2012 IEEE International Conference on Robotics and Automation
RiverCentre, Saint Paul, Minnesota, USA
May 14-18, 2012

Visual Teach and Repeat Using Appearance-Based Lidar

Colin McManus[1], Paul Furgale[2], Braden Stenning[3], and Timothy D. Barfoot[3]

Abstract— **Visual Teach and Repeat (VT&R) has proven to be an effective method to allow a vehicle to autonomously repeat any previously driven route without the need for a global positioning system. One of the major challenges for a method that relies on visual input to recognize previously visited places is lighting change, as this can make the appearance of a scene look drastically different. For this reason, passive sensors, such as cameras, are not ideal for outdoor environments with inconsistent/inadequate light. However, camera-based systems have been very successful for localization and mapping in outdoor, unstructured terrain, which can be largely attributed to the use of sparse, appearance-based computer vision techniques. Thus, in an effort to achieve lighting invariance and to continue to exploit the heritage of the appearance-based vision techniques traditionally used with cameras, this paper presents the first VT&R system that uses appearance-based techniques with laser scanners for motion estimation. The system has been field tested in a planetary analogue environment for an entire diurnal cycle, covering more than 11km with an autonomy rate of 99.7% of the distance traveled.**

Fig. 1. An image of our field robot during a 1.1km autonomous repeat traverse at the Ethier Sand and Gravel pit in Sudbury, Ontario, Canada.

I. INTRODUCTION

Visual Teach and Repeat (VT&R) is a technique that can allow a robot to repeat any previously driven route fully autonomously, requiring only an onboard visual sensor (see Figure 1). This makes VT&R extremely well suited for a number of tasks where GPS is either not available or unreliable; examples include sample-and-return missions on foreign planets [1], autonomous underground tramming for mining [2], and autonomous convoying [3], to name a few.

VT&R is a two-phase technique comprised of a *teach pass* and a *repeat pass*. During the teach pass, the system uses a visual sensor to construct a map of the environment, which can either be represented as metric [4], topological [5], or a hybrid of the two [2]. During the repeat pass, the system localizes against the archived maps, and in some cases also performs relative motion estimation [1], in order to accurately retrace the previously driven route.

Metric map representations are seldom used in VT&R systems, but they have been successfully applied in both indoor [4], [6] and outdoor settings [7]. Topological map representations are far more common and generally store images at the vertices of a graph of topologically-connected places. Topological VT&R can be thought of as a succession of visual homing steps, where the goal is to match the current image with the closest archived image, in a topological sense. Matsumoto et al. [5] developed a system that would archive a collection of camera images during the teaching phase and would match against these images using a cross-correlation procedure during the repeating; they referred to this collection of images as a *view-sequenced route representation*, which formed the basis for a number of similar systems [8], [9], [10]. As metric information is not available, most of these techniques used bearing-only control laws [11], [12], [13] or visual servoing [14], [15], [16] for control; however, many of these systems generally display loose bounds on the accuracy of the robot's lateral error. Owing to this fact, hybrid topological/metric maps [17] appear to offer the best of both worlds, as they avoid the costly construction of a globally consistent map by representing local maps as nodes in a graph and benefit from accurate metric information for control.

There have been many successful VT&R systems that have used hybrid topological/metric map representations, ranging from an autonomous underground mining system [2] that used a SICK laser as the primary sensor, to a planar, outdoor system that used an omnidirectional camera [18]. However, the most relevant work comes from Furgale and Barfoot [1], who were the first to develop a fully 3D VT&R system for outdoor, unstructured environments and validated the system in the Canadian High Arctic. The route teaching involved capturing and logging stereo images for post-processing into a series of locally consistent overlapping maps, where 3D landmarks were embedded within each local map. For localization, their system would interleave visual odometry

[1] Mobile Robotics Group, University of Oxford, Oxford, England; colin@robots.ox.ac.uk
[2] Autonomous Systems Lab, ETH Zürich, Zürich, Switzerland; paul.furgale@mavt.ethz.ch
[3] Autonomous Space Robotics Lab, University of Toronto Institute for Aerospace Studies, Toronto, Canada; {tim.barfoot,braden.stenning}@utoronto.ca
*Work carried out while in the Autonomous Space Robotics Lab, University of Toronto Institute for Aerospace Studies.

(VO) with localization against the map, which was done for computational reasons as they were unable to perform both within the same control cycle. The system was field tested on Devon Island, and of the 32.9km traveled, 99.6% was traversed autonomously. However, one of the main failure modes that was encountered with this stereo-based system was change in ambient lighting, which made previously visited locations appear very different in some situations.

This failure mode highlights a very important issue, which is that all passive camera-based systems are dependent on ambient lighting conditions; this is a problem in outdoor environments that lack adequate levels of light, such as the Moon, or consistent levels of light, such as on Earth. Unlike cameras, active sensors such as light detection and ranging (lidar) sensors, use their own light source to illuminate the scene, making them a favourable alternative in light-denied environments. What differentiates our work from the numerous examples of lidar-based systems that use the classic iterative closest point (ICP) scan matching algorithm [19], [20], [21], is that we render 2D lidar intensity images from the raw data, in order to apply the same sparse appearance-based computer vision techniques that have been successfully used with camera-based VT&R systems (a notable exception being May et al. [22]). These lidar intensity images look very similar to a standard grayscale camera image, but with the added benefit of looking the same in the light and in the dark (see Figure 2 for an example of a camera/lidar intensity image). Combined with the azimuth, elevation, and range data, a lidar provides all the necessary appearance and metric information required for motion estimation. The appearance-based lidar techniques used in this system are based on the method described by McManus et al. [23], who used a laser scanner for appearance-based VO.

