ_d controls the maximum possible weight. For all experiments in this paper, we use $\sigma_d = 0.01$, and $\tau_d = 1.5$. We parameterize the plane by a unit vector, $\mathbf{n}$, and offset, b, such that any point $\mathbf{x}$ on the plane satisfies

$$\mathbf{n}^T\mathbf{x} + b = 0 \ .$$

From this equation, we define a weighted least-squares problem to solve for $\mathbf{n}$ and b by minimizing J_p:

$$J_p := \frac{1}{2}\sum_{i=1}^{N} w_i(\mathbf{n}^T\mathbf{q}_m^{i,m} + b)^2 - \frac{1}{2}\lambda\left(\mathbf{n}^T\mathbf{n} - 1\right) \ , \tag{5}$$

where N is the number of features in the map and λ is a Lagrange multiplier that ensures $\mathbf{n}$ is a unit vector. Solving for the minimum of this equation results in the eigenproblem

$$\mathbf{A}\mathbf{n}^\star = -\lambda\mathbf{n}^\star \ ,$$

where

$$W := \sum_{i=1}^{N} w_i \ ,$$

$$\mathbf{A} := \sum_{i=1}^{N} w_i\left(\mathbf{q}_m^{i,m}\right)\left(\mathbf{q}_m^{i,m}\right)^T - \frac{1}{W}\left(\sum_{i=1}^{N} w_i\mathbf{q}_m^{i,m}\right)\left(\sum_{i=1}^{N} w_i\mathbf{q}_m^{i,m}\right)^T \ ,$$

and $\mathbf{n}^\star$, the unit vector that minimizes J_p, is the eigenvector of $\mathbf{A}$ corresponding to its minimum eigenvalue. Figure 4 illustrates this process, showing the camera and vehicle poses, the weighted sparse feature points, and the resulting plane fit.

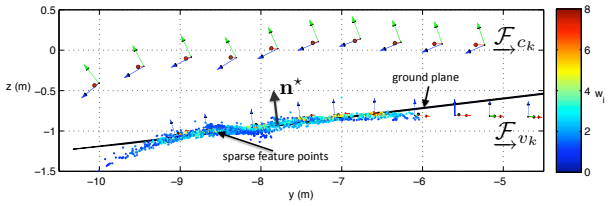

Fig. 4. Side view of a single submap showing the camera frames, $\underrightarrow{\mathscr{F}}_{c_k}$, the vehicle frames, $\underrightarrow{\mathscr{F}}_{v_k}$, the sparse feature points, and the ground plane fit to the features.

The unit vector $\mathbf{n}^\star$ is the normal of the xy-plane of the projection frame, $\underrightarrow{\mathscr{F}}_p$, expressed in $\underrightarrow{\mathscr{F}}_m$. We now calculate the rotation, $\mathbf{C}_{m,p}$, that transforms vectors from $\underrightarrow{\mathscr{F}}_p$ to $\underrightarrow{\mathscr{F}}_m$. Using the shorthand $c_a := \cos(a)$ and $s_b := \sin(b)$, the rotation $\mathbf{C}_{m,p}$ may be parameterized by Euler angles, (α,β,γ), such that

$$\mathbf{C}_{m,p} = \begin{bmatrix} c_\alpha c_\beta & s_\alpha c_\beta & -s_\beta \\ c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & -s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & c_\beta s_\gamma \\ c_\alpha s_\beta c_\gamma + s_\alpha c_\gamma & -s_\alpha s_\beta c_\gamma - c_\alpha s_\gamma & c_\beta c_\gamma \end{bmatrix} \ . \tag{6}$$

We know that $\mathbf{n}^\star$ expressed in $\underrightarrow{\mathscr{F}}_p$ is $\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$, which leads to the following constraint:

$$\mathbf{n}^\star = \mathbf{C}_{m,p}\begin{bmatrix}0\\0\\1\end{bmatrix} = \begin{bmatrix}-s_\beta\\c_\beta s_\gamma\\c_\beta c_\gamma\end{bmatrix}$$

