A Framework for Infrastructure-Free Warehouse Navigation

Matthew Gadd, Paul Newman

***Abstract*— This paper presents a universally applicable graph-based framework for the navigation of warehouse robots equipped with only monocular cameras. We strongly advocate the use of relative pose information stored in a topological map, rather than a globally consistent metric representation of the environment. We show how multiple traversals of adjacent workspaces can be naturally "stitched" together in the course of a typical warehouse picking and shelving schedule to create a network of reusable paths in which the robot can efficiently localise and plan new routes. This allows us to command the robot to return to any of the previously visited locations not necessarily through the same route that we taught it. Unlike state-of-the-art teach and repeat systems using stereo vision, our approach exploits the strongly planar nature of the data obtained from a downward-facing camera, and creates odometric constraints by tracking the perceived texture of the floor and computing a simple homography. To demonstrate the robustness of our system, we validate our approach on datasets collected over a week-long period within a challenging and representative environment in the form of a warehouse shelving area.**

I. INTRODUCTION

Robots operating in warehouse environments lack access to a global position system (GPS). Retrofitting these environments with infrastructure to accurately locate and position robots may be undesirable both in terms of cost and lack of flexibility in the face of operational rearrangement of the business unit. The manual creation and automatic reuse of paths through the robot's environment is known as teach and repeat (T&R). T&R alleviates the requirement for outright simultaneous localisation and mapping (SLAM), which may suffer from the size of the environment, and provides a natural framework for learning appropriate behaviour in environments the robot may repeatedly traverse.

This paper describes a framework for warehouse navigation, using monocular cameras mounted on the robot and a T&R operational strategy, which is suitable for a wide range of sensor modalities. During a learning phase, the robot is piloted along a route within a section of the warehouse, in this way being educated by operators during their normal picking and shelving tasks. The system encourages a comprehensive and interconnected representation of the warehouse in the form of a large network grown over the course several such operator-led missions. During the repeat phase, the robot localises within this network of submaps using a simple and efficient search in the local graph neighbourhood. Planning in the network using this same search algorithm allows the robot to autonomously navigate from one point of interest to another, not necessarily via a path that it was previously taught.

Mobile Robotics Group, University of Oxford, Oxford, England; {mattgadd, pnewman}@robots.ox.ac.uk

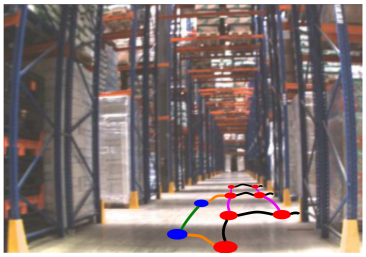


Fig. 1. A typical warehouse environment. We espouse a topological framework which stores representations for places on map (red) and query (blue) nodes. To enable robust navigation in warehouse environments, we dictate some requirements for a characterisation of a place-to-place relation stored on map (black), query (green) and loop closure (purple) edges.

The paper is organised as follows. Section II gives a brief overview of the state-of-the-art in mapping and path following. Section III provides the operational and algorithmic details of our approach. Section IV details and evaluates some pertinent real-world examples which are concluded in Section V to indicate the promise of the proposed system in enabling large-scale, infrastructure-free warehouse navigation.

II. RELATED WORK

Order-picking is responsible for more than 60% of operational costs in goods distribution [1], yet existing systems for navigation in warehouses impair flexibility by embedding infrastructure into the workspace, for example by following a series of barcode stickers on the floor [2].

For mobile robots operating in large or dynamic environments with only onboard sensors, an intuitively sound behaviour would consist in memorising some key characteristics of an already driven path, and to utilise these references as checkpoints for a future navigation task along a similar route. Scenarios in which this approach would be useful are numerous and extend beyond warehousing, including planetary sample return missions [3], charging station homing for electric wheelchairs [4], and autonomous underground tramming for mining [5].

In [5], laser range finder data and data from odometric sensors were fused to build an atlas of metric maps. Laser-based systems such as these are well suited to indoor or underground structured environments in which it can be guaranteed that any walls will be within the range of a laser sensor. Additionally, recent work by [6], in which data from a spinning three-dimensional laser scanner and a system based on iterative closest point matching take into consideration changes in ambient lighting, further enhanced the robustness of laser-based systems. Cameras however, and particularly monocular sensors, present a low-cost alternative to expensive laser scanners, especially considering the density of information they provide. Cameras capture the geometry and appearance of a scene unique to a particular viewpoint and thus offer robustness to small changes in the world.

An early maxim for vision-based map building was proposed in [7], admitting to the inevitable global inconsistency of maps in the face of sensing uncertainty, but calling for the recognition that maps need only be locally consistent to enable autonomy. Subscribing to this notion, early visual teach and repeat (VT&R) work centred around the idea of a view-sequenced route representation [8] that was purely topological and required only matching between the current view and the memorised sequence using template matching by correlation techniques. In [3] the effectiveness of a hybrid topological/metric representation was exemplified by repeating 32 km of routes taught with a stereo camera with 99.6% autonomy, all without the use of a global positioning system (GPS). The work of [9] built the visual memory for path following in a purely metric fashion, estimating the poses of a subset of the camera positions with respect to the reference sequence and a set of landmarks in a global coordinate system. In contrast, our system is purely topological, and we show how the robot can effectively navigate around its environment without access to a metrically accurate representation of that environment, so long as it has access to accurate frame-to-frame characterisation of its motion, which we formalise and demand certain requirements of.

VT&R has the prerequisite of visual navigation, which appearance-based techniques approach by comparing large portions of the input image with prototype images captured during the teach pass. An impressive demonstration of such a system was shown in [10] in which a template from each new image was correlated in the Fourier domain in order to recover the difference in relative orientation, and thus the desired steering angle. They reported more than 18 km of tests using an omnidirectional imaging system.

