Continually Improving Large Scale Long Term Visual Navigation of a Vehicle in Dynamic Urban Environments

Winston Churchill and Paul Newman

***Abstract*— This paper is about long term navigation in dynamic environments. In previous work we introduced a framework which stored distinct visual appearances of a workspace, known as experiences. These are used to improve localisation on future visits. In this work we introduce a new introspective process, executed between sorties, thats aims by careful discovery of the relationships between experiences, to further improve the performance of our system. We evaluate our new approach on 37km of stereo data captured over a three month period.**

I. Introduction

This paper is concerned with improving the performance of a long term navigation system. Before we present the novel contributions of this paper in Section III, we first recap our previous work on experience-based navigation [1] in Section II. Implementation details and system performance are briefly discussed in Section IV, results are presented in Section V and related work is covered in Section VI.

II. Background: Experience-Based Navigation

A. Motivation

For robotic systems to achieve truly long-term autonomy they must be able to deal with dynamic workspaces. Changes to an environment can happen for a variety of reasons and at different rates. Moving objects, such as people and cars can cause sudden structural change, the trajectory of the sun produces different lighting conditions over the period of a day and the passage of the seasons results in a long term change in appearance.

One area that is affected by this problem is Visual Odometry (VO), in which it is often implicitly assumed that changes in the scene appearance are solely as a result of the ego-motion of the camera. The majority of traditional navigation approaches build a single map on the initial visit and hope this is sufficient for future use. The assumption being that the environment will not change appearance drastically, and thus it is possible to localise using this single snapshot.

Features may be added or updated during future visits. However should the system continue to accumulate features for all time? Given enough time, in such an approach the map will become bloated with features, many of which will have no relevance to each other. Also what should happen if the workspace has drastically changed appearance - for example comparing a map created on a bright sunny afternoon with a misty morning? We propose instead of attempting to fuse data for all time into a single frame of reference, to allow each experience of the world to remain independent, but to capture their topological relationships in a graph.

The authors are with the Oxford University Mobile Robotics Group {winston,pnewman}@robots.ox.ac.uk

Before we discuss how our experienced base navigation works we briefly introduce some key terminology. Firstly what our VO system produces, secondly what an experience is, and finally how localisers operate over experiences.

B. Visual Odometry

Visual Odometry (VO) is a well understood problem and several systems have previously been demonstrated [2], [3]. We now briefly introduce the notation of ours, and what we get as output. For a sequence of stereo frames $\mathcal{F}^{\mathbf{k}} = \{\mathcal{F}_0, \ldots, \mathcal{F}_k\}$, taken at times k, our VO system produces a corresponding sequence of nodes, n_k. A 6 degree-of-freedom transformation t_k links sequential nodes n_{k-1} to n_k,

$$t_k = [x, y, z, \theta_r, \theta_p, \theta_q]^T \tag{1}$$

where θ_r, θ_p and θ_q are roll, pitch and yaw respectively.

Frame to frame motion estimation is achieved by matching the image descriptors of 3D landmarks. Landmarks are created when previous ones cannot be matched to the current frame $\mathcal{F}_k$. When this happens, the new landmarks are stored in the node, n_k, from which they were first observed. A landmark, $l_{i,k}$, is described as follows:

$$l_{i,k} = [x, y, z]^T \tag{2}$$

where i is a global landmark index and k denotes which node the 3D vector is relative to. Finally, every landmark observed in $\mathcal{F}_k$ is recorded in a list in n_k. Many of these landmarks will be stored in other nodes - principally the one from which they were first observed.

Landmarks can be stored in one frame, but observed in another. For the estimation process it is required that a landmarks position can be expressed relative to different nodes. To achieve this we define the following operation, ${}^{p}\Pi_{q}$, which transforms a landmark from frame p to frame q.

$$l_{*,q} \leftarrow {}^{p}\Pi_{q}(l_{*,p}) \tag{3}$$

C. Experiences

An experience is simply a sequential subset of the output from the VO system. We denote each experience as ${}^{j}\mathcal{E}$. The conditions which trigger the saving of an experience are discussed in Section II-E.