Defining the components of $\mathbf{n}^\star =: \begin{bmatrix}n_1 & n_2 & n_3\end{bmatrix}^T$, we can solve for β and γ:

$$\beta = \text{asin}(-n_1) \tag{7}$$

$$\gamma = \text{atan2}(c_\beta n_2, c_\beta n_3) \tag{8}$$

The last Euler angle, α, is ambiguous (the plane normal is only a two-degree-of-freedom constraint) so we introduce a final constraint that the x-axis of $\underrightarrow{\mathscr{F}}_{v_0}$ lies in the xz-plane of $\underrightarrow{\mathscr{F}}_p$. Using $\mathbf{C}_{m,p}$ and the vehicle path from (3) and (4), we can transform the reference path to the projection frame:

$$\rho_m^{v_k,v_0} = \rho_m^{v_k,m} + \mathbf{C}_{r_0,v_0}\rho_{v_0}^{r_0,v_0} \tag{9}$$

$$\rho_p^{v_k,v_0} = \mathbf{C}_{m,p}^T\rho_m^{v_k,v_0} \tag{10}$$

$$\mathbf{C}_{v_k,p} = \mathbf{C}_{v_k,m}\mathbf{C}_{m,p} \tag{11}$$

At this point, the map is saved to disk with the following information:

- a vehicle reference path with L poses (indexed by ℓ), $\{\rho_p^{\ell,p}\}$, expressed in $\underrightarrow{\mathscr{F}}_p$, calculated from (3), (9), and (10)
- a rotation $\mathbf{C}_{p,m}$ that defines the projection to a local ground plane, calculated from (7), (8), and (6)
- a set of N features (indexed by i), each with position $\mathbf{q}_m^{i,m}$ and SURF descriptor $\mathbf{v}_i$

After saving the map to disk, older poses and features are removed from the database in memory. We build the submaps to overlap by 50% as the common data between maps smooths out the transitions [3]. Figure 5 shows a short section of a map database, the ground plane of each submap,

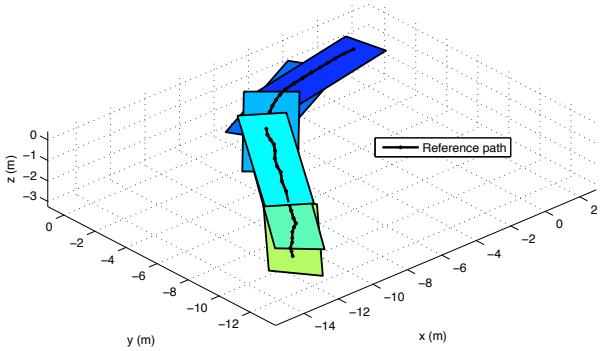

Fig. 5. A view of six overlapping submaps with the reference path plotted above.

and the reference path. When the teach pass is complete, a database of maps is available for use in the repeat pass.

Each submap is between 500 kilobytes and 2 megabytes depending on the number of features tracked (which is scene dependent). This size includes extra data that is used solely for algorithm evaluation and not to repeat the route. Averaged over all teach passes, this amounts to 348 megabytes per kilometer. The teach pass processes an image approximately every 0.2 meters, 5000 images per kilometer. An appearance-based approach using the rectified stereo images would occupy 2.9 gigabytes per kilometer and saving all of the keypoints and descriptors would take up 1.3 gigabytes per kilometer (assuming 500 stereo keypoints per frame). By aggregating data, our system offers a significant savings in storage over a pure appearance-based approach.

B. Route Repeating

Using the map database described in the previous section, the rover is able to repeat a learned route in either direction any number of times, provided the camera is facing the same direction as it was when the route was learned. Neither direction switching during path following nor local obstacle detection have been implemented, although both should be possible [3].