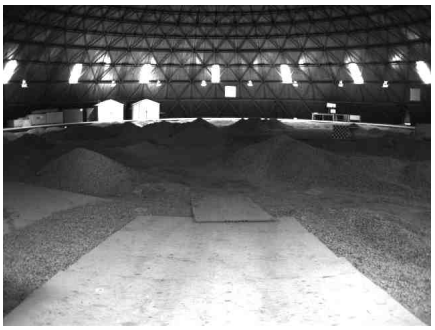
This paper will detail the design and implementation of a lighting-invariant VT&R system that uses appearance-based lidar for long-range, over-the-horizon navigation in outdoor, unstructured environments. This system has been validated and tested in a planetary analogue environment, autonomously repeating over 11km in a variety of lighting conditions with an autonomy rate of 99.7% of the distance traveled. The system only relies on frame-to-frame VO using sparse bundle adjustment, but is able to build maps online and repeat routes accurately and consistently with root-mean-squared path errors on the order of centimetres.

Like Furgale and Barfoot [1], the VT&R system we present falls under the topological/metric map category, since the map is a topological network of keyframes that contain metric information; more specifically, we construct a *pose graph* [24]. What differentiates the system described in this paper with the one from Furgale and Barfoot is the following: (i) local maps are represented as augmented keyframes instead of locally consistent submaps, (ii) the map building occurs online as opposed to offline, (iii) VO is performed in image space as opposed to Euclidean space, and (iv) both VO and localizing against the map are performed in the same control cycle.

We will begin with a brief overview of our appearance-based lidar technique in Section II, which covers topics such as image formation, keypoint generation, and our estimation framework. Section III will present our efficient, sensor-generic VT&R framework, where we will introduce our novel *sliding local map* approach for improved matching robustness. Section IV will describe the hardware used in our outdoor, long-range experiments and present the results from teaching a route during daylight outdoors and autonomously repeating the same route every 2-3 hours for over 25 hours. Section V provides a discussion of our results, mainly focussing on the various off-nominal situations that arose and how the system coped with these situations. Lastly, Section VI concludes with some future work that still remains.

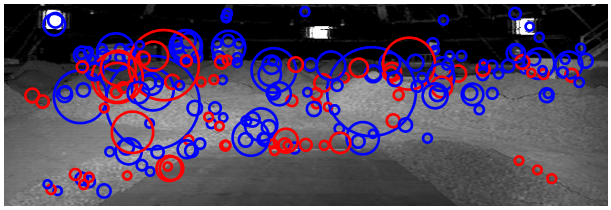
II. APPEARANCE-BASED LIDAR

This section will provide a brief overview of our appearance-based lidar technique, which involves three main steps: (i) image formation, where the raw lidar data are processed into a stack of intensity, azimuth, elevation, and range images, (ii) keypoint generation, to create metric keypoints for motion estimation, and (iii) our back-end estimation framework.



(a) Camera intensity image.

(b) Processed lidar intensity image.



(c) Processed lidar intensity image with SURF keypoints shown. Red circles represent dark-on-light patches and blue circles represent light-on-dark patches.

Fig. 2. Camera intensity image and processed lidar intensity image of the same scene. Image sizes have been adjusted to correspond to their field of view. Camera intensity image, 52.5^oV $\times$ 70^oH field of view (FOV), 512×384 pixels, 15Hz framerate. Autonosys intensity image, 30^oV$\times90^o$H FOV, 480×360 pixels, 2Hz framerate.

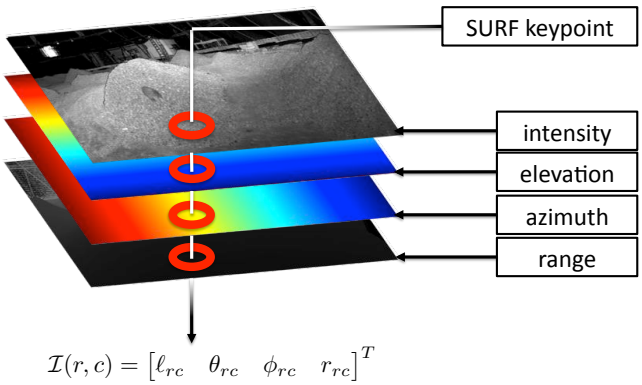


Fig. 3. The image stack generated from the raw laser-rangefinder data. SURF keypoints are found in image space at sub-pixel locations and bilinear interpolation is used to find the azimuth, elevation, and range of the keypoint. Linearized error propagation from image space to azimuth/elevation/range is then used to determine the uncertainty of the measurement.

A. Image Formation

The first step in image formation is to develop a camera model, which requires knowledge of the specific sensor being used. The lidar used in our experiments is an Autonosys LVC0702 lidar that provides approximately equally spaced azimuth and elevation samples, making a spherical camera model an intuitive choice. Due to the fact that most objects in a natural environment are not very reflective, the raw intensity image is extremely dark and requires image processing for use in a feature detector. The approach by McManus et al. [23] preprocessed raw lidar images by applying adaptive histogram equalization and a Gaussian low-pass filter. Although this technique was shown to be successful, it failed to take into account the coupling between intensity and range. Theoretically, squared range corrections should be applied [25], [26], [27], but we found that this darkened the image in the near field, which is not ideal for VO, since most of the tracked features are on the ground. Instead, we found that a linear range correction (i.e., multiplying the intensity values by their associated range) and rescaling the brightness values into the $[0, 255]$ range proved to work well (distant features become more visible, which are useful for orientation information). Figure 2 shows a camera image and a processed lidar intensity image of the same scene.

After processing the intensity image, the associated azimuth, elevation, and range data are assembled into an array in the exact same order as the intensity image, forming an *image stack*. This concept will prove useful in the next section, which introduces how keypoint measurements and their associated uncertainties are generated.

