2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Daejeon Convention Center
October 9-14, 2016, Daejeon, Korea

Bridging the Appearance Gap: Multi-Experience Localization for Long-Term Visual Teach and Repeat

Michael Paton, Kirk MacTavish, Michael Warren, and Timothy D. Barfoot[1]

Abstract— **Vision-based, route-following algorithms enable autonomous robots to repeat manually taught paths over long distances using inexpensive vision sensors. However, these methods struggle with long-term, outdoor operation due to the challenges of environmental appearance change caused by lighting, weather, and seasons. While techniques exist to address appearance change by using multiple experiences over different environmental conditions, they either provide topological-only localization, require several manually taught experiences in different conditions, or require extensive offline mapping to produce metric localization. For real-world use, we would like to localize metrically to a *single* manually taught route and gather additional visual experiences *during* autonomous operations. Accordingly, we propose a novel multi-experience localization (MEL) algorithm developed specifically for route-following applications; it provides continuous, six-degree-of-freedom (6DoF) localization with relative uncertainty to a privileged (manually taught) path using several experiences simultaneously. We validate our algorithm through two experiments: i) an offline performance analysis on a 9km subset of a challenging 27km route-traversal dataset and ii) an online field trial where we demonstrate autonomy on a small 250m loop over the course of a sunny day. Both exhibit significant appearance change due to lighting variation. Through these experiments we show that safe localization can be achieved by bridging the appearance gap.**

I. INTRODUCTION

Autonomous route-following methods such as Visual Teach and Repeat (VT&R) [2] allow long-range robot autonomy without the need for an accurate global map. Instead, by localizing against a previous manually driven route in a relative pose map, autonomous navigation over large trajectories is achievable [1]. Such an approach is applicable in many environments with repeated traversals over constrained paths such as factory floors, orchards, mines, urban road networks, and exploratory search-and-return missions.

Both VT&R and a large number of related localization algorithms depend on vision as the sensing modality, providing either topological maps [3] or locally metric maps through stereo vision [4]. The cost of such vision systems is low compared to Light Detection And Ranging (LiDAR) [5], at the expense of robustness to external lighting. While applications such as those in factories and mines (with suitable lighting) mean appearance change is minimal, many applications require operation in environments with vastly differing appearance. Lighting change is a significant factor over modest time scales, where shadows move throughout the day and cloud cover can change appearance from one minute to the next. Over periods of weeks or months, seasonal changes due to foliage and snow cover can also dramatically affect appearance. Even the robot, through terrain modification from repeated traverses (tire tracks, vegetation damage) can contribute to this appearance change. Autonomous repeats using unmodified grayscale imagery with VT&R have been demonstrated with teach-to-repeat time differences up to approximately 4 hours [6], extended to inter-day with colour-constant imagery [1]. However, long-term localization across significant daily and seasonal change is a current limitation of vision-in-the-loop systems that require continuous, metric localization; a necessity for active path-tracking control.

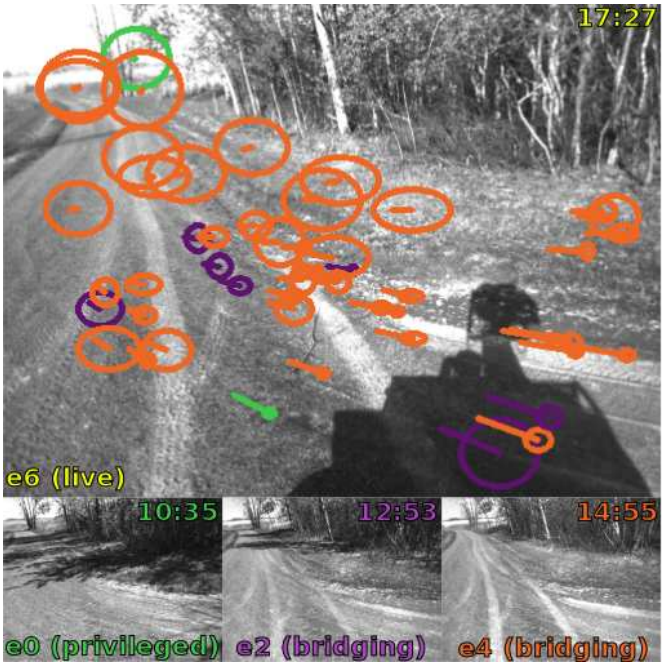


Fig. 1: Multi-experience localization (MEL) estimates the pose of a live experience with respect to a manually taught privileged experience for use in visual teach and repeat (VT&R), an autonomous route-following framework. Intermediate experiences are used to *bridge the appearance gap* between live and privileged views. Above, live experience e6 has an inadequate number of feature matches to privileged experience e0 (green) to safely localize, due to changing shadows over the elapsed 7 hours. Supplementary matches from intermediate bridging experiences e2 (purple) and e4 (orange) are added to localize e6 robustly with respect to e0. The time of each experience is noted in its top-right corner.

Currently, VT&R is limited to localizing against only one previous manual experience. We seek to address this limitation with a single overarching enhancement: the ability to continuously estimate, with uncertainty, the localization between the live experience and a privileged (manual) experience, by using several other intermediate experiences

[1]All authors are with the University of Toronto Institute for Aerospace Studies (UTIAS), 4925 Dufferin St. Toronto, ON M3H 5T6

simultaneously to bridge the appearance gap (see Figure 1). Our work differs from other systems in that we would like to have only a single manually taught experience (the privileged experience) and add the bridging experiences *during* autonomous operations. To sufficiently limit scope, we state upfront that this paper does not address the scalability problem of localizing to a large, increasing number of experiences, which *is* a requirement for true long-term operation.

The remainder of this paper is outlined as follows. Section II provides related work, Section III details our approach, Section IV sets up the experiments, Section V provides our results, and Section VI concludes the paper.