Only a single map is loaded into memory at a time, and the localization module interleaves *relative localization* (visual odometry), and *global localization* against the current map. The localization block estimates the camera's position, $\rho_m^{c_k,m}$, and attitude, $\mathbf{C}_{c_k,m}$. Equations (3), (9), and (10) are then used to produce $\rho_p^{v_k,p}$, the position of the vehicle in the projection frame. The attitude of the vehicle in the projection frame, $\mathbf{C}_{p,v_k}$ is computed using (11), then decomposed into a yaw-pitch-roll Euler-angle sequence. The yaw value of this sequence is the vehicle's heading in the projection frame, θ_k. Defining the components, $\rho_p^{v_k,p} =: [x_k \;\; y_k \;\; z_k]^T$, we can express the two-dimensional robot pose, $\rho_k = [x_k \;\; y_k \;\; \theta_k]^T$ This planar pose of the robot and the projected reference path are passed to a unicycle-model version of the planar path-tracking algorithm described by [3].

This planar pose estimate feeds in to a route management system that triggers map handoffs, and monitors the route-following system for errors. The route manager tracks the closest point on the current reference path. When the vehicle reaches the middle of a reference path, a map handoff is triggered. This involves the following steps:

- Loading the next map from disk
- Updating the feature database used for localization
- Updating the reference path used by the path tracker
- Updating the transformation from $\underrightarrow{\mathscr{F}}_m$ to $\underrightarrow{\mathscr{F}}_p$

The topologically-connected sequence of small, overlapping submaps, enables long-range path following despite an inaccurate global reconstruction. Figure 6 shows a plot of the recorded GPS and reconstructed path for a route approximately 5 kilometers long. The two paths have been aligned at the start but the reconstruction quickly diverges, flying a kilometer into the air. However, when the rover is traveling along the path, only one of this route's 1732 submaps is loaded into memory. The rover localizes against the features in the map and attempts to repeat the same path within a local ground plane. As the images at the bottom of Figure 6 show, to the robot, there is no inconsistency as the map is locally Euclidean.

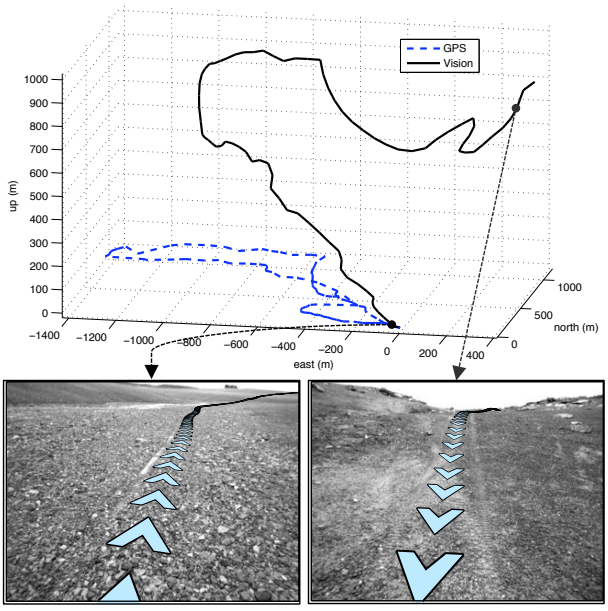

Fig. 6. The visual reconstruction of a five kilometer rover traverse plotted against GPS (Top). Although the reconstruction is wildly inaccurate at this scale, locally it is good enough to enable retracing of the route. The bottom images show views from either end of the path, with the reference path plotted as a series of chevrons. To the rover, the map is locally Euclidean.