In contrast, algorithms that use sparse image features but rely on planarity of the camera's motion can be successful and can reduce the complexity of the problem, which is especially useful in systems such as ours to which a downward-facing camera provides only floor imagery. The method in [11] consists in tracking visual features in panoramic views of the environment and uses only the bearing of the measurements to develop a control law to drive the robot between viewpoints instead of successively triangulating the features. The visual servoing law used in [12] is aided by an upward-facing camera and tracking of features to solve three-degree-of-freedom homographies. Our method operates entirely on a frame-to-frame basis, and we show that reasonably textured surfaces provide sufficient features to solve for a robust estimation of the incremental odometry. We emphasise this point by successfully localising on several significantly different textured surfaces.

VT&R systems are susceptible to both gradual and ephemeral lighting variations. In an effort to deal with this, [13] use a light detection and ranging device to generate synthetic images and then apply vision techniques for motion estimation. Warehouse shelving areas are mostly subject to modest lighting changes during operating hours, and so our approach delegates this required robustness to a simple preprocessing of recorded or captured images.

In [14] a proposed network of reusable paths as an extension to VT&R systems uses an arbitrary graph of nodes rather than a simple linear chain of poses that would restrict the robot to moving, often very inefficiently, along the exact route taught to it. A byproduct of our localisation allows for the "stitching" together of different experiences of adjacent areas into a large network that comprehensively describes areas the robot commonly travels to on its scheduled tasks. Furthermore, using the same simple graph search algorithm that is the mainstay for our localisation, we are able to furnish the robot with the ability to plan efficient routes through the workspace that do not necessarily correspond to the trajectory it was taught.

III. System Overview

The key processing components of our pipeline for warehouse navigation are depicted in Figure 2. The teach phase builds the topological map adhering to the universal framework described in Section III-A and implemented as detailed in Section III-B. The bidirectional information flow between localiser and database storing this map allows the robot to grow a network representation of the warehouse in its repeat phase by the method expressed in Section III-C, enabling autonomy which is briefly mentioned in Section III-D.

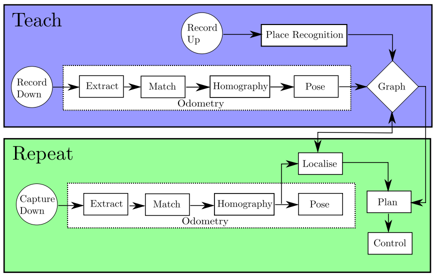


Fig. 2. An overview of the key processing steps in our system, with some detail regarding our particular choice of implementation. However, the framework is universal, with the central graph database granting read and write access to both mapping and localisation processes, allowing for large network representations of warehouse environments to be grown, over which space the robot can robustly localise and traverse autonomously.

The coordinate frames used in our system are shown in Figure 3. The robot frame $\underrightarrow{\mathcal{F}}_{R_k}$ and camera frame $\underrightarrow{\mathcal{F}}_{C_k}$ are related by a static rigid transformation $\mathbf{T}_{R,C}$ which is obtained by measuring the downward pitch, β, and position of the sensor relative to the vehicle centre of mass.

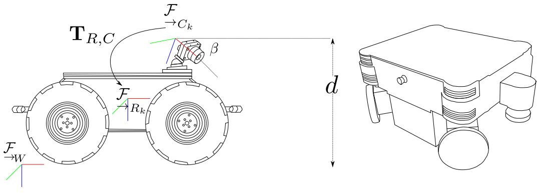

$\mathbf{T}_{R,C}$
$\underrightarrow{\mathcal{F}}_{C_k}$
β
d
$\underrightarrow{\mathcal{F}}_{R_k}$
$\underrightarrow{\mathcal{F}}_{W}$

Fig. 3. Clearpath Husky A200 and a Neobotix MP-70 robotic platforms used in our research, sole Pointgrey Firefly sensor used for navigation, and relevant coordinate frames (Red x, green y, and blue z constitute a North-East-Down frame convention). Lasers are only used for obstacle avoidance.

A. Topological Framework

Cameras are far from driftless global exteroceptive sensors, and can suffer inconsistency introduced by the attempt to solve in a global metric frame [15]. We thus strongly advocate the use of topological maps in the form of networks of interconnected places, an example of which is shown in Figure 4. Each element in the sets of map nodes, M, and query nodes, Q, must store appropriate representations of distinct places in the environment, while directed edges between nodes store an invertible relational operator, ${}^{i}\phi_j = {}^{j}\phi_i^{-1}$, between places, which must facilitate composition $\Gamma({}^{i}P_n) = {}^{i}\phi_j \oplus {}^{j}\phi_k + \ldots + \oplus^{m}\phi_n$, along a path $P = \{{}^{i}\phi_j, {}^{j}\phi_k, \ldots {}^{m}\phi_n\}$ of connected places. These compositions are not required to be accurate for far-flung i and n as we require only locally accurate place-to-place relations. We also require also the existence of a norm $|{}^{i}\phi_j| = |{}^{j}\phi_i| \geq 0$ to represent the "closeness" of two places. The graph is bidirectional in practice, to facilitate ease in implementation of a graph search, Ψ, over the space of this relational operator.

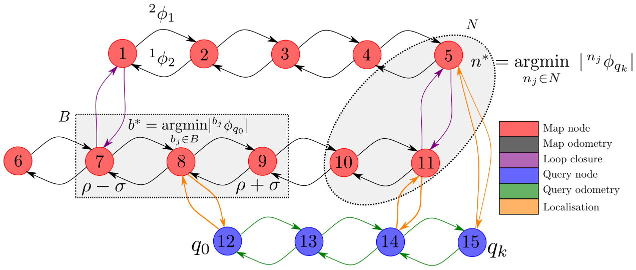


Fig. 4. Our topological maps are implemented as graph structures stored within a database. Nodes store a representation of distinct places, which may be a visual image, laser scan, etc. Edges indicate the connectedness of the environment on a local scale and store an appropriate relation between places, which may be a fundamental or essential matrix, metric pose, wheel odometry, etc. Localisation within the topological map is limited to optimisation within local graph neighbourhoods. This restricted approach is efficient, is facilitated by the success of recent localisations, and is made possible by the ability of the localiser to amend the underlying graph representation of the world.