II. RELATED WORK

The autonomous route-following algorithm presented in this paper builds on the VT&R single-experience-localization system presented in [2]. This method is effective at performing long-range, vision-based route following, but is highly susceptible to lighting change. Lighting-invariant VT&R can also be achieved through the use of active sensors in place of a stereo camera, such as intensity imagery generated from LiDAR [5], and dense point-cloud registration [7]. While proven effective in situations of extreme visual appearance change through realistic field trials, these techniques rely on more costly sensors. The issue of lighting change in a vision-based, autonomous, route-following system can also be mitigated through the use of colour-constant images and additional stereo cameras [1]. However, these methods are still susceptible to failure in certain environments [6].

There has been significant recent work on the topic of localization across large appearance change. Some algorithms address this by attempting to utilize more descriptive environment features [8]. Others provide topological localization through the alignment of image sequences, providing localization across appearance change as drastic as night vs. day [9], [10], [11]. Despite great success, these methods are not suitable for vision-in-the-loop navigation on their own, as they provide topological localization only (assuming low-level lane following or other control algorithms will solve the local navigation issues). Computing vision-based, metric localization across appearance change is a difficult issue due to the reliance of most metric-estimation methods on point-based visual features, illumination values, or gradient-constancy assumptions, which are all highly susceptible to appearance change. One method of overcoming this problem is the use of prior training data. By training custom Support Vector Machine (SVM) classifiers that describe a specific scene in multiple experiences, appearance-invariant *scene signatures* can be learned and used for coarse metric localization [8], but are not precise enough on their own for our approach to autonomous route following.

Our current efforts were heavily inspired by the seminal Experience-Based Navigation (EBN) framework [4], which attempts to localize against a number of past experiences, and provides accurate metric localization to the one that is the most similar. In this system, when localization fails, the live Visual Odometry (VO) output is saved to the map as a new experience. When there are multiple experiences available in the map, a series of independent parallel localizers attempt to solve the state-estimation problem, with the 'best' result used. The computational performance of EBN was addressed in [12], where past performance metrics are used to decide which experiences to use. EBN was further expanded through the use of active sensors in [13]. While EBN has proven highly effective at providing metric localization across seasonal changes, it is not ideal for use in VT&R. This is because the resulting metric localization may be provided with respect to any of the prior experiences, implying that each experience must be driven manually (or somehow labeled as 'safe' for autonomous repeating). In VT&R, there is always some path-tracking error (in addition to localization error); if we simply add new, independent experiences during autonomous operations, the path-tracking error will build from one experience to the next as the appearance shifts, analogously to the effect of taking a photocopy of a photocopy. To be able to continue to use a single privileged (manual) experience in VT&R, we must use several experiences simultaneously, rather than independently, in order to localize safely. We therefore view our resulting Multi-Experience Localization (MEL) as a generalization of EBN to support VT&R.

Another work closely related to our multi-experience approach is the use of 'Summary Maps', presented in [21]. This method provides accurate, metric localization across seasonal appearance change through a multi-experience map that is pruned and curated offline. This mapping strategy was validated through a multi-season dataset where different offline maintenance techniques were compared to ensure real-time performance for a large number of experiences. Online localization using Summary Maps has been shown to provide accurate, metric localization across seasonal appearance change. While successful, this method requires downtime between traverses to perform mapping on an offline server, which is not ideal for many applications.

III. THEORY

A. System Overview

The MEL system outlined in this paper is designed to enable robust localization for long-term, autonomous route following. Similar to the concepts introduced in [2], our multi-experience VT&R system consists of both a teach and a repeat phase. During the teach phase, the robot is manually driven along a safe route, adding this privileged experience to the map. During the repeat phase the robot autonomously repeats by following the privileged path while adding a new experience to the existing map. This new autonomous experience is metrically localized to the privileged experience and can be used to help in future localizations; however, the goal is to always report the localization with respect to the privileged path. The robot repeats a path by sending high-frequency localization updates to a (learning-based) Model Predictive Control (MPC) path-tracking controller [14], that

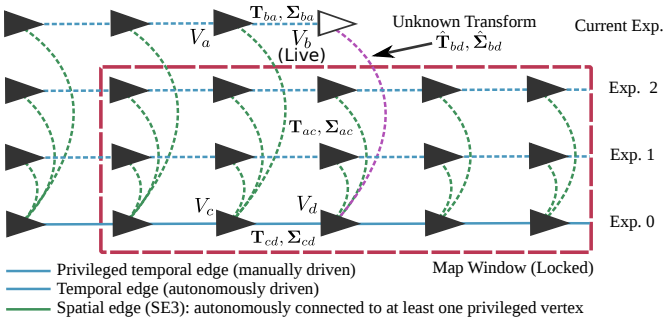


Fig. 2: Overview of the multi-experience localization (MEL) problem and the spatio-temporal pose graph (STPG) data structure used as our map. We wish to estimate the unknown transform and uncertainty, $\{\hat{\mathbf{T}}_{bd}, \hat{\mathbf{\Sigma}}_{bd}\}$ (dashed, purple line), between the live vertex, V_b, and the target vertex, V_d, in the privileged path (solid blue line). This is achieved by matching all landmarks in V_b to landmarks observed in the map window (dashed, red rectangle), transformed into the coordinate frame of V_d. This setup allows for outlier rejection and a simple optimization of $\{\hat{\mathbf{T}}_{bd}, \hat{\mathbf{\Sigma}}_{bd}\}$ against a map of locked landmarks with uncertainty.

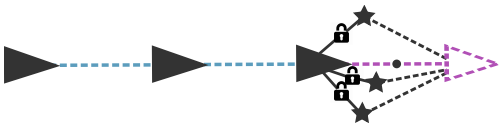
leverages multiple experiences to improve controller performance over time. The remainder of this section provides details on the mapping process (Section III-B) and the multi-experience localization process (Section III-C).