B. Keypoint Generation

An image stack, $\mathcal{I}$, is composed of an intensity image, $\mathcal{I}_\ell$, an azimuth image, $\mathcal{I}_\theta$, an elevation image, $\mathcal{I}_\phi$, and a range image, $\mathcal{I}_r$, which can be evaluated at any integer row, r, and column, c, as, $\boldsymbol{\mathcal{I}}_{rc}$, a 4×1 column,

$$\boldsymbol{\mathcal{I}}_{rc} := \mathcal{I}(r,c) = \begin{bmatrix} \ell_{rc} & \theta_{rc} & \phi_{rc} & r_{rc} \end{bmatrix}^T,$$

where ℓ_{rc}, θ_{rc}, ϕ_{rc}, and r_{rc} are the scalar intensity, azimuth, elevation, and range stored at this location in the image stack. For simplicity, we assume that the elements of each image are independent, identically-distributed samples such that

$$\begin{aligned} \boldsymbol{\mathcal{I}}_{rc} &= \overline{\boldsymbol{\mathcal{I}}}_{rc} + \delta\boldsymbol{\mathcal{I}}_{rc}, \quad \delta\boldsymbol{\mathcal{I}}_{rc} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \\ \mathbf{R} &:= \text{diag}\left\{\sigma_\ell^2, \sigma_\theta^2, \sigma_\phi^2, \sigma_r^2\right\}, \end{aligned}$$

where $\overline{\boldsymbol{\mathcal{I}}}_{rc}$ is the true value, $\delta\boldsymbol{\mathcal{I}}_{rc}$ is zero-mean Gaussian noise, and the components of $\mathbf{R}$ are based on the properties of the sensor (e.g., taken from the datasheet or derived empirically).

Using a GPU implementation of the SURF algorithm, keypoint detection in the intensity image, $\mathcal{I}_\ell$, returns a list of image locations, $\mathbf{y}_i = \begin{bmatrix} u_i & v_i \end{bmatrix}^T$, with associated covariances, $\mathbf{Y}_i$, where u_i, and v_i are generally not integers (see Figure 2(c)). An azimuth, elevation, and range measurement, $\mathbf{z}_i$, is computed via bilinear interpolation of $\mathcal{I}_\theta$, $\mathcal{I}_\phi$, and $\mathcal{I}_r$, according to $\mathbf{z}_i = \boldsymbol{\mathcal{B}}(\mathbf{y}_i, \mathcal{I}_\theta, \mathcal{I}_\phi, \mathcal{I}_r)$. The uncertainty, $\mathbf{Q}_i$, associated with $\mathbf{z}_i$, is produced by propagation of $\mathbf{R}$ and $\mathbf{Y}_i$ through the interpolation equations, such that $\mathbf{Q}_i := \mathbf{J}_i \mathbf{Y}_i \mathbf{J}_i^T + \mathbf{R}$, where $\mathbf{J}_i = \left.\frac{\partial \boldsymbol{\mathcal{B}}}{\partial \mathbf{y}}\right|_{\mathbf{y}_i}$ (see Figure 3 for an illustration of this image stack concept).

C. Bundle Adjustment

We use the keypoints generated from our image stack to compute the reprojection error, which is the standard error term in bundle adjustment. Each measurement, $\mathbf{z}_{k,j}$, corresponds to an observation of landmark j at time k. The error term, $\mathbf{e}_{k,j}$, is given by

$$\mathbf{e}_{k,j} := \mathbf{z}_{k,j} - \mathbf{g}\left(\mathbf{x}_{0,k}, \mathbf{p}_0^{j0}\right),$$

where $\mathbf{x}_{0,k}$ is a column of state variables for the camera pose at time k and expressed in the base frame, $\mathbf{p}_0^{j0}$ is a column of state variables for landmark j expressed in the base frame, and $\mathbf{g}(\cdot)$ is our sensor model. We use 3-point RANSAC [28] for outlier rejection and then proceed with a sliding-window bundle-adjustment approach to accomplish VO, where we optimize over all landmark positions between two adjacent frames, $\mathbf{p}_0^{j0}$, as well as the current state estimate, $\mathbf{x}_{0,k}$, but leave the previous state, $\mathbf{x}_{0,k-1}$ fixed. In addition, to bound the estimate within a local neighbourhood of its previous location, we include a no-motion prior on the current state, $\mathbf{x}_{0,k}$. It should be noted that in each control cycle, our system performs frame-to-frame VO to update its estimate in the map and then matches against the closest keyframe in the map. In both cases, we use the same bundle adjustment framework.

III. SYSTEM OVERVIEW

This section provides a detailed overview of the VT&R system, combining the concepts and methods shown in the previous section. A description of the online mapping process during the teach pass will be covered, as well as the dynamic local map construction used during the repeat pass.

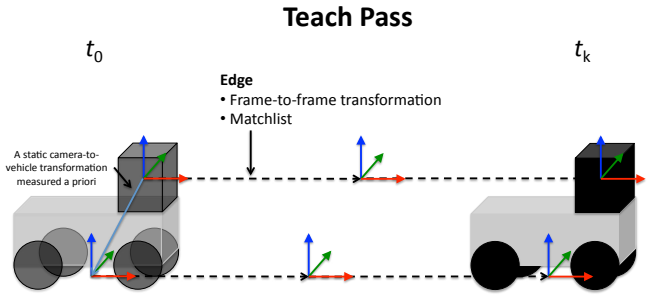


Fig. 4. The taught path is built as a pose graph consisting of relative frame transformations between poses. Vertices in the graph store keyframes containing keypoints (e.g., azimuth, elevation, range), SURF descriptors, camera calibration/geometry information, and timestamps. Edges store relative frame transformations and lists of inter-frame keypoint matches. New vertices are added to the graph when the robot travels a certain distance or when it rotates by a certain amount. Once the taught path is constructed, the path is transformed into the vehicle reference frame using a camera-to-vehicle transformation.