This graph representation for maps lends itself well to both linear chains and topologically connected maps that have been amended with loop closures. The section to follow describes our particular choice of place representation and relational operator.

B. Mapping

We make visual sensors our modality of choice due to the low cost of highly dense information available as well as their suitability for place recognition. Despite the significant success demonstrated in the robotics community with stereoscopic vision [3], we choose not to use a sensor which relies on the appearance of the shelving areas, as they are subject to weekly if not daily variation during the rotation of stock in a warehouse. Stereo cameras can also be susceptible to narrow canyon-like environments in which particular features are not visible from many positions the robot finds itself in and race out at the camera at the sides of the image (which are subject to significant distortion). We thus choose to track the texture on the floor in front of the robot using a monocular camera and use the relational operator ${}^{i}\phi_j = \{\mathbf{H}_{i,j}, \mathbf{T}_{i,j}\}$ consisting of a robustly estimated homography and associated (up to scale) relative pose, as described in the following sections. Naturally, compositions take the form of matrix multiplications $\Gamma({}^{i}P_n) = {}^{i}\phi_j {}^{j}\phi_k \ldots {}^{m}\phi_n$.

1) Robust Homography Estimation: Figure 5 shows the camera motion and projection model used in our system. Adapting the notation of [16], consider two images $\mathbf{x}_{k-1}, \mathbf{x}_k \in \mathbb{R}^3$ (at subsequent time intervals) of a 3-D point $\mathbf{X}^{\pi}$. $\mathbf{X}^{\pi}$ is situated on a plane π that is parametrised by its normal $\mathbf{n}^{\pi}$ and the perpendicular distance to the centre of the first camera view, d.

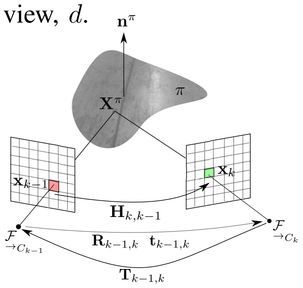

$\mathbf{n}^{\pi}$
$\mathbf{X}^{\pi}$
π
$\mathbf{x}_{k-1}$
$\mathbf{x}_k$
$\mathbf{H}_{k,k-1}$
$\underset{\rightarrow C_{k-1}}{\mathcal{F}}$
$\underset{\rightarrow C_k}{\mathcal{F}}$
$\mathbf{R}_{k-1,k}$ $\mathbf{t}_{k-1,k}$
$\mathbf{T}_{k-1,k}$

Fig. 5. Camera projection and motion model. Two views of the same planar surface are related by a homography that is induced by the plane.

The image points are related by a planar-induced homography $\mathbf{H}_{k,k-1}$ such that $\mathbf{x}_k = \mathbf{H}_{k,k-1}\mathbf{x}_{k-1}$ (a consequence of the epipolar constraint). We choose to detect and match points of interest across frames using SURF features [17], due to their robustness to viewpoint changes (the robot does not revisit locations with exactly the same attitude) and speed of implementation, yet our framework is easily adaptable for swapping this subsystem out. Nevertheless, given a set of such correspondences $\mathbf{x}_{k-1}^{j} \leftrightarrow \mathbf{x}_k^{j}, j = 1, 2, \ldots, n$ $(n \geq 4)$ from frame-to-frame sparse feature point matching it is possible to construct a system of equations

$$\mathbf{A}\mathbf{h}_{k,k-1} = \mathbf{0} \in \mathbb{R}^{3n} \tag{1}$$

where $\mathbf{A} = [\mathbf{a}^1, \ldots, \mathbf{a}^n] \in \mathbb{R}^{3n \times 9}$ and each $\mathbf{a}^j = \mathbf{x}_{k-1}^{j} \otimes \widehat{\mathbf{x}_k^{j}} \in \mathbb{R}^{9 \times 3}$. The row-major version of the homography, $\mathbf{h}_{k,k-1} \in \mathbb{R}^9$ can be found as the eigenvector of $\mathbf{A}^T\mathbf{A}$ corresponding to the minimum eigenvalue. To robustly estimate the homography, this estimation is wrapped in a RANSAC process [18] and the homography with the largest set of inliers is retained. This robustly estimated homography alone is sufficient to enable localisation within our topological framework, as shown in Section III-C. Motion planning and control, however, require a euclidean characterisation of the relationship between places, which is described in the following section.

2) Pose Decomposition: It can be shown that the relative pose between the subsequent frames $\mathbf{T}_{k-1,k}$, parametrised by a rotation $\mathbf{R}_{k-1,k}$ and translation $\mathbf{t}_{k-1,k}$ is related to the homography as

$$\mathbf{H}_{k,k-1} = \mathbf{R}_{k-1,k} + \frac{1}{d}\mathbf{t}_{k-1,k}\mathbf{n}^{\pi T} \tag{2}$$

The singular-value decomposition (SVD) of the calibrated homography $\mathbf{K}^T\mathbf{H}_{k,k-1}\mathbf{K} = V\Sigma V^T$, where the camera intrinsics are encoded in $\mathbf{K} \in \mathbb{R}^3$, gives us access to a set of four possible solutions for the motion (up to scale) between the two frames and the plane normal $\{\mathbf{R}_{k-1,k}, \frac{1}{d}\mathbf{t}_{k-1,k}, \mathbf{n}^{\pi}\}_i, (i = 1, 2, 3, 4)$. The candidate solutions are constructed using the singular values and vectors encoded in Σ and V – the reader is referred to [16] for more detail. Nevertheless, in our system, we disambiguate these solutions by applying a host of constraints:

- $\langle \mathbf{n}^{\pi}, [0, 0, 1]^T \rangle \leq \alpha_{\mathbf{R}}$
- $\langle \mathbf{R}_{k-1,k}\mathbf{n}^{\pi}, [0, 0, 1]^T \rangle \leq \alpha_{\mathbf{R}}$
- $\langle \mathbf{n}^{\pi}, \mathbf{t}_{k-1,k} \rangle \leq \alpha_{\mathbf{t}}$

which take into account the maximum angle that the normal can make with the plane, $\alpha_{\mathbf{R}}$, as well as the maximum translation, $\alpha_{\mathbf{t}}$, we expect frame-to-frame.