Concretely, ${}^{j}\mathcal{E}$ is a sequence of nodes, connected via transformations and the set of all landmarks observed in the sequence. Individual nodes within an experience are specified

as ${}^{j}\mathcal{E}_m$. Note that once an experience has been saved, its internal data is never modified. No landmarks are added or updated, or pose estimates refined based on observations from future visits to the area.

D. Localisers

In our previous work we introduced the notion of a localiser which operates on an experience. Each experience is assigned a localiser, which computes the transformation from a node in its experience to the current stereo frame from the live sequence, $\mathcal{F}_k$. These transformations are computed in exactly the same way as the VO system computes the motion between two stereo frames. The only difference is that the landmarks do not come from the previous frame, $\mathcal{F}_{k-1}$, instead they come from the previous position in the experience. We do not attempt to modify an experience once it is stored, so a localiser can treat it as constant.

Localisers must be able to tell when they are "lost". This occurs when the live sequence of stereo frames cannot be matched against the current position in the associated experience. Given the live frame, each one produces a binary result indicating if localisation has been successful:

$$ {}^{j}\mathbf{L}(\mathcal{F}_k) = \begin{cases} 1 & \text{if localised} \\ 0 & \text{if lost.} \end{cases} \tag{4} $$

For every successful localiser of $\mathcal{F}_k$, each one can return the node $\mathcal{F}_k$ was closest to.

$$ {}^{j}\mathcal{E}_m \leftarrow {}^{j}\mathbf{L}() \tag{5} $$

If a localiser becomes lost, it does not attempt to re-localise to future images. It can be re-started if it receives outside help. One source of outside assistance is other localisers, operating on different experiences. How localisers share position information is introduced in the next section, increasing this interaction and improving its utility is the focus of this paper.

E. Experienced based navigation

We now explain how our previous work operates to achieve long term navigation in changing environments. While running we always perform VO on the live image stream. The question of whether this is saved as a new experience is a function of the result of the localisers running on the current experience set. If N or more localisers are successful, we believe our current representation of the local region to be sufficient and discard the VO output. However when the number of successful localisers falls below N, we create a new experience from the VO output. This continues until the number of successful localisers returns to N or above, when saving is stopped.

Experience creation is driven by the success or failure of localisation to prior experiences. This results in us naturally capturing the varying complexity of the world. In areas of high visual variation we store more experiences, while in regions that remain visually similar over time, we save relatively few experiences as our prior ones are sufficient for localisation. If we allow N to be greater than one, more experiences will be created, but we also become more resilient to localisation failures.

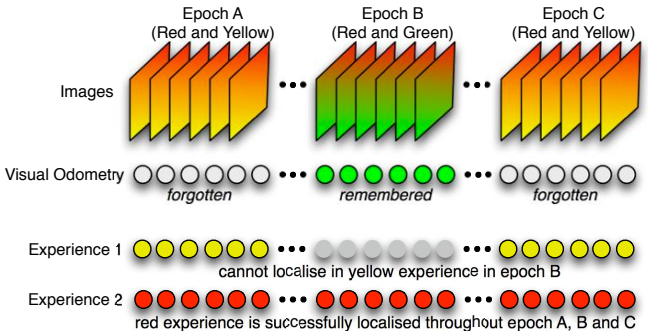

Fig. 1. An outline of our previous work. While traversing an environment a stereo camera produces a stream of frames. A visual odometry (VO) system consumes these to estimate motion and landmark positions. In this example, the decision to save or discard the VO output is a function of the ability to localise against experience 1 and 2. In epochs A and C the live frame stream is successfully localised against both experiences, meaning the VO output is not saved. However in epoch B only experience 2 is localised against the live frame. In this example we require the minimum number of successful localisers to be 2, so in this region a new experience is created from the VO output. This figure is reproduced from [1].

III. Stitching Experiences

A. Motivation

Ideally we strive to store the minimum number of experiences needed to represent an environment, a surplus leads to wasted computation and memory resources. We want just the right number to facilitate long term navigation.[1] This motivates us to fully exploit all the information in each and every experience, and also how they inter-relate. Failure to do so reduces our ability to re-localise within an experience, leading to an unnecessary genesis of a new experience.