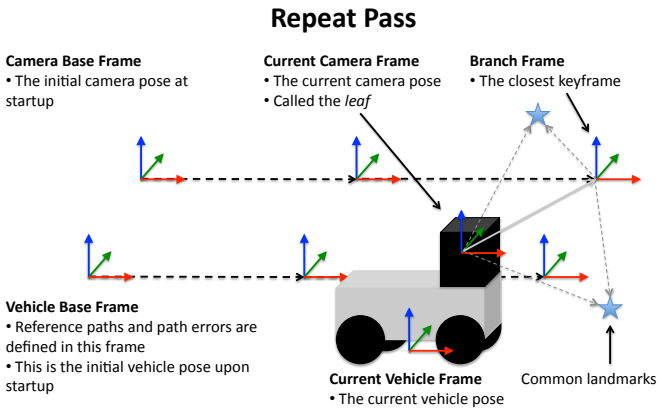


Fig. 5. During the repeat pass, images from the current sensor frame, called the *leaf*, are used for a frame-to-frame VO estimate and then matched against the nearest keyframe from the teach pass, called the *branch*.

A. Teach Pass

During the teach pass, the system builds a topologically connected network of keyframes, which is either added to or begins the creation of a *pose graph* [24]. For each keyframe, the following information is stored:

- *Keypoints and Descriptors* - A list of keypoints, $\mathbf{z}_i$, associated uncertainties, $\mathbf{Q}_i$, and associated 64-element SURF descriptors, as described in Section II-B,
- *Camera Calibration/Geometry Information* - Sensor-specific camera geometry used to convert a keypoint, $\mathbf{z}_i$, to a Euclidean landmark, $\mathbf{p}_i$, and vice versa,
- *Timestamps* - Used for synchronization,
- *Images* - Used purely for visualization of feature tracks.

Frame-to-frame transformation matrices and their uncertainties are stored along the edges that connect two keyframes in the pose graph, as well as a *matchlist*, which specifies the post-RANSAC matching keypoints between the two frames.

B. Repeat Pass

When repeating a route, the system acquires the appropriate chain of relative transformations from the pose graph (in the order that is specified) and constructs the taught route in the vehicle base frame. Since the route was constructed in the frame of the camera, we transform the path into the vehicle base frame according to the following:

$$\mathbf{T}_{v_0,v_k} = \mathbf{T}_{v,c}\mathbf{T}_{c_0,c_k}\mathbf{T}_{v,c}^{-1},$$

where $\mathbf{T}_{v,c}$ is the camera-to-vehicle transformation, which is a fixed transformation that is measured a priori and $\mathbf{T}_{c_0,c_k}$ is the transformation from frame k to frame 0 as seen in the camera base frame for this particular path. Once the path has been built in the vehicle frame, at each timestep, the system performs the following steps for localization (see Figure 5).

1) **Frame-to-frame VO** - This provides an incremental pose update to achieve a good guess for the next step of localizing against the nearest keyframe. Keypoint matching is done using a nearest neighbour approach in descriptor space and the bundle adjustment formulation discussed in Section II-C is used for frame-to-frame VO. It should be noted that the estimation takes place in the nearest keyframe's reference frame, called the *branch*. This makes the approach completely relative, as the estimation is never performed in a fixed global reference frame.
2) **Localization against the map** - The system localizes against the nearest keyframe on the pose graph (nearest in a Euclidean sense), using the keypoint matching and outlier rejection methods discussed earlier. This provides a relative transformation estimate, $\mathbf{T}_{c_b,c_k}$, between the current camera pose at time k, called the *leaf*, and the nearest keyframe, called the *branch*. As is done for frame-to-frame VO, the estimation is done in the branch reference frame. After matching against the map, the new estimate is transformed into the vehicle frame for the path tracker according to: $\mathbf{T}_{v_0,v_k} = \mathbf{T}_{v,c}\mathbf{T}_{c_0,c_b}\mathbf{T}_{c_b,c_k}\mathbf{T}_{v,c}^{-1}$, where $\mathbf{T}_{v,c}$ is the fixed camera-to-vehicle transformation, $\mathbf{T}_{c_0,c_b}$ is the transformation from the current branch to the camera base frame, and $\mathbf{T}_{c_b,c_k}$ is the newly updated leaf-to-branch transformation.

C. The Sliding Local Map

This keyframe-to-keyframe matching is clearly less costly than a multi-frame bundle adjustment method, but it does give up accuracy to a multi-frame approach because it only considers the nearest keyframe. During preliminary testing, it was discovered that simple keyframe-to-keyframe matching was not robust enough to large movements and the algorithm would often fail to localize against the map. Inspired by the continuous relative representation of Sibley et al. [24], we addressed this problem by introducing a *sliding local map*, which attempts to embed the nearest keyframe with additional information from the surrounding keyframes. Note that this approach results in a map that is *locally consistent*,

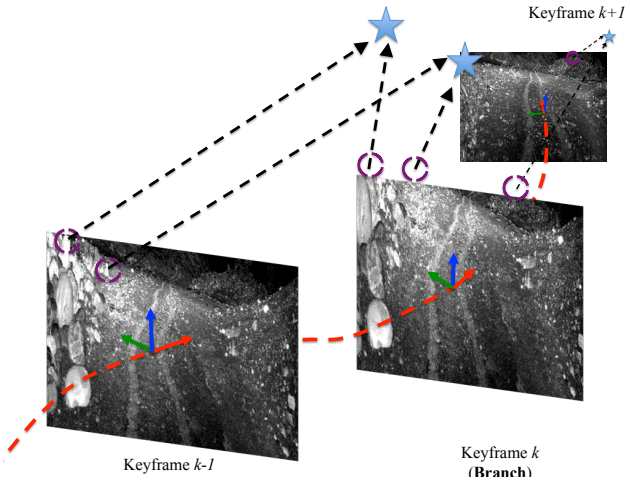


Fig. 6. An illustration of the local map construction, where the nearest keyframe, called the *branch*, is embedded with keypoints from surrounding keyframes. Including additional keypoints in this manner increases map matching due to the non-identical teach and repeat trajectories that lead to slight viewpoint changes.

which is enough for route repeating. We construct this sliding local map as follows (see Figure 6 for an illustration).