IV. System Evaluation

We have tested our algorithm in the urban environment surrounding the University of Toronto Institute for Aerospace Studies (UTIAS) and in planetary analog terrain on Devon Island, near the Haughton-Mars Project in the Canadian High Arctic. Out of 32.919 kilometers traveled, only 0.128 kilometers were piloted manually, an autonomy rate of 99.6%. The path-following experiments are described more fully in [5]. The evaluation in this paper focuses on showing that our manifold map enables path following in non-planar

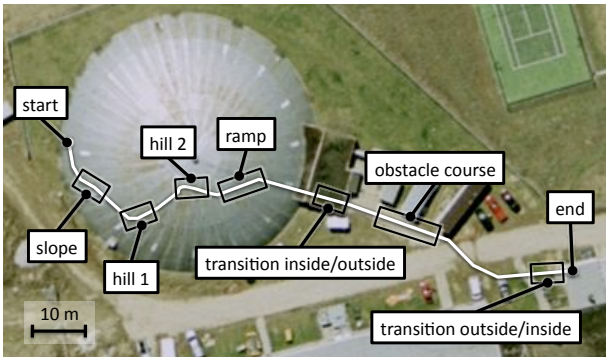

Fig. 7. An overhead view of the route built to test non-planar camera motion.

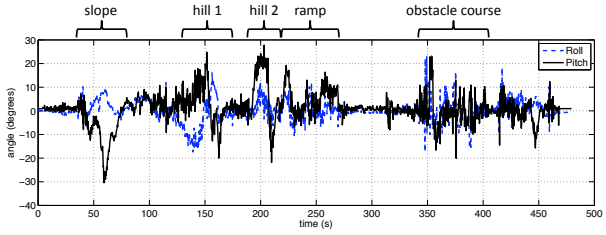

Fig. 8. The pitch and roll of the rover during the first teach pass of this route.

environments. To test this, we built a route at UTIAS where the rover experienced large three-dimensional motion. An overhead view of the route is shown in Figure 7 and the pitch and roll of the camera (as measured by an inclinometer) is shown in Figure 8.

The rover started inside our indoor test facility on a raised platform. It descended a slope, climbed two hills, ascended a ramp, and then drove through a narrow corridor leading outdoors. There, it traversed an obstacle course, crossed the road, and finished the route by parking in our laboratory. The platform experienced pitch and roll up to 27° and moved from an indoor, low-light environment to outdoors and back. The video attachment to this paper shows the rover driving this route as well the rover's view.

We taught this route twice, once during development of the obstacles, and once after they were complete. The routes were repeated 7 and 5 times, respectively. Every repeat pass was successful, despite the three-dimensional motion of the camera. Figure 9 shows the teach pass corridor (the track of the teach pass laterally extended ± 2 meters for illustration) with the tracks of the repeat passes overlaid. Sections where the algorithm experienced global localization dropouts are highlighted in blue. Figure 10 shows some sequences from the repeat pass that convey the magnitude of the camera motion on this route.

The localization had the most trouble during the steep hills and at the end of the route. The steep hills were constructed of gravel that moved as the rover drove over them, changing the appearance of the route every time it was repeated. In these cases, the relative localization would carry the algorithm through sections with appearance changes. Descending

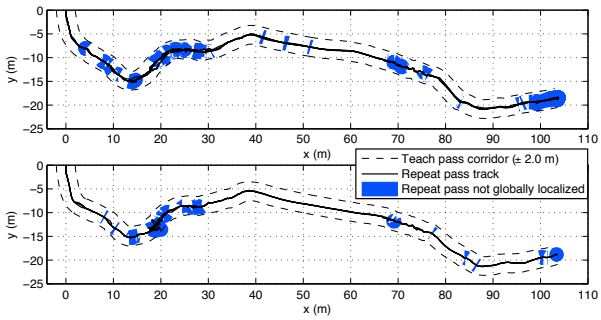

Fig. 9. Results from the two experiments run on a three-dimensional route. The top plot is one teach pass and seven repeat passes made while building the route. The bottom plot is one teach pass and five repeat passes done after the route was complete. All repeat passes were completed fully autonomously despite significant non-planar camera motion.

the slope of each hill also caused significant motion blur in the low-light environment. Motion-blurred images caused failures in both the relative and global localization systems. At the bottom of the slope, the rover would successfully relocalize against the current submap, correct for the motion of the rover during the motion-blurred frames, and continue along the route.