To avoid the degenerate case of a stationary robot, we only add nodes and edges to the graph when the translation between two keyframes, $\mathbf{T}_{f_2,f_1} = \prod_{k=f_1+1}^{f_2} \mathbf{T}_{k-1,k}$ exceeds a threshold $\xi_{\mathbf{t}}$ such that $|\mathbf{t}_{f_2,f_1}| \geq \xi_{\mathbf{t}}$. A similar threshold $\xi_{\mathbf{R}}$ is applied to the rotation the robot undergoes. Spatially ordering the map in this way allows the robot to repeat trajectories at any speed.

3) Geometric Properties of the Homography: Consider two consecutive images I_{k-1} and I_k related by a homography such that $I_k(\mathbf{x}_k) = I_k(\mathbf{H}\mathbf{x}_{k-1}) \simeq I_{k-1}(\mathbf{x}_{k-1}) \forall \mathbf{x}_k \subseteq I_k, \mathbf{x}_{k-1} \subseteq I_{k-1}$. Let the warped version of an image be $\tilde{I} = \mathbf{H}I$. Let us define geometric properties of this warping, encoded by the homography. First consider the overlap of two consecutive images after applying the robust homography estimation

$$\mathrm{h}_{\mathrm{area}}(\mathbf{H}_{k,k-1}) = \tilde{I}_{k-1} \cap I_k \tag{3}$$

Second, consider the discrepancy between image coordinates $\mathbf{x}_k^T \mathbf{H}_{k,k-1} \mathbf{x}_{k-1}$ and define a distance metric

$$\mathrm{h}_{\mathrm{distance}}(\mathbf{H}_{k,k-1}) = 2\eta^2 \left(\sqrt{1 + \frac{\mathbf{x}_k^T \mathbf{H}_{k,k-1} \mathbf{x}_{k-1}}{\eta^2}} - 1\right) \tag{4}$$

Where the scaling factor η is called the Huber regularisation threshold [19]. These functions are candidates for the norm of the relational operator $|{}^{i}\phi_j|$ which will both be used to represent “closeness” of places in a local neighbourhood during localisation.

C. Localisation

We can localise within graphs adhering to the topological framework by a simple local search and optimisation as originally illustrated in Figure 4. Our localisation begins by a brute-force style search over a very limited portion of the graph. Let $B = \{b_j \mid \rho - \sigma \leq j \leq \rho + \sigma\}$ be the set of all map nodes within the database in a chain of length σ around a seed node, ρ, provided by the user. We locate the first query, q_0, at the map node closest to it in the space of geometric distances

$$b^* = \underset{b_j \in B}{\mathrm{argmin}}\ \mathrm{h}_{\mathrm{distance}}(\mathbf{H}_{b_j,q_0}) \tag{5}$$

The system proceeds by performing the incremental odometry estimation of III-B. We illustrate our localisation by choosing a simple breadth first search strategy $\Psi = \Psi_{BFS}$, yet other methods are easy to integrate. If q_k is the latest query node added to the database, let $N = \{n_j \mid \Psi_{BFS}(q_k, n_j) \leq d_{max}\}$ be the set of all map nodes reachable from it by the search to a maximum depth, d_{max}. The chained homography from the query to each reachable node is wrapped up in the search such that each node n_t discovered by expanding a parent n_s is assigned a *weak* estimate $\hat{\mathbf{H}}_{q_k,n_t} = \hat{\mathbf{H}}_{q_k,n_s}\mathbf{H}_{n_s,n_t}$. Similarly to the initialisation above, a potential localisation result is found by minimising in the space of distances on this search estimate

$$n^* = \underset{n_j \in N}{\mathrm{argmin}}\ \mathrm{h}_{\mathrm{distance}}(\hat{\mathbf{H}}_{n_j,q_k}) \tag{6}$$

However, we impose a strict requirement on bonafide localisation results such that they must satisfy a further geometric constraint, κ, on their consequent overlap area: $\mathrm{h}_{\mathrm{area}}(\hat{\mathbf{H}}_{n^*,q_k}) \geq \kappa$. If the localisation candidate passes these initial constraints, robust pose estimation can follow.

At this point however, we guide the homography estimation. If $\tilde{I}_{q_k} = \hat{\mathbf{H}}_{n^*,q_k} I_{q_k}$ is the query image warped under the action of the search estimate homography, matching finds correspondences $\tilde{\mathbf{x}}_{q_k} \leftrightarrow \mathbf{x}_{n^*}$ in the frame of the candidate, before unwarping the detected features $\mathbf{x}_{q_k} = \hat{\mathbf{H}}_{n^*,q_k}\mathbf{x}_{n^*}$ and proceeding with estimation. Ensuring features have a similar orientation in frames of comparison provides an additional level of protection against spurious matching between frames that are captured some time apart, and not in an incremental fashion.

We impose one last constraint on the candidate, and require that the percentage of inliers in the RANSAC estimation of the homography exceeds some threshold ψ. The localisation result is then used to amend the graph such that an edge is created between it and the query node.