1) Pick a window of keyframes surrounding the nearest keyframe at timestep k.
2) For each common landmark between keyframes $k-i$ and $k-i+1$ in the set of all matches, M_i, compute the keypoint measurement in branch keyframe using the sensor model:

$$\forall m \in M_i, \text{compute} \quad \mathbf{z}_{k,m} = \mathbf{g}\left(\mathbf{x}_{k-i+1,k}, \mathbf{p}_{k-i+1}^{m,k-i+1}\right),$$

where we note that the relative transformations from $k-i+1$ to k are all available from the pose graph. These additional keypoints are added to the branch in order to include additional information in the local map.
3) After the local map has been built up from all of the surrounding keyframes, finding keypoint matches, rejecting outliers, and computing the corrected pose of the vehicle follows the exact same procedure as outlined in Section II.

It should be noted that all of this occurs online during the repeat pass.

D. Off-Nominal Scenarios

There are numerous off-nominal modes that can occur while repeating, which include failing to localize against the map and/or frame-to-frame VO failures. These can occur when there is sufficient motion blur, large path deviations leading to viewpoint changes, scene changes (e.g., due to rain moistening the ground), or lost image packets. In order to be robust to such failure modes, the system responds in the following ways. If localizing against the map is unsuccessful, the system will continue to move and use frame-to-frame VO for up to 3m. Afterwards, if still unable to match against the map, the system will stop and enter a search mode where the current image is matched against a series of images in the database around the latest branch estimate. This search mode will continue until it exhaustively searches all the images in the database. A successful match against the map occurs if more than 10 keypoints are matched 5 times. Once a successful match has been determined, the localization estimate is updated and the system resumes following.

IV. EXPERIMENTS

This section presents long-range VT&R field tests with a high-framerate Autonosys lidar. All tests were conducted at the Ethier Sand and Gravel pit in Sudbury, Ontario, Canada, as it was a suitable planetary analogue environment due to its lack of vegetation and sandy/rocky terrain (see Figure 1). In total, over 11km of autonomous driving was achieved and post-processed DGPS was used for groundtruth. The experiment involved manually teaching a 1.1km route outdoors at approximately 7:45 pm in sunlight and autonomously repeating that route every 2-3 hours for 25 hours. What will follow is a description of the hardware used in this 25 hour experiment, followed by the experimental results.

A. Hardware

The mobile-platform used in these experiments was a six wheeled, skid steered vehicle that has an articulated chassis with three individual pods, where the fore and aft pods can pitch and roll relative to the middle pod. The vehicle was equipped with a Thales DG-16 Differential GPS unit, an Autonosys LVC0702 lidar, and two Macbook Pro computers (one used to interface with the Autonosys to port-forward data packets and the other for all of the lidar processing and control). In addition to the onboard DGPS, another DGPS was set up as a static base station to allow for real-time kinematic corrections (for groundtruthing purposes only). The Circular Error Probability[1] (CEP) for these differential GPS units is 40cm. An image of the field robot equipped with its sensors is shown in Figure 7.

The Autonosys LVC0702 is a high-framerate amplitude modulated continuous wave (AMCW) lidar that combines a nodding and hexagonal mirror to scan a 45^oV/90^oH FOV with a pulse repetition rate (PRR) of 500,000 points/second [29]. The LVC0702 provides 15-bit intensity information, has a maximum range of approximately 53.5m and can scan as fast as 10Hz; however, increasing the frame rate results in lower image resolutions. For these experiments, the vertical field of view of the sensor was reduced from 45^o to 30^o at $(0^o, -30^o)$ in order to capture 480x360 images at 2Hz and increase the resolution in the vertical field of view.

B. Field Trials

A 1154m route was taught during sunlit conditions around 7:45 pm and autonomously repeated every 2-3 hours for a total of 10 runs, covering over 11km. It should be stressed that the lighting varied from full daylight to full darkness over the course of this experiment and the system was always matching to the full daylight conditions (i.e., the

[1]CEP is defined as the radius of a circle where 50% the data will fall.

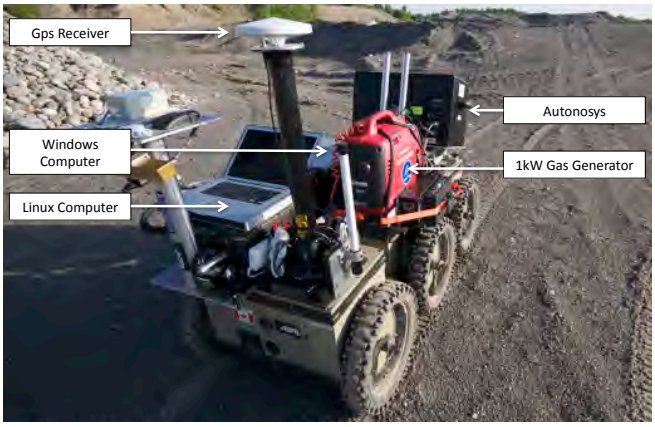


Fig. 7. ROC6 field robot and its sensor configuration. The robot is equipped with the high-framerate Autonosys lidar at the front, a GPS receiver at the rear, a 1kW gas generator, and two laptop computers (the Windows computer is directly connected to the Autonosys and port-forwards raw data data to the Linux computer, which performs the localization in ROS).

teach pass). The route was taught to resemble a realistic exploration mission, traversing to a number of dead-ends and thus requiring backtracking to explore new areas.