This experiment was performed before our field trials on Devon Island to prove that the teach-and-repeat system would work on three-dimensional terrain. During our field trials we tested the algorithm over many three-dimensional routes. In all cases, three-dimensional motion of the camera was not a limiting factor for route following.

V. Conclusion

We have designed and tested a stereo teach-and-repeat navigation system capable of dealing with non-planar camera motion. To achieve this, we have used concepts from manifold mapping to show how to move from three-dimensional localization, to two-dimensional path following. Furthermore, we show that the computational complexity of route following can be decoupled from the path length by using a sequence of small, topologically-connected submaps. Through this technique, a rover may autonomously traverse a multi-kilometer route in unstructured, three-dimensional terrain, without an accurate global reconstruction. We have tested this algorithm on a short route designed to cause extremely non-planar camera motion. The route was taught twice and repeated 12 times. All repeat passes were completed fully autonomously, despite difficult lighting conditions and extreme camera motion.

Future work may involve implementing a better reconstruction method such as the one described in [4]. Furthermore, we would like to integrate our mapping and localization system with an autonomous terrain assessment/path-planning algorithm capable of detecting changes in the environment, planning around them, and finding the previously taught route.

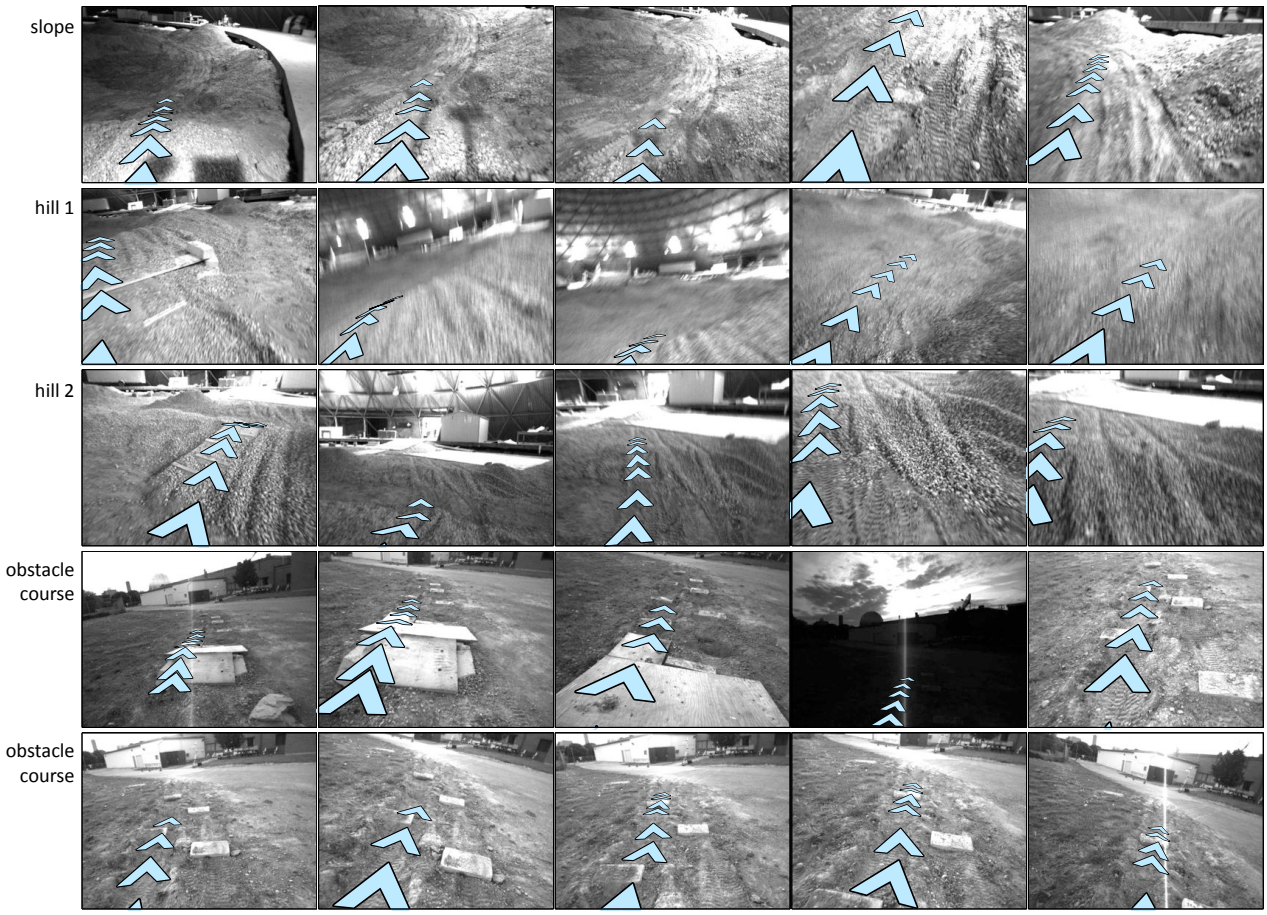

Fig. 10. Image sequences (left to right) taken from a repeat pass corresponding to some of the features in Figure 7 that show our algorithm performing despite very non-planar camera motion. The path, plotted as chevrons, confirms that localization is indeed done in three-dimensions.

VI. Acknowledgements

Funding for our field trials was provided by The Canadian Space Agency's Canadian Analogue Research Network (CARN) program and the Natural Sciences and Engineering Research Council of Canada (NSERC) funded the remaining work. Thanks to Pascal Lee and the entire Mars Institute/Haughton Mars Project team for support during our field trials on Devon Island.

References