The algorithm is divorced from any probabilistic framework, and thus lacks introspection. We thus consider it critical to the success of the algorithm that each localisation candidate passes a multitude of tests such as this, to prevent erroneous amendments of the graph (which would have the incremental effect of degrading all further searches over the graph). In the face of so many checks for robustness, localisation occasionally stalls. In order to avoid missing too many opportunities to create meaningful edges in the graph (problematic if the robot has failed to localise for long enough such that the maximum depth of the graph search prevents it from ever recovering), we perform another very modest brute-force recovery by searching over the set of map nodes $F = \{f_j \mid \varrho - \sigma \leq j \leq \varrho + \sigma\}$ around an estimate ϱ obtained from a constant velocity model. Our results will show that this contingency need only be resorted to in the case of extreme lens glare or unmodeled robot kinematics. Upon recovery, localisation proceeds in a well-behaved manner.

D. Planning and Control

If the user positions the robot close to a start node q_{start} and requests that it travel to a destination q_{goal}, let $P = \{p_j \mid p_j \in \Psi_{BFS}(q_{start}, q_{goal})\}$ be the set of nodes

Parameter	Value	Insight
d	0.6	Measurable static camera height [m]
β	0.735	Measurable downward pitch [rad]
$\alpha_{\mathbf{R}}$	0.342	Approximately perpendicular plane normal, 70 [°]
$\alpha_{\mathbf{t}}$	0.100	Varies with frame rate and maximum speed
$\xi_{\mathbf{t}}$	0.5	Varies with frame rate and maximum speed
$\xi_{\mathbf{R}}$	0.1	Varies with frame rate and maximum angular speed
η	100	$\mathrm{h}_{\mathrm{distance}}$ should vary smoothly
σ	20	Assume fairly reliable seed by operator
d_{max}	25	Must be larger than σ yet still modest
κ	0.6	Strict localisation requirement
ψ	8	At least eight points required for solution
θ	3	Drift within small segments is negligible [m]

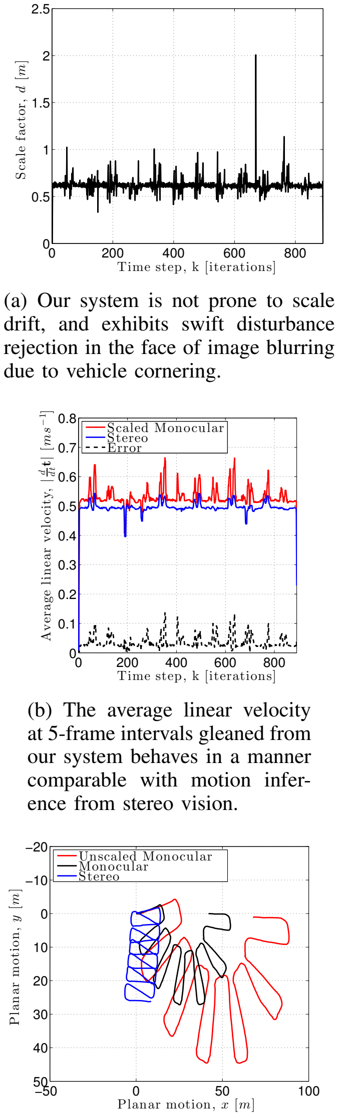
TABLE I

TABLE OF PARAMETERS AND JUSTIFICATION FOR THEIR CHOICE.

along the shortest sequence of edges linking q_{start} and q_{goal}. $P = \cup_{s=1}^{n_s} P_s$ is segmented into n_s groups P_s by chaining relative poses and ensuring that the robot does not travel too far within each segment, such that $|\mathbf{t}_{P_s(1),P_s(n_{P_s})}| \leq \theta$. The cubic hermite spline method of [20] is used to interpolate between points along each segment and generate a smooth trajectory for the robot to follow, as well as to ensure that there are no spatial or velocity discontinuities at the confluence of adjacent segments. Implementing the work of [21], the robot is controlled to follow these planned trajectories whereby angular corrections (limited to a maximum magnitude) are made to the heading while the robot is travelling at a speed scheduled by a simple proportional-integral (PI) control strategy.

IV. EXPERIMENTS

The experiments presented here were conducted on two mobile robotic platforms particularly suited to warehousing, shown in Figure 3: A Clearpath Husky A200 and a Neobotix MP-70. Offline results were generated on a machine with 2.3 GHz Intel Core i7 processing power, 8 GB RAM, 1600 MHz DDR3. Images were captured by a Pointgrey Firefly at 60 fps and a resolution of 752×480. The camera intrinsic calibration was obtained by the OCamCalib implementation of [22], which was used to undistort incoming images. Furthermore, some robustness to lighting changes was available by normalising each image with the histogram equalisation method of [23]. A summary and explanation of the choice of parameter values is shown in Table I.



(a) Our system is not prone to scale drift, and exhibits swift disturbance rejection in the face of image blurring due to vehicle cornering.

(b) The average linear velocity at 5-frame intervals gleaned from our system behaves in a manner comparable with motion inference from stereo vision.

(c) Compared to stereo odometry, ground plane trajectories yielded by our system are appear warped on a global scale, yet are locally accurate.