Figure 8 shows a GPS plot of both the teach pass (sunlit conditions) and the first repeat pass (complete darkness), a keypoint matching plot showing frame-to-frame VO matches and map matches, and the lateral tracking error for the entire traverse. For this run, the root-mean-squared (RMS) lateral error was 8.2cm and the route was completed fully autonomously without human intervention. Table I shows the repeat pass number, the start and end time, and the percentage of distance traveled autonomously for each run. Note that repeat pass 3 is not listed due to a software and hardware malfunction that occurred early on in the run. It should be stressed that this was not a failure of the algorithm, but rather a failure in the implementation, which is why it is not included in the table.

We are reporting autonomy rates instead of RMS path errors for each run due to a number of limitations in our groundtruth. Firstly, there was a discrepancy in the measured difference between the repeat pass runs and the teach pass run due to the fact that different satellites were observed in each run because of the long periods of time between the various trials. Secondly, our DGPS has a CEP of 40cm, which is actually quite large compared to the level of accuracy of the VT&R technique; this is especially true since the rover was always moving so there was no GPS averaging. Thirdly, due to physical constraints on the platform, the GPS receiver was mounted on the opposite end of the robot from the actual sensor, meaning that the estimated lateral error and the measured lateral error could have been different depending on the orientation of the rover pods[2]. Lastly, we encountered GPS dropouts in a number of runs, which is why we have chosen to focus on the autonomy rates instead of GPS-measured lateral error. It should be kept in mind that the system would not have been able to complete the run fully autonomously if the lateral errors were not under a metre as we are unable to match against the map beyond this point.

[2]Onboard inclinometer data was available, but was extremely noisy and not used in processing these results.

TABLE I

AUTONOMY RATES FOR ALL REPEAT PASS RUNS.

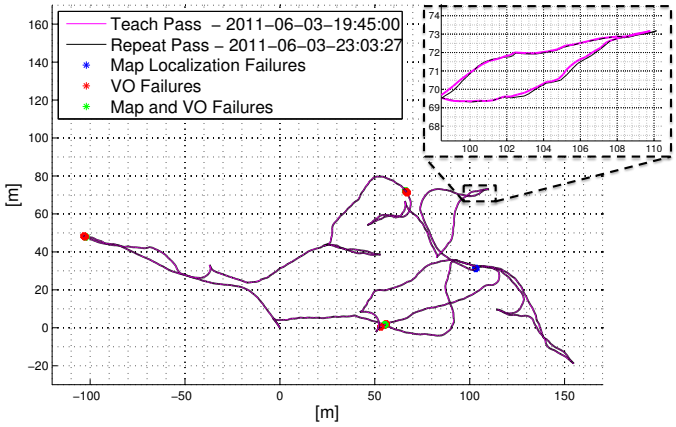
Repeat Pass No.	Start Time (hh:mm:ss)	End Time (hh:mm:ss)	Distance Covered Autonomously (%)
1	23:03:27	00:03:39	100
2	01:26:53	02:50:34	99.85
4	05:00:28	05:56:26	99.91
5	09:47:12	10:57:13	99.48
6	11:51:36	13:20:19	98.49
7	14:15:54	15:35:51	99.46
8	16:25:05	17:32:41	100
9	18:24:19	19:18:41	100
10	20:31:06	21:37:36	100
11	22:58:43	23:50:06	100

V. DISCUSSION

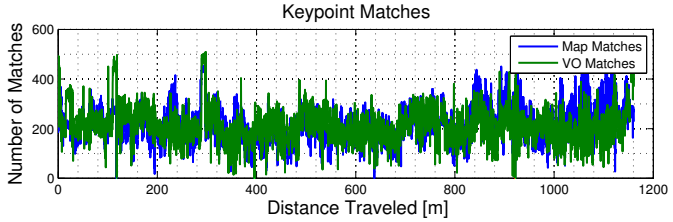
The main discussion points on which we wish to focus are the total number of off-nominal modes encountered during our field trials and how our system responded. It should be noted that after repeat run 1 and prior to repeat run 5, it had rained quite heavily and changed the reflective properties of the soil. This proved to be an interesting test of the system's robustness to slight environmental changes.

Referring to Figure 9, it is interesting to note that repeat run 1 had the highest number of VO/map matches, followed by a small dip near noon and then approaching a relatively constant value for the rest of the runs. It was expected that map matches would drop over time as the environment changed; however, it is difficult to explain why VO matches dropped, since VO is based on matching current and previous frames. One possible explanation could be due to the heavy rainfall, which changed the reflectivity of the soil and resulted in less texture overall (towards hour 19, it was noticed that the ground had dried significantly). The dip during noon could also be explained by the fact that the sun was very direct and strong during this time, which could affect the lidar as it must filter incoming light [23].

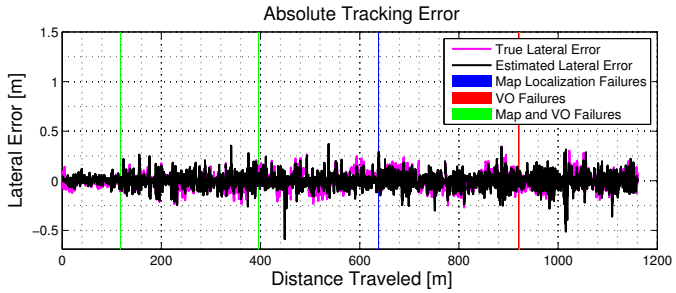
Figure 10 shows the various off-nominal modes that occurred as distance traveled versus the time of day from the teach run. We wish to stress that most of these off-nominal cases did not require manual interventions, as the system was able to recover using the strategies discussed in Section III. Interestingly, the highest number of simultaneous VO and map failures occurs at the 12-hour mark, with the number of map failures following a similar trend. Note that the large map failure peak at the 23-hour mark was caused by a software issue during the early portion of the run. As this software issue was not a failure of our algorithm, we have discounted these map failures as indicated by the dashed lines. Thus, ignoring this software issue, and noting the drop-off in the number of matches around the 12-hour mark (see Figure 9), the results confirm what was observed by McManus et al. [23], which is that the lowest number of



(a) GPS tracks of the teach pass and repeat pass with various off-nominal modes plotted in different colours. A zoomed-in portion demonstrates the centimeter-level accuracy of our system.