[1] A. Howard, L. E. Parker, and G. S. Sukhatme, "Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection," *The International Journal of Robotics Research*, vol. 25, no. 5–6, pp. 431–447, 2006.

[2] A. Howard, G. S. Sukhatme, and M. J. Mataric, "Multirobot simultaneous localization and mapping using manifold representations," in *Proceedings of the IEEE*, vol. 94, no. 7, 2006, pp. 1360–1369.

[3] J. Marshall, T. Barfoot, and J. Larsson, "Autonomous underground tramming for center-articulated vehicles," *J. Field Robot.*, vol. 25, no. 6-7, pp. 400–421, 2008.

[4] G. Sibley, C. Mei, I. Reid, and P. Newman, "Adaptive relative bundle adjustment," in *Robotics Science and Systems (RSS)*, Seattle, USA, June 2009.

[5] P. Furgale and T. Barfoot, "Stereo mapping and localization for long-range path following on rough terrain," in *IEEE Intl. Conf. on Robotics and Automation, ICRA*, Anchorage, Alaska, USA, May 2010.

[6] R. Brooks, "Visual map making for a mobile robot," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, Al, Ed., vol. 2, 1985, pp. 824–829.

[7] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous localization and map building in large-scale cyclic environments using the atlas framework," *The International Journal of Robotics Research*, vol. 23, no. 12, pp. 1113–1139, 2004.

[8] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, June 2006.

[9] Y. Matsumoto, M. Inaba, and H. Inoue, "Visual navigation using view-sequenced route representation," *1996 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 83–88 vol.1, April 1996.

[10] A. M. Zhang and L. Kleeman, "Robust appearance based visual route following for navigation in large-scale outdoor environments," *Int. J. Rob. Res.*, vol. 28, no. 3, pp. 331–356, 2009.

[11] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *Int. J. Comput. Vision*, vol. 74, no. 3, pp. 237–260, 2007.

[12] S. Šegvić, A. Remazeilles, A. Diosi, and F. Chaumette, "A mapping and localization framework for scalable appearance-based navigation," *Comput. Vis. Image Underst.*, vol. 113, no. 2, pp. 172–187, 2009.

[13] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Omnidirectional vision based topological navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.

[14] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, 2008.