B. Map Building

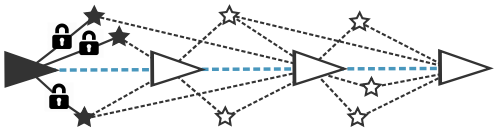
The map used in our system, which we refer to as a Spatio-Temporal Pose Graph (STPG), is depicted in Figure 2. This data structure is an undirected graph, $G = \{V, E_s, E_t\}$, where V is a set of vertices, E_t is a set of *temporal* edges, and E_s is a set of *spatial* edges. Vertices, each with an associated reference frame, $\mathcal{F}$, store raw sensor observations and triangulated 3D landmarks with associated covariances and descriptors[1]. An edge in the graph links vertices metrically with a relative $SE(3)$ transformation with uncertainty. Temporal edges (blue lines) link vertices that are temporally adjacent, while spatial edges (green lines) link vertices that are temporally distant yet spatially close. Temporal edges can furthermore be denoted as *privileged* if they were collected while the robot was being taught a route or *autonomous* if the robot was repeating a route; this distinction is illustrated in Figure 2 as solid and dashed lines, respectively. We define an *experience* as a collection of vertices linked by temporal edges.

Mapping consists of adding either a privileged or autonomous experience to a new or existing STPG while computing data products and temporal edge transformations through our stereo VO pipeline, which is similar to Parallel Tracking and Mapping (PTAM) [15] and is illustrated in Figure 3. For each incoming stereo frame (the *live* frame), sparse visual features are extracted and triangulated. Features are represented by a stereo measurement, $\{\mathbf{y}, \mathbf{Y}\}$, where $\mathbf{y}$ is the 4×1 keypoint position of the left-right stereo features and $\mathbf{Y}$ is the 4×4 covariance on that measurement. We

[1]We use SURF features triangulated from stereo measurements in our implementation, but the overall system is generic to any point-based sparse visual feature.



(a) High-rate: frame-to-vertex VO. Landmark estimates (black stars) at the latest vertex are locked, and the motion estimate is solved using new matches from the live frame (purple triangle) to the last vertex (black triangle) while using a smoothing trajectory prior (black dot).



(b) Low-rate: sliding-window vertex bundle adjustment. Transforms and landmarks connected to unlocked vertices (white triangles) are optimized and those connected to locked vertices (black triangles) are locked.

Fig. 3: VO pipeline showing the parallel high-rate, approximate (a) and low-rate, accurate (b) estimators similar to [15].

use upright Speeded Up Robust Features (SURF) [16] to detect and describe keypoints and calculate $\mathbf{Y}$ based on the octave and Hessian of the response. The stereo measurement is triangulated via the inverse stereo camera model to obtain a 3D landmark including uncertainty, $\{\mathbf{p}, \mathbf{\Phi}\}$, where $\mathbf{p}$ is the 4×1 positional mean in homogeneous coordinates and $\mathbf{\Phi}$ is the uncertainty represented by a 3×3 covariance.

Extracted landmarks from the live view are matched via their appearance to locked landmarks in the latest graph vertex (a.k.a., keyframe) and motion computed (Figure 3a). A trajectory (velocity and position) estimate is produced at frame rate from the optimization and can be queried to predict future motion. This prediction is used to project landmarks into the new frame (reducing image search space for matching) and compensates for latency between the localization system and the path-tracking controller. If the translational or rotational motion is large, or the number of matched features between the live view and the last graph vertex drops too low, the live frame is inserted as a new vertex in the graph; otherwise, it is discarded. Upon insertion of a new vertex, a temporal edge linking to the previous vertex is added. If the robot is being taught a path, this edge is flagged as privileged.

Following vertex insertion, graph optimization is performed on a sliding window of the latest vertices in the graph (Figure 3b) using our Simultaneous Trajectory Estimation And Mapping (STEAM) [17] engine; smoothing factors are added to the relative transforms to ensure stability in the estimated trajectory during areas of poor feature tracks. After optimization, the updated poses, landmarks, and their uncertainties are re-inserted into the graph.

C. Multi-Experience Localization

This section describes the MEL algorithm, our main contribution. Depicted in Figure 2, this localization algorithm is designed to support long-term, autonomous route following. The overall objective of the algorithm is to estimate the posterior transform and uncertainty, $\{\hat{\mathbf{T}}_{bd}\hat{\mathbf{\Sigma}}_{bd}\}$, between the

most recent vertex in the live run, V_b, and the estimated closest vertex in the privileged path, V_d. This is achieved by minimizing the measurement error of landmarks in the multi-experience map window (red, dashed rectangle) observed by V_b. Throughout the algorithm we make use of the prior term, $\{\check{\mathbf{T}}_{bd}, \check{\mathbf{\Sigma}}_{bd}\}$, obtained by compounding the uncertain transforms [18],

$$\{\mathbf{T}_{ba}, \mathbf{\Sigma}_{ba}\},\ \{\mathbf{T}_{ac}, \mathbf{\Sigma}_{ac}\},\ \{\mathbf{T}_{cd}, \mathbf{\Sigma}_{cd}\}, \tag{1}$$

which are computed through previous VO and previous localization estimates. The MEL pipeline consists of the following main steps: a) Landmark Transformation, b) Multi-Experience Matching, and c) State Estimation.