(b) Keypoint matches for frame-to-frame VO and matching against the map.



(c) Lateral error measured by GPS and estimated lateral error from the localization system versus distanced traveled.

Fig. 8. Repeat pass 1 results. Top image: GPS tracks of the teach pass and repeat pass. Middle image: keypoint matches over the entire traverse. Bottom image: measured and estimated lateral errors over the entire traverse.

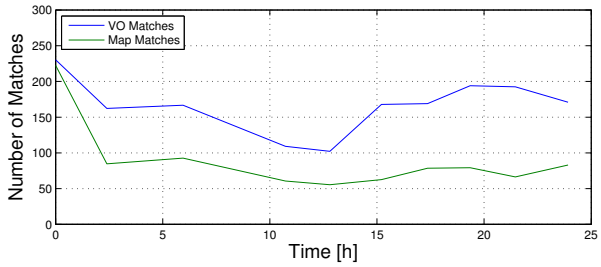


Fig. 9. Average number of VO matches and map matches per repeat run. Map matches drop to their lowest when matching roughly 12 hours apart due to the lidar's slight sensitivity to ambient lighting.

matches occur between lidar intensity images separated by 12 hours. Again, this is not a surprising result given the fact that lidar sensors are somewhat affected by ambient lighting conditions; however, even in the worst case (run 6), the system was able to repeat the taught route with an autonomy rate of 98.49%, which is a feat that could not be accomplished with a passive sensor under the dramatic lighting changes studied here.

Most of the cases that required human intervention occurred when the system failed to localize against the map and was unable to recover. As discussed in Section III, when the system fails to localize against the map, it will use VO until it reaches a distance threshold of 3m. After reaching this threshold, the system will stop the vehicle and begin its search of the map. This recovery method worked well with the stereo-based system by Furgale and Barfoot [1] since their VO was reasonably accurate up to 50m, allowing the system to traverse past feature-poor areas. However, in the case of this system, metric VO is very inaccurate, even over short distances (e.g., 5-10m). This is the result of a strong assumption that was silently used in Section II-A.

More specifically, the assumption that was used is that when forming the image stack, all of the pixels in the image arrive at the same instant in time, which is of course false, since the laser is continually scanning while moving. This means that range values from one part of the image arrive later than others, creating a distorted view of the scene that results in a biased estimate [30]. In theory, since the timestamp of each laser reading is known, a time correction could be applied to compensate for this distortion, but this is the focus of future work. As demonstrated, long-range, accurate VO is not necessary for VT&R, which is one of the major strengths of the technique. However, the lack of accurate VO means that the system is less robust to map localization failures since VO will not be able to accurately guide the vehicle forward in hopes of relocalizing against the map. This was indeed the case for the regions where manual control was needed, as the inaccurate VO was unable to successfully bring the vehicle beyond these feature-poor regions. Clearly, applying motion compensation for better VO would be a significant improvement to the system's robustness and is the subject of ongoing work.

VI. CONCLUSION

This paper has detailed the design, implementation, and testing of a lighting-invariant Visual Teach and Repeat system that combines fast and effective appearance-based computer vision techniques with a state-of-the-art high-framerate lidar sensor. The main purpose of this research was to design a method that would enable long-range autonomous retro-traverses for planetary sample and return missions; however, there are many other applications for this technique that extend beyond the space domain (e.g., patrolling, underground mining, and convoying). By using lidar as the primary sensor, this system is able to avoid one of the more challenging aspects of visual perception in outdoor environments: dynamic lighting conditions, which proved to be a limiting factor for

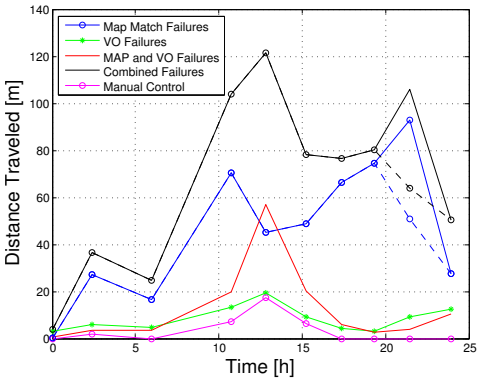


Fig. 10. Total number of failures measured by distance traveled versus the time since the teach pass. A dashed line has also been drawn to indicate the number of map match failures for run 10, discounting a software bug that caused localization failures in the second dead-end. The black line is simply the sum of all the failures, indicating that matching images roughly 12 hours apart yields the worst repeating performance. It is important to note that most of these map matching or VO failures recovered fully autonomously. The magenta line represents distance covered manually.

Furgale and Barfoot [1]. Through long-range field tests in a planetary analogue environment, the system's robustness and overall effectiveness was demonstrated on over 11km of travel, 99.7% of which was traversed fully autonomously.

VII. ACKNOWLEDGMENTS

The authors would like to extend their deepest thanks to the staff of the Ethier Sand in Gravel Pit in Sudbury, Ontario, Canada for allowing us to conduct our field tests on their grounds. We also wish to thank Dr. James O'Neill from Autonosys for his help in preparing the sensor for our field tests. In addition, we would also like to acknowledge Andrew Lambert from the Autonomous Space Robotics Lab (ASRL) for his help in preparing the GPS payload, Chi Hay Tong from ASRL for his work on the GPU SURF algorithm, Goran Basic from ASRL for designing and assembling the Autonosys payload mount, and Hang Dong from ASRL for his help with the Autonosys field testing. Lastly, we also wish to thank NSERC and the Canada Foundation for Innovation, DRDC Suffield, the Canadian Space Agency, and MDA Space Missions for providing us with the financial support necessary to conduct our experiments.