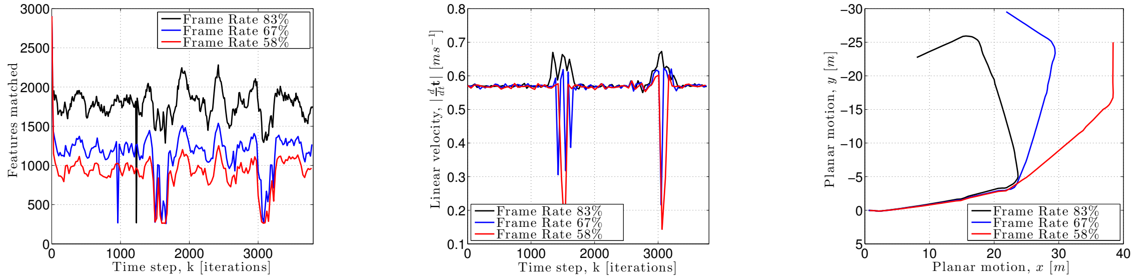
Fig. 6. In these mapping experiments, we illustrate the robustness and accuracy of our motion inference and consequently our maps on a local scale

A. Mapping

To test the local consistency of the maps generated by our system, Figure 6(a) shows an evolving scale factor calculated against state-of-the-art stereo visual odometry [24] as $d = \frac{|\mathbf{t}_{stereo}|}{|\mathbf{t}_{mono}|}$, which is equivalent to the height of the camera as originally shown in Figure 3. The scale factor is not prone to drift, and shows graceful disturbance rejection in recovering after taking corners and experiencing image blur.

Indeed, Figure 6(c) shows a map built over 250 m^2 of a warehouse shelving area. While prone to accumulate errors over time, the scaled trajectory achieved with our system is locally accurate, due to a fairly precise inferred vehicle velocity shown in Figure 6(b). To obtain a notion of accuracy at a local scale, we propose the measurement of the average error in inferred velocity $e_{speed} = \frac{1}{N}\sum|v_{mono} - v_{stereo}|$ which for this particular experiment amounts to $e_{speed} = 0.0353$ m/s, or only 7.1326% of the actual speed, which we will show is sufficient for localisation over long distances.

We can produce maps that are locally accurate even over floor surfaces that are relatively textureless. To illustrate this, we show in Figure 7(c) the modest degeneration of the map quality in the face of significant frame rate reductions, which starves the homography estimation of features to match on a frame-to-frame basis as shown in Figure 7(a). We argue and will show that these maps are sufficiently accurate for robust localisation and refer the reader to Figure 7(b) in which the motion recoverable from our pose estimation is shown in the form of steady-state linear robot velocity to be immune to harsh frame rate reductions.



(a) The ebbing quantity of candidate features for matching disappear when consecutive floor images lose significant overlap.

(b) The linear velocity inferred from our pose estimation remains constant in the face of reasonable reductions in floor imagery overlap.

(c) We retain local accuracy in mapping as the homography estimation is starved of candidate features for matching.

Fig. 7. In these mapping experiments, we illustrate the robustness and accuracy of our motion inference and consequently our maps on a local scale.

B. *Localisation*

We campaign vehemently for the disregard of difficult to obtain globally accurate maps, and argue that our locally accurate odometry coupled with robust localisation is sufficient to deploy autonomous mobile robots in real-world environments. These localisation experiments are thus the critical results of this paper. We begin our investigation with a motivating example by showing in Figure 8(b) localisation capability in a highly textured office environment. We are capable of uninterrupted successful localisation around the entirety of the office. In Figure 8(c) we exploit the topological links between the query and map trajectories to compare the inferred vehicle velocity from dead reckoning (frame-to-frame odometry estimation along the query trajectory) with the rate at which the robot must have been travelling to have localised at waypoints along the map trajectory, and show them to be complementary. We then explore localisation in our warehouse domain of interest, showing similar consistency in both localisation performance as evident in Figure 8(e) and Figure 8(h) and motion accuracy as illustrated in Figure 8(f) and Figure 8(i). Our system is capable of such performance even in the face of surfaces which are dramatically illuminated or suffer from a dearth of features for matching, Figure 8(d). We go even further to show the extensibility of this work to dynamically lit, and busy outdoor environments. We drove the robot around a city block, Figure 8(j), and were able to successfully localise for several hundred metres, Figure 8(k). Furthermore, the system was consistent in its recovery of the velocity of the vehicle, Figure 8(l), which tapered off towards the end of each outing (the battery, replaced each outing, depleted).

Our chief concern with regards to localisation capability is the anticipated period over which the robot can be expected to travel without successful localisation, vulnerable to errors in wheel odometry. To this end, Figure 9 shows the probability of the robot travelling further than a certain distance $P(X \geq x) = \sum p(x)$ before recovering localisation. Indoor, highly textured, and smoothly surfaced environments such as our office exemplar never require localisation recovery. Yet even in sparsely textured warehouses and kinematically unconstrained sidewalks, it is important to note that our system never requires the robot to travel blind for more than 1.4 m, and will quite probably relocalise much sooner than that.

C. *A simple teach and repeat experiment*

The suitability of our topological framework to enabling autonomy is illustrated in Figure 10 in which a robot is taught a trajectory by manual control, and is able to repeat it automatically. As the driving inputs for the path following control, it would not be appropriate to compare trajectories obtained with our system via monocular imagery. Instead, as effective ground truth, we compare manually piloted and autonomously driven trajectories obtained from stereo visual odometry [24]. The topological framework and localisation therein is such that amendments are made to the underlying graph structure, which allows the robot to grow a network of paths which it can use for autonomous missions. Figure 10(c) shows an example of localisation before driving beyond the extent of a taught map. The robot is then able to automatically traverse the entire environment in an order that it did not experience.