a) Landmark Transformation: The first step of MEL is to transform all landmark means and uncertainties originating from vertices in the active map window from their respective coordinate frames to $\mathcal{F}_d$, the coordinate frame of V_d and the one in which localization is to be computed. Given a 3D landmark expressed in a some vertex map frame, $\mathcal{F}_m$, with mean and covariance, $\{\mathbf{p}_m, \mathbf{\Phi}_m\}$, the transformation to $\mathcal{F}_d$ is given by:

$$\mathbf{p}_d = \mathbf{T}_{dm}\mathbf{p}_m \tag{2}$$

$$\mathbf{\Phi}_d = \mathbf{D}^T \mathbf{p}_d^{\odot} \mathbf{\Sigma}_{dm} \mathbf{p}_d^{\odot T} \mathbf{D} + \mathbf{D}^T \mathbf{T}_{dm} \mathbf{D} \mathbf{\Phi}_m \mathbf{D}^T \mathbf{T}_{dm}^T \mathbf{D}, \tag{3}$$

where $\odot$ is an homogeneous-point operator [18] given by

$$\begin{bmatrix} \boldsymbol{\epsilon} \\ \eta \end{bmatrix}^{\odot} = \begin{bmatrix} \eta \mathbf{1} & -\boldsymbol{\epsilon}^{\wedge} \\ \mathbf{0}^T & \mathbf{0}^T \end{bmatrix}, \tag{4}$$

with

$$\boldsymbol{\epsilon}^{\wedge} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix}^{\wedge} = \begin{bmatrix} 0 & -\epsilon_3 & \epsilon_2 \\ \epsilon_3 & 0 & -\epsilon_1 \\ -\epsilon_2 & \epsilon_1 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \tag{5}$$

and $\mathbf{1}$ is the identity matrix. This process is carried out on all landmarks in the map window to produce a set of landmarks with 3D position and uncertainty, all expressed in the privileged frame, $\mathcal{F}_d$. While it may seem superfluous to transform the locations and uncertainties of landmarks that are not yet known to be inlier matches, these help refine the matching process, making it faster and more robust.

Bookkeeping the uncertainties is a critical aspect of making MEL work well; the bridging experiences are daisy-chained over time from the privileged experience, so while they may have more matches to the live experience (than those directly from the privileged experience) due to more similar appearance, the spatial uncertainty of those matches may be higher when transformed to the privileged frame, $\mathcal{F}_d$. Keeping track of all the uncertainties ensures we properly weight all the matches in our localization.

b) Multi-Experience Matching: The goal of multi-experience matching is to associate every landmark in V_b to a landmark in the map window. The process begins with labeling all landmarks in the live vertex as unmatched. Vertices in the map window are sequentially examined starting from V_d in a breadth-first-search pattern. We chose to center the search around the privileged target vertex as a heuristic for prioritizing landmarks that have the lowest uncertainty in the target privileged frame. For every new vertex visited, the transformed map landmarks associated with this vertex are projected into the camera frame of vertex V_b using the prior term, $\{\check{\mathbf{T}}_{bd}, \check{\mathbf{\Sigma}}_{bd}\}$. Each landmark associated with this vertex is then checked for matching feasibility to the unmatched live landmarks by comparing keypoint position and descriptor appearance. This process continues until one of three criteria are met: i) a sufficient number of matches are found, ii) the amount of time has surpassed the allowance, or iii) the map window is exhausted. As the process of comparing visual features is costly and the size of the map window grows linearly with experiences, this process is the most computationally expensive step of multi-experience localization.

Upon completion of landmark matching, the problem is set up so that there are associated 3D landmarks in the coordinate frames of V_b and V_d. This information is sent through a locally optimized Maximum Likelihood Estimation SAmple Consensus (MLESAC) [19] implementation to remove outliers and provide an initial estimate of the posterior transform between V_b and V_d.

c) State Estimation: We now seek the optimal posterior,

$$\{\hat{\mathbf{T}}_{bd}, \hat{\mathbf{\Sigma}}_{bd}\}, \tag{6}$$

given the prior term, $\{\check{\mathbf{T}}_{bd}, \check{\mathbf{\Sigma}}_{bd}\}$, as well as associated data between V_b and map landmarks in the coordinate frame of V_d. This can be achieved by minimizing the following objective function:

$$J(\mathbf{T}_{bd}) = \frac{1}{2} \sum_{j=1}^{M} \mathbf{e}_j{}^T \mathbf{R}_j^{-1} \mathbf{e}_j + \frac{1}{2} \mathbf{e}^T \mathbf{R}^{-1} \mathbf{e}. \tag{7}$$

The first term in J sums the squared reprojection error of map landmarks. Given a map landmark, j, with mean and uncertainty, $\{\mathbf{p}_{d,j}, \mathbf{\Phi}_{d,j}\}$, expressed in the coordinate frame of V_d and a stereo measurement of j, $\mathbf{y}_j$, with uncertainty, $\mathbf{Y}_j$, expressed in the camera frame of V_b, the reprojection error is defined by

$$\mathbf{e}_j = \mathbf{y}_j - \mathbf{g}(\mathbf{T}_{bd}\mathbf{p}_{d,j}), \tag{8}$$

$$\mathbf{R}_j = \mathbf{Y}_j + \mathbf{G}_j \mathbf{T}_{bd} \mathbf{D} \mathbf{\Phi}_{d,j} \mathbf{D}^T \mathbf{T}_{bd}^T \mathbf{G}_j^T, \tag{9}$$

where $\mathbf{g}(\cdot)$ is the stereo measurement model and $\mathbf{G}_j$ is its Jacobian (evaluated at $\mathbf{p}_{b,j} = \mathbf{T}_{bd}\mathbf{p}_{d,j}$). This weights each error by uncertainty in the measurement and the map. The second term of Equation 7 constrains the optimization problem by the prior with

$$\mathbf{e} = \ln(\check{\mathbf{T}}_{bd}\mathbf{T}_{bd}^{-1})^{\vee}, \quad \mathbf{R} = \check{\mathbf{\Sigma}}_{bd}, \tag{10}$$

where $\vee$ is the inverse operator of $\wedge$ [18]. To obtain an optimal posterior estimate, $\hat{\mathbf{T}}_{bd}$, Equation 7 is iteratively linearized and refined in a nonlinear least-squares optimization using our STEAM engine [17]. In the absence of any matches between the live image and map, the prior estimate (based on VO) is returned.