REFERENCES

[1] P. Furgale and T. Barfoot, "Visual teach and repeat for long-range rover autonomy," *Journal of Field Robotics, special issue on "Visual mapping and navigation outdoors"*, vol. 27, no. 5, pp. 534–560, 2010.

[2] J. Marshall, T. Barfoot, and J. Larsson, "Autonomous underground tramming for center-articulated vehicles," *Journal of Field Robotics*, vol. 25(6-7), pp. 400–421, 2008.

[3] A. Richardson and M. Rodgers, "Vision-based semi-autonomous outdoor robot system to reduce soldier workload," in *Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE) 4364*, Orlando, Florida, United States, April 16 2001, pp. 12–18.

[4] E. Baumgartner and S. Skaar, "An autonomous vision-based mobile robot," *IEEE Trans. on Auto. Control*, vol. 39(3), pp. 493–502, 1994.

[5] Y. Matsumoto, M. Inaba, and H. Inoue, "Visual navigation using view-sequenced route representation," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, vol. 1, 1996, pp. 83–88.

[6] K. Kidono, J. Miura, and Y. Shirai, "Autonomous visual navigation of a mobile robot using a human-guided experience," *Robotics and Autonomous Systems*, vol. 40(2-3), pp. 124–132, 2002.

[7] E. Royer, M. Lhuillier, M. Dhome, and J. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *Int. Journal of Computer Vision*, vol. 74(3), 2007.

[8] T. Ohno, A. Ohya, and S. Yuta, "Autonomous navigation for mobile robots referring pre-recordedimage sequence," in *Proceedings of the IEEE/RSJ Int. Conf. on Intel. Robots and Sys.*, 1996, pp. 672–679.

[9] S. Jones, C. Andresen, and J. Crowley, "Appearance based processes for visual navigation," in *Proceedings of the IEEE Int. Conf. on Intelligent Robots and Systems*, 1997.

[10] L. Tang and S. Yuta, "Vision based navigation for mobile robots in indoor environment by teaching and playing-back scheme," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, vol. 3, Seoul, Korea, May 21-26 2001, pp. 3072– 3077.

[11] K. Bekris, A. Argyros, and L. Kavraki, "Exploiting panoramic vision for bearing-only robot homing," *Img. Beyond the Pinhole Cam.*, 2006.

[12] O. Booij, B. Terwijn, Z. Zivkovic, and B. Krose, "Navigation using an appearance based topological map," in *Proceeding of the IEEE Int. Conf. on Robotics and Automation*, 2007.

[13] O. Koch, M. Walter, A. Huang, and S. Teller, "Ground robot navigation using uncalibrated cameras," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, Anchorage, Alaska, United States, 2010.

[14] F. Fraundorfer, C. Engels, and D. Nister, "Topological mapping, localization and navigation using image collections," in *Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems*, 2007.

[15] A. Diosi, A. Remazeilles, S. Segvic, and F. Chaumette, "Outdoor visual path following experiments," in *Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007.

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

[17] S. Simhon and G. Dudek, "A global topological map formed by local metric maps," in *Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 3, Victoria, BC, Canada, Oct 13-17 1998.

[18] A. M. Zhang and L. Kleeman, "Robust appearance based visual route following for navigation in large-scale outdoor environments," *The Int. Journal of Robotics Research*, vol. 28(3), 2009.

[19] I. Rekleitis, J.-L. Bedwani, and E. Dupuis, "Over-the-horizon, autonomous navigation for planetary exploration," in *Proceedings IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007, pp. 2248–2255.

[20] D. Wettergreen, D. Jonak, D. Kohanbash, S. Moreland, S. Spiker, and J. Teza, "Field experiments in mobility and navigation with a lunar rover prototype," in *Proceedings of the 7th Int. Conf. on Field and Service Robotics*, Cambridge, MA, United States, July 14-16 2009.

[21] O. Wulf, A. Nuchter, J. Hertzberg, and B. Wagner, "Benchmarking ubran six-degree-of-freedom simultaneous localization and mapping," *Journal of Field Robotics*, vol. 25, no. 3, pp. 148–163, 2008.

[22] S. May, S. Fuchs, E. Malis, A. Nuchter, and J. Hertzberg, "Three-dimensional mapping with time-of-flight cameras," *Journal of Field Robotics*, vol. 26, no. 11-12, pp. 934–965, 2009.

[23] C. McManus, P. Furgale, and T. Barfoot, "Towards appearance-based methods for lidar sensors," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[24] G. Sibley, C. Mei, I. Reid, and P. Newman, "Vast-scale outdoor navigation using adaptive relative bundle adjustment," *The Int. Journal of Robotics Research*, vol. 29, no. 8, pp. 958–980, 2010.

[25] A. Vain, S. Kaasalaine, U. Pyysalo, A. Krooks, and P. Litkey, "Use of naturally available reference targets to calibrate airborne laser scanning intensity data," *Sensors*, vol. 9, pp. 2780–2796, 2009.

[26] D. Donoghue, P. Watt, N. Cox, and J. Wilson, "Remote sensing of species mixtures in conifer plantations using lidar height and intensity data," *Remote Sensing of Environment*, vol. 110, pp. 509–522, 2007.

[27] B. Holfe and N. Pfeifer, "Correction of laser scanning intensity data: data and model-driven approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 6, pp. 415–433, 2007.

[28] M. Fischler and R. Bolles, "Random sample and consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[29] J. ONeill, W. Moore., K. Williams, and R. I. Bruce, "Scanning system for lidar," United States patent US 0053715 A1, Mar. 4 2010.

[30] C. McManus, P. Furgale, and T. Barfoot, "Towards lighting-invariant visual navigation: An appearance-based approach using scanning laser-rangefinders," *Submitted to Robotics and Autonomous Systems on October 17, 2011. Manuscript # ROBOT-D-11-00284.*