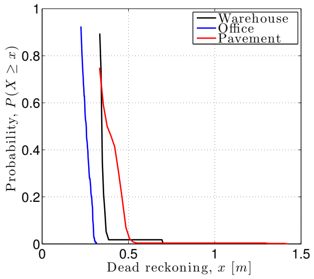
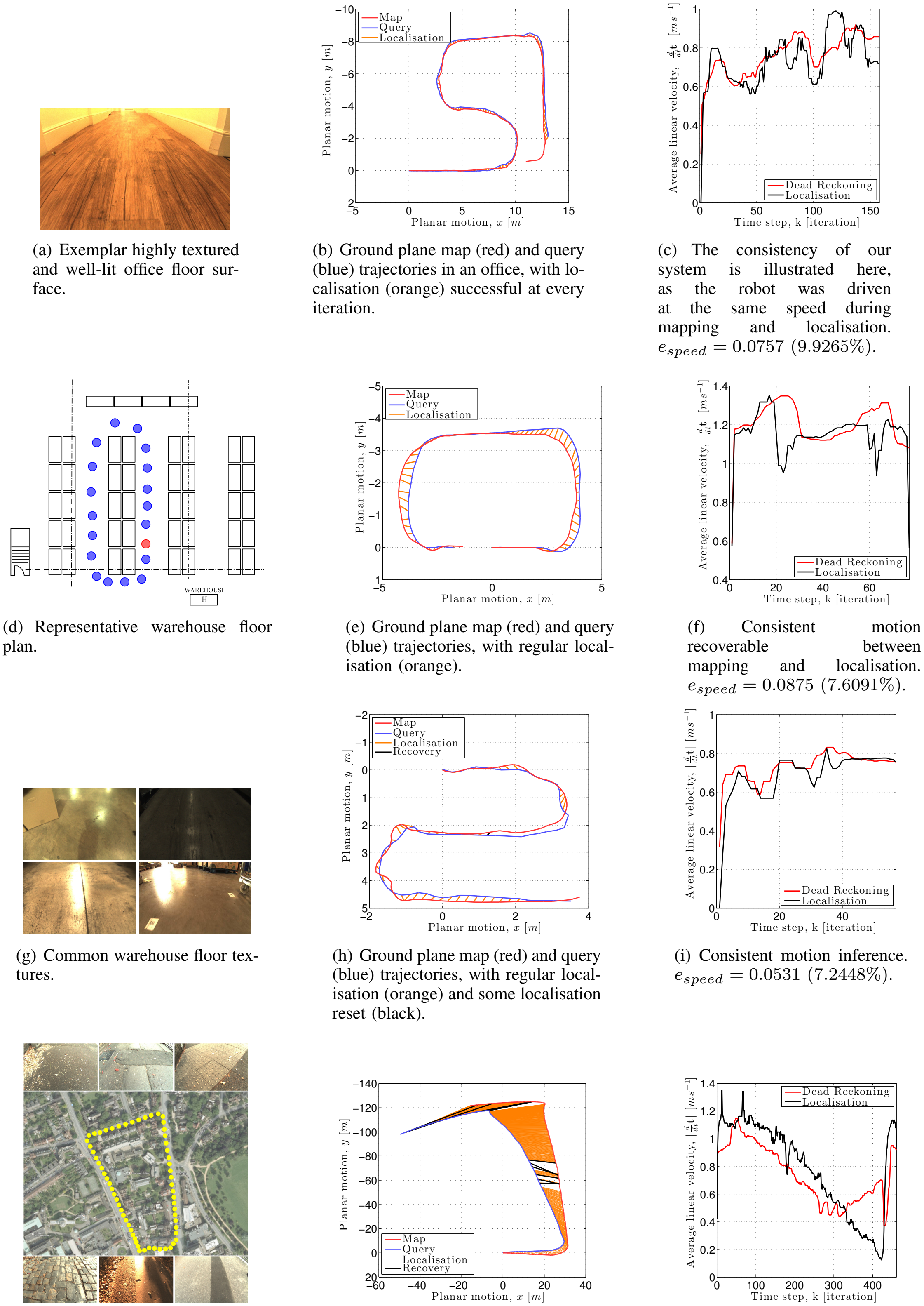


Fig. 9. A cumulative distribution of the probability of travelling blind for more than a certain distance before resuming normal localisation behaviour, from which it is clear our topological framework and localisation within local graph neighbourhoods is robust to localisation failures even in the most challenging environments.

V. CONCLUSION

We have presented a framework for topological navigation for robots operating in warehouse environments equipped with only downward-facing monocular cameras. Mapping and localisation are purely topological and require no knowledge of the global properties of the environment. We have experimentally validated our approach within three significantly different environments against state-of-the-art stereo odometry and shown it to be consistent in motion



(a) Exemplar highly textured and well-lit office floor surface.

(b) Ground plane map (red) and query (blue) trajectories in an office, with localisation (orange) successful at every iteration.

(c) The consistency of our system is illustrated here, as the robot was driven at the same speed during mapping and localisation. $e_{speed} = 0.0757\ (9.9265\%)$.

(d) Representative warehouse floor plan.

(e) Ground plane map (red) and query (blue) trajectories, with regular localisation (orange).

(f) Consistent motion recoverable between mapping and localisation. $e_{speed} = 0.0875\ (7.6091\%)$.

(g) Common warehouse floor textures.

(h) Ground plane map (red) and query (blue) trajectories, with regular localisation (orange) and some localisation reset (black).

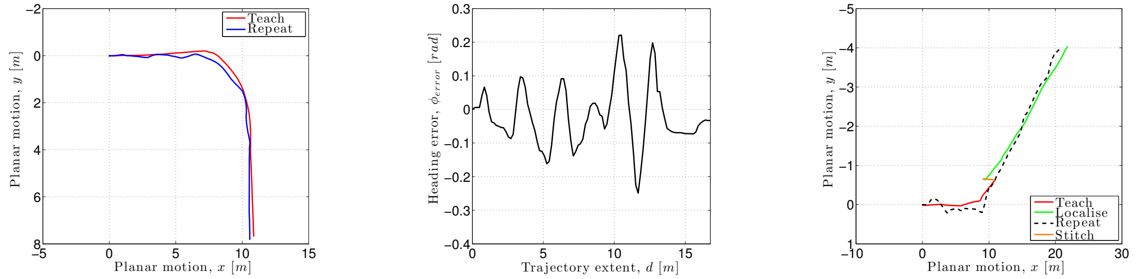
(i) Consistent motion inference. $e_{speed} = 0.0531\ (7.2448\%)$.