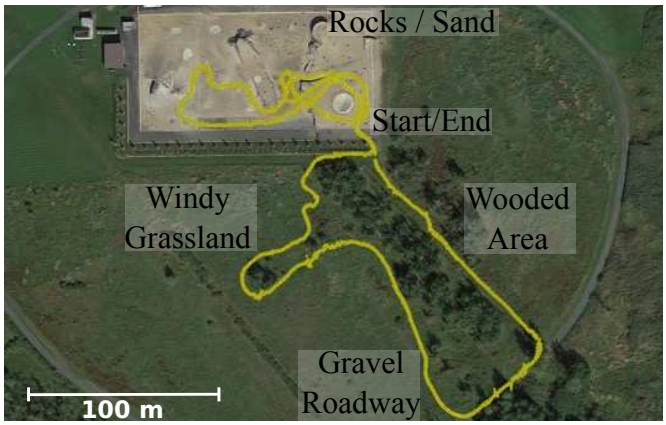


Fig. 4: Satellite imagery of the Canadian Space Agency's Mars Emulation Terrain and surrounding woodland. A 1km route was driven 27 times across a wide variety of lighting conditions to gather the dataset used for evaluation.

IV. Experimental Setup

This section describes the experiments conducted to validate our MEL algorithm. For both experiments, a Clearpath Grizzly Robotic Utility Vehicle (RUV) (see Figure 5b) is used, fitted with a Point Grey XB3 camera system.

A. Offline Localization Experiment

To assess the impact of adding bridging experiences in the new multi-experience framework, a series of experiments were conducted offline using a stereo-imagery dataset collected at the Canadian Space Agency (CSA)'s Mars Emulation Terrain (MET) in Montreal, Quebec [1]. The full set of data consists of a single 1km teach pass on day 1, covering a wide variety of environments (simulated Mars terrain, grassy field and wooded paths; see Figure 4), which is then followed by 26 autonomous repeats on the same path over the same and following days (see Table I). Appearance changed significantly due to sunny conditions with harsh shadows on the first day and overcast weather on the second, as well as terrain modification from the vehicle (tire tracks). The autonomous repeats were performed using (previously published) colour-constant imagery and the robot maintained path-following autonomy over 99.9% of the route [1]. For the experiments in this paper, however, we use only the raw grayscale imagery to challenge the performance of the MEL framework over a subset of the repeat runs.

The offline experiments consisted of simulating localization using varying numbers of experiences from the CSA dataset, (Table I). The first experience, e0, is the manually driven, privileged experience, while the remainder are autonomously driven experiences (e1-e6, e16, e27) of the same route. Details of the localization experiments (the sets of experiences) are listed in Table II. The first set of experiments (g0-g5) analyze the performance of localization between the privileged experience (e0) and an experience gathered approximately seven hours later (e6) with an increasing number of bridging experiences. We choose to focus on e6 for the reason that it has the worst localization performance against e0 (even when tested with colour-constant imagery

TABLE I: Overview of the experiences in the CSA dataset.

ID	Start Time	Duration [hh:mm]	Δt [hh:mm]	Sky Condition
e0	2014/05/12 10:35	00:34	00:00	sunny
e1	2014/05/12 11:40	00:28	01:05	sunny
e2	2014/05/12 12:53	00:27	02:18	sunny
e3	2014/05/12 13:35	00:26	03:00	sunny
e4	2014/05/12 14:55	00:31	04:20	sunny
e5	2014/05/12 16:06	00:32	05:31	cloudy
e6	2014/05/12 17:27	00:26	06:52	sunny
e16	2014/05/13 11:00	00:27	24:25	cloudy
e27	2014/05/15 08:50	00:25	70:31	sunny

TABLE II: Overview of the graph configurations used for multi-experience localization evaluation.

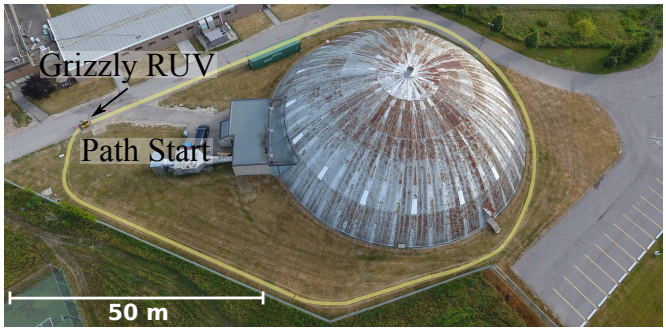
ID	Live experience	Privileged experience	Bridge experiences
g0	e6	e0	–
g1	e6	e0	e3
g2	e6	e0	e2, e4
g3	e6	e0	e1, e3, e5
g4	e6	e0	e1, e2, e4, e5
g5	e6	e0	e1, e2, e3, e4, e5
g6	e27	e0	–
g7	e27	e0	e1, e2, e4, e5
g8	e27	e0	e1, e2, e4, e5, e16

in previous experiments) due to significant lighting changes. Making the assumption that the change is roughly equal throughout the day, we add experiences in a mean-splitting pattern as a simple heuristic; e.g., e6 is localized against e0 in test g0, then e3 is added as the bridging experience, etc. This is continued until test g5, where all bridging experiences are used to localize e6. Identifying the optimal set of bridging experiences is not a goal of this paper; we instead aim to show how such experiences can be used effectively.

The next set of experiments (g6-g8) test the performance of e27 as the live experience. The experience e27 is the most temporally distant to e0 (by more than 70 hours) and contains significant terrain modification due to the robot carving deep troughs in the forest environment and other robots creating tire tracks in the sand in the MET. To test localization to e27, we introduce three experiments using an increasing number of bridging experiences. In experiment g6, we test using no bridging experiences. In experiment g7, we add a set of experiences that capture the lighting change seen in the first day, (e1, e2, e4, e5). Finally, in experiment g8, we use the aforementioned set as well as one experience from the second day, e16, that captures overcast conditions and terrain modification. We hypothesize that adding an extra experience with overcast conditions will significantly increase performance with the rationale that overcast conditions are easy to localize against, regardless of lighting conditions.

B. Online VT&R Experiment

A small field trial was conducted to demonstrate the online performance of our MEL algorithm in a fully autonomous route following setting. The experiment consisted of manually teaching a 250m path (Figure 5a) and autonomously repeating the path every hour for approximately 10 hours. In



(a) Overview of the experiment grounds, the grizzly RUV is show at the start of the approximately 250m path.

(b) Close up of the Grizzly RUV platform autonomously repeating the path with relevant sensors displayed.

Fig. 5: Imagery of the online VT&R experiment conducted at the UTIAS campus.

this experiment, we use grayscale *and* color-constant stereo images as an input to our system similar to our previous work [1]. This helps mitigate the effects of lighting change, thus reducing the number of bridging experiences needed.

C. Evaluation Metrics

To evaluate MEL using the aforementioned experiments, we selected three metrics: a) Cross-track uncertainty, b) Feature inlier count, and c) Computation time.

a) Cross-track uncertainty: This is our primary metric for judging localization success. We define cross-track uncertainty as the one-standard-deviation uncertainty of our lateral translation estimate relative to the privileged path. This tells us how uncertain we are to the left or the right while following the privileged path, and can be directly interfaced with our path-tracking controller to provide safe autonomous driving based on the lateral constraints of the path. It is important to note that while uncertainty is calculated at every stage of the algorithm from keypoint detection to landmark transformation, we have not yet performed a rigorous evaluation of our uncertainty estimates with respect to ground truth to ensure consistency. Therefore we treat this metric as a way to compare relative performance between experiments and do not necessarily trust the exact scale of our uncertainty estimates.

b) Feature inlier count: This metric is simply the number of inliers observed (after MLESAC) at each localization point over the entire traverse for each experiment. In the offline experiment, this metric evaluated as a Cumulative Distribution Function (CDF) provides a measure of how much each experience adds to the state estimation problem by examining experiments g0-g5. In the online experiment, evaluating the median inlier count for each repeat highlights the ability to localize in real-time even with the addition of more experiences. For localization success, we require at least 10 self-consistent feature matches (i.e., after MLESAC) or fall back to the prior alone (i.e., VO).

c) Computation time: As complexity of the MEL algorithm scales linearly with the number of experiences in the worst case scenario (when matching reaches the upper bound of time), we are interested in observing the average computation time of localization for each experiment. In order for this algorithm to support vision-in-the-loop route following, this solve time needs to be fast enough to support the incoming localization requests from the path tracker. We currently have this set to 250ms, which is based on the average rate of vertex additions to the graph, primarily based on distance driven by the robot.

V. RESULTS

The goal of this section is to demonstrate significantly improved localization performance using MEL with a varying number of bridging experience (none to several) and show that MEL is capable of performing online while progressively adding extra bridging experiences. This is achieved by analyzing the results of the experiments with respect to the performance metrics detailed in the previous section. For the offline experiments, we analyze localization uncertainty and feature inlier count. For the online experiment, we analyze feature inlier count and localization computation time. We deliberately choose not to compare our algorithm with the most closely related work, Experience-Based Navigation [4] with the rationale that we are solving a different localization problem (i.e., to a single privileged path) and therefore the results are not quantitatively comparable.

A. Offline Localization Experiment

a) Cross-track uncertainty: Results of all offline experiments with respect to cross-track uncertainty are presented in Figure 6. This shows the CDF of the cross-track uncertainty for the entire traverse for each experiment. In the worst case scenario (g0, single-experience localization), the robot would have driven with a maximum cross-track uncertainty of approximately 3.5m. Only 70% of the traverse would be driven with a cross-track uncertainty less than 1m. This is in stark contrast with experiment g2, which uses bridging experiences spaced evenly two hours apart from each other. Maximum cross-track uncertainty observed in g2 never exceeded 0.3m. Additionally, 90% of the traverse was driven with a cross-track uncertainty of less than 0.05m. This is shown in Figure 7 with example images of some sections highlighted in Figure 8.

The results from g0-g5 indicate that when appearance is gradually changed between the live and privileged view, adding more bridging experiences improves the localization performance. The dramatic increase in performance between

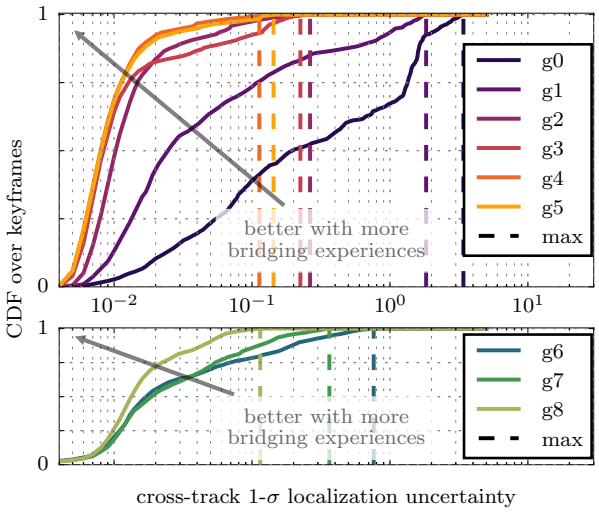


Fig. 6: Cumulative distribution (over keyframes) of the cross-track (left-right) localization uncertainty. This figure reads: *for Y fraction of the traverse, the robot's* 1σ *cross-track uncertainty was less than X metres*. The uncertainty units are in metres, though it is a relative measure (see Cross-track uncertainty). The first set of experiments, g0–g5 (see Table II), show dramatic improvement with the addition of the first two bridging experiences, and very little improvement beyond. The second set of experiments, g6–g8, also shows significant improvement with more bridging experiences.