(j) Satellite view of the route driven along the pavement around a city block and representative surface scenery, which is often dramatically lit or rather featureless.

(k) Every so often, localisation fails in the face of changes in lighting or unmodelled vehicle movement. Our system is able to gracefully recover (black) and can resume normal localisation. We successfully localised (blue) for more than 200 metres.

(l) On both outings, the depleting power supply to the robot caused it to taper off in speed, which our system is capable of representing and tracking. $e_{speed} = 0.1780\ (22.5481\%)$.

Fig. 8. Demonstration of indoor and outdoor localisation along surfaces of varying texture and driveability, for up to several hundred metres.



(a) Using frame-to-frame odometry estimated by our system, we are able to control the robot to automatically repeat (blue) a trajectory it is manually piloted along (red). Both trajectories are computed by stereo odometry, and are as such decoupled from the control inputs

(b) While poorly tuned, it is clear that the controller is making effective use of the heading error fed to it by our system, leading to an oscillatory yet stable heading error profile over the course of the repeated trajectory.

(c) Our system allows us to leverage multiple experiences (red and green) of overlapping regions of the same workspace, and enables the robot to automatically repeat (black) two segments taught at distinct times and linked (orange) as a byproduct of our localisation.

Fig. 10. The navigation system we advocate lends itself well to enabling autonomy, illustrated here by a very simple teach and repeat experiment in a parking lot.

inference and robust to localisation failures. Pedagogically, and experimentally, our method is illustrated by a specific set of algorithmic and hardware choices, yet we assert that our framework is agnostic to the choice of sensor modality. Finally, we provided a short demonstration of the suitability of our method to enabling autonomy of warehouse robots, and is particularly useful for growing large and comprehensive network representations of those environments.

VI. ACKNOWLEDGEMENTS

Matthew Gadd is supported by the FirstRand Laurie Dippenaar and Oppenheimer Memorial Trust scholarships. Paul Newman is supported by EPSRC Leadership Fellowship EP/J012017/1.

REFERENCES

[1] J. P. van den Berg and W. Zijm, “Models for warehouse management: Classification and examples,” *International Journal of Production Economics*, vol. 59, no. 1, pp. 519–528, 1999.

[2] R. D’Andrea, “Guest editorial: A revolution in the warehouse: A retrospective on kiva systems and the grand challenges ahead,” *Automation Science and Engineering, IEEE Transactions on*, vol. 9, no. 4, pp. 638–639, 2012.

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

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

[5] J. Marshall, T. Barfoot, and J. Larsson, “Autonomous underground tramming for center-articulated vehicles,” *Journal of Field Robotics*, vol. 25, no. 6-7, pp. 400–421, 2008.

[6] P. Krüsi, B. Bücheler, F. Pomerleau, U. Schwesinger, R. Siegwart, and P. Furgale, “Lighting-invariant adaptive route following using iterative closest point matching,” *Journal of Field Robotics*, 2014.

[7] R. A. Brooks, “Visual map making for a mobile robot,” in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2. IEEE, 1985, pp. 824–829.

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

[9] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, “Monocular vision for mobile robot localization and autonomous navigation,” *International Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.

[10] A. M. Zhang and L. Kleeman, “Robust appearance based visual route following for navigation in large-scale outdoor environments,” *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 331–356, 2009.

[11] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki, “Robot homing by exploiting panoramic vision,” *Autonomous Robots*, vol. 19, no. 1, pp. 7–25, 2005.

[12] G. Blanc, Y. Mezouar, and P. Martinet, “Indoor navigation of a wheeled mobile robot along visual routes,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 3354–3359.

[13] C. McManus, P. Furgale, B. Stenning, and T. D. Barfoot, “Lighting-invariant visual teach and repeat using appearance-based lidar,” *Journal of Field Robotics*, vol. 30, no. 2, pp. 254–287, 2013.

[14] B. E. Stenning, C. McManus, and T. D. Barfoot, “Planning using a network of reusable paths: A physical embodiment of a rapidly exploring random tree,” *Journal of Field Robotics*, vol. 30, no. 6, pp. 916–950, 2013.

[15] G. Sibley, C. Mei, I. Reid, and P. Newman, “Planes, trains and automobilesautonomy for the modern robot,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 285–292.

[16] Y. Ma, *An invitation to 3-d vision: from images to geometric models*. springer, 2004, vol. 26.

[17] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 404–417.

[18] M. A. Fischler and R. C. Bolles, “Random sample 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.

[19] P. Heise, S. Klose, B. Jensen, and A. Knoll, “Pm-huber: Patchmatch with huber regularization for stereo matching,” in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2360–2367.

[20] H. Mettke, “Convex cubic hermite-spline interpolation,” *Journal of Computational and Applied Mathematics*, vol. 9, no. 3, pp. 205–211, 1983.

[21] G. M. Hoffmann, C. J. Tomlin, D. Montemerlo, and S. Thrun, “Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing,” in *American Control Conference, 2007. ACC’07*. IEEE, 2007, pp. 2296–2301.

[22] D. Scaramuzza, A. Martinelli, and R. Siegwart, “A toolbox for easily calibrating omnidirectional cameras,” in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 5695–5701.

[23] G. Finlayson, S. Hordley, G. Schaefer, and G. Yun Tian, “Illuminant and device invariant colour using histogram equalisation,” *Pattern recognition*, vol. 38, no. 2, pp. 179–190, 2005.

[24] F. Fraundorfer and D. Scaramuzza, “Visual odometry: Part ii: Matching, robustness, optimization, and applications,” *Robotics & Automation Magazine, IEEE*, vol. 19, no. 2, pp. 78–90, 2012.