g1 and g2 indicates that, in sunny conditions, an experience every two hours is most likely sufficient for autonomous navigation. Results from g6-g8 further show this trend. It is interesting to note that the single-experience experiment, g6, has a generally lower uncertainty than the single-experience experiment, g0. This is because despite e27 being the farthest in time from the privileged experience, e0, it is the closest in appearance; e27 took place at 08:50, three days later, when it was also sunny outside and only about two hours earlier (but on a different day). Results from g8 show that the addition of an overcast experience greatly increases performance.

b) Feature inlier count: The feature inlier counts with respect to the offline experiments are presented in Figure 9. This figure shows the CDF of inlier matches found between the live view and experience map for each experiment. The number of inliers found is highly correlated with the cross-track uncertainty. Therefore, the order of inlier-match performance across experiments is similar to that seen in the cross-track uncertainty results.

B. Online VT&R Experiment

Results from the online experiment are displayed in Figure 10. They show the statistical distribution of both median feature inlier counts and MEL computation times for each autonomous repeat (dots) performed in the experiment.

a) Feature inlier count: Median feature inlier count for each repeat is shown in Figure 10a. The results show that the inlier count (blue line) slowly drops for the first two hours of appearance change before stabilizing near 50 inliers.

b) Computation time: Timing results are shown in Figure 10b. The results show that despite accumulating up to nine experiences in the map at the final run, the median computation time remains below 100 ms. It is worth noting

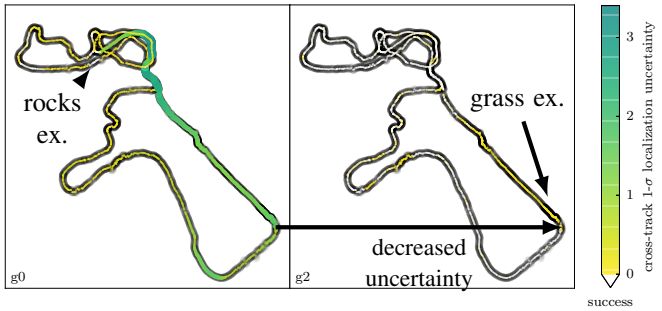
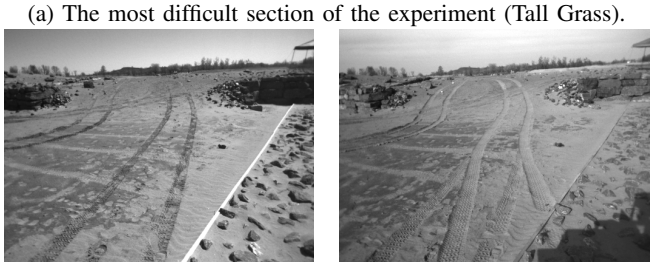


Fig. 7: Cross-track 1σ uncertainty (coloured) for two interesting experiments (g0 and g2) overlaid on the GPS path of the vehicle (black, with a 3m margin), with successful localizations coloured white. The top section of the path is in a rocks-and-sand environment, while the bottom section is in vegetation. The section of the path that remains difficult even for g2 is in 1m tall grass (see Figure 8a), with tall trees on either side that cast long shadows. Images from the rocks and grass examples can be see in Figure 8.

(a) The most difficult section of the experiment (Tall Grass).



(b) An easy section of the experiment (Rock Piles).

Fig. 8: Example images each showing e0 (left) and e6 (right), showing (a) grass (hard) and (b) rocks (easy) localization situations.

the gradual increase in computation costs. This is due to the number of experiences in the map. In order for this algorithm to be tractable in longer operations, the maximum amount of experiences used in a localization solve will need to be capped.

VI. CONCLUSIONS AND FUTURE WORK

This paper presented a multi-experience localization (MEL) algorithm designed for autonomous route-following applications. A key contribution is the ability to localize to a single privileged (manual) experience by using bridging experiences. Through an in-depth dataset analysis, we showed that our algorithm is able to provide metric localization to a privileged experience across significant lighting change by bridging the appearance gap with intermediate experiences.

While the results of the paper are promising, there are areas that could benefit from future work. First, the issue of temporal scale must be addressed. In the short term, we intend to limit the number of experiences used in a MEL solve

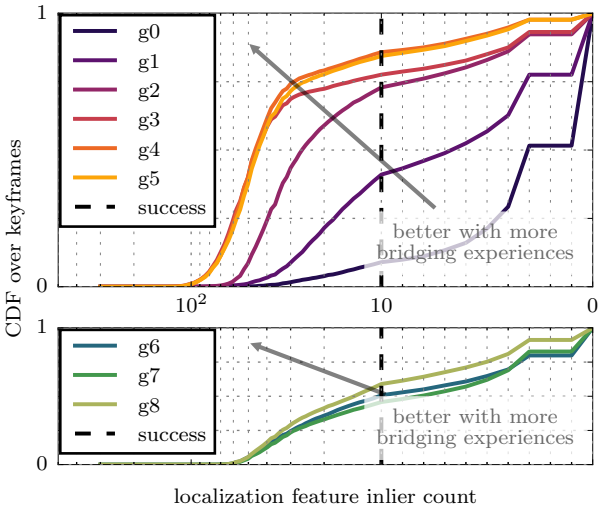
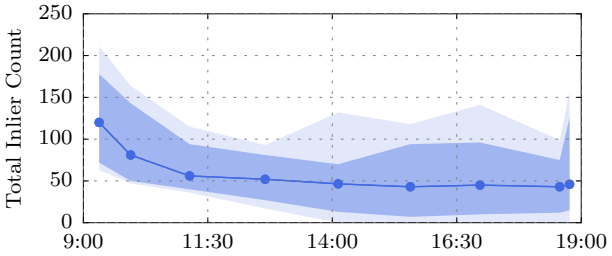
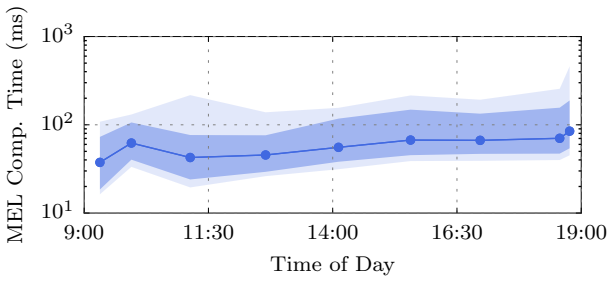


Fig. 9: Cumulative distribution (over keyframes) of the number of inlier matches to any previous experience being used. We require at least 10 inliers (dashed line) to accept the localization as a success, which is why g0 performed so poorly in the uncertainty metric; only 10% of the time did it have enough inliers to be accepted. By far the most dramatic improvement is adding one (g1) or two (g2) bridging experiences when trying to localize e6.



(a) Progression of the number of inlier matches throughout the day for the MEL algorithm as appearance changes due to lighting.



(b) Progression of the computation time throughout the day for the MEL algorithm as the STPG grows with multiple experiences. *note: log scale on y axis*

Fig. 10: Results of the online VT&R experiment. All data is represented as Tukey boxplots, where the lines correspond to the median, the dark shaded areas represent the bounds of the min/max whiskers, and the light shaded area show the bounds of outlier points.

and develop fast methods to recommend the best experiences online. In the long term, we intend to explore an anytime, active-matching approach [20] that adds bridging experiences until the localization uncertainty is sufficiently low to meet the path-tracking controller's requirements. Finally, we plan on performing additional field trials to measure accuracy vs. ground truth as well as inter-seasonal appearance change.

ACKNOWLEDGMENT

This work was supported financially and in-kind by Clearpath Robotics and the Natural Sciences and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN).

REFERENCES

[1] M. Paton and K. MacTavish and C.J. Ostafew and T.D. Barfoot, "Expanding the Limits of Vision-Based Localization for Long-Term Route-Following Autonomy," *Journal of Field Robotics*, special issue on Field and Service Robotics, to appear, 2016.

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

[3] A. Glover, W. Maddern, and M. Warren, "OpenFABMAP: An open source toolbox for appearance-based loop closure detection," in *Proc. of ICRA*, (Saint Paul), 2012.

[4] W. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *The Int. Journal of Robotics Research*, vol. 32, no. 14, pp. 1645–1661, 2013.

[5] C. McManus, P. Furgale, B. Stenning, and T. Barfoot, "Visual teach and repeat using appearance-based lidar," in *Proc. of the Int. Conf. on Robotics and Automation (ICRA)*, (St. Paul, Minnesota, USA), 2012.

[6] M. Paton, F. Pomerleau, and T. Barfoot, "In the dead of winter: Challenging vision-based path following in extreme conditions," in *Proc. of Field and Service Robotics (FSR)*, 2015.

[7] P. Krüsi, B. Bücheler, F. Pomerleau, U. Schwesinger, R. Siegwart, and P. Furgale, "Lighting-Invariant Adaptive Route Following Using ICP," *Journal of Field Robotics*, vol. 32, no. 4, pp. 534–564, 2014.

[8] C. McManus, B. Upcroft, and P. Newman, "Learning place-dependant features for long-term vision-based localisation," *Autonomous Robots*, vol. 39, no. 3, pp. 363–387, 2015.

[9] M. Milford and G. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proc. of ICRA*, 2012.

[10] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Robust visual robot localization across seasons using network flows.," in *Proc. of the Conf. on Artificial Intelligence*, 2014.

[11] E. Pepperell, P. Corke, and M. Milford, "Automatic image scaling for place recognition in changing environments," in *Proc. of ICRA*, 2015.

[12] C. Linegar, W. Churchill, and P. Newman, "Work Smart, Not Hard: Recalling Relevant Experiences for Vast-Scale but Time-Constrained Localisation," in *Proc. of ICRA*, 2015.

[13] W. Maddern, G. Pascoe, and P. Newman, "Leveraging Experience for Large-Scale LIDAR Localisation in Changing Cities," in *Proc. of (ICRA)*, (Seattle, WA, USA), 2015.

[14] C. J. Ostafew, A. P. Schoellig, T. D. Barfoot, and J. Collier, "Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking," *Journal of Field Robotics*, vol. 33, no. 1, pp. 133–152, 2016.

[15] G. Klein and D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," in *Int. Symposium on Mixed and Augmented Reality*, pp. 1–10, nov 2007.

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

[17] S. Anderson and T. Barfoot, "Full steam ahead: Exactly sparse gaussian process regression for batch continuous-time trajectory estimation on SE(3)," in *Proc. of IROS*, 2015.

[18] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with three-dimensional poses for use in estimation problems," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.

[19] O. Chum, J. Matas, and J. Kittler, "Locally optimized RANSAC," in *Pattern recognition*, pp. 236–243, Springer, 2003.

[20] M. Chli and A. J. Davison, *Proc. of the European Conf. on Computer Vision (ECCV)*, ch. Active Matching. 2008.

[21] P. Mhlfellner and M. Brki and M. Bosse and W. Derendarz and R. Philippsen and P. Furgale, "Summary Maps for Lifelong Visual Localization," *Journal of Field Robotics*, Early View, 2016.