The opportunity to exploit experiences optimally occurs naturally in two forms. The first is in starting a localiser to run on an experience, as every experience is not always relevant to the current position in the world. Consider a vehicle traveling from a relatively unchanging area to one of high visual variance, it needs to know when to activate the extra experiences related to that area. The second is when an experience temporarily becomes insufficient to explain the current visual feed (the localiser fails). For example a recently parked high sided vehicle totally obscures the scene as we pass. The experience will be relevant a short while later, at which point it needs to be re-started. As each experience is independent, global position estimation between experiences cannot be achieved by integrating the local relative transformations, as these are these are globally inaccurate.

As global position look-up is not possible, we look for other ways localisers can be started or re-started. One option would be to use an external loop closer such as FAB-MAP [4], where the live image is matched to a node in the saved

[1] We note that certain parts of the environment require more experiences than others, on account of greater visual variation.

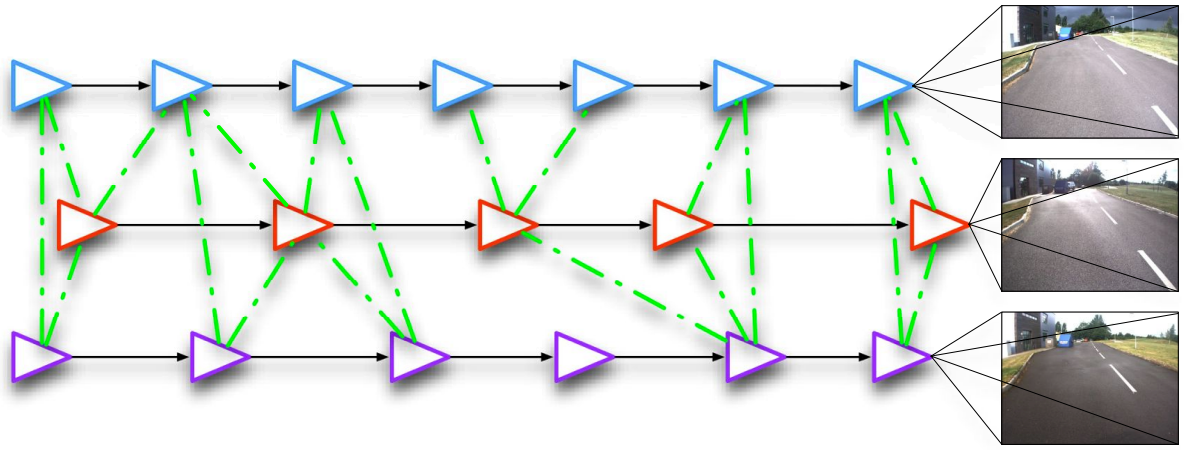

Fig. 2. Each experience can be represented as a graph, where the stereo frames are represented as nodes (shown as triangles) and metric transformation information describes how they are connected (indicated by the directed black arrows). The figure above shows three experiences, blue, red and purple. These can be formed into a single larger graph G, which contains all experiences. Edges can then be added between experiences (denoted as undirected dashed green lines) indicating the nodes refer to the same physical place in the world. This is demonstrated for the far right node in each experience. Each associated image is of the same place in the environment, so edges can be created between them. The focus of this work is how to increase the quality and quantity of these edges and what impact this has on localisation performance.

experience. However this is unlikely to be particularly successful due to the relatively low recall rate at high precision. Another approach would be to annotate experiences with metadata, such Global Positioning System (GPS) points, or higher-order descriptions such as known road junctions or buildings, however these are not always available and GPS often suffers from significant drift over time.

B. Experience Graph

As previously explained, an experience is a set of connected nodes. Consider if these were sub-graphs in a single large graph containing all experiences, G. Now assume we have a method of introducing an edge between two sub-graphs that indicates the two connected nodes observed the same physical place in the world. When a successful localiser arrives at a node with one of these edges, it can query the edge for which experience and location within that experience is at the other end. The localiser associated with the other experience can then be informed where to start (if the localiser is not currently active) or to restart (if the localiser is lost). Using this connected graph, localisers can inform each other when to start, and to aid each other when lost. In our previous work these inter-experience edges were referred to as "places".

The focus of this work is how the inter-experience edges of G can affect the performance of the localisation system. By introducing as many high quality edges as possible, we increase the information shared between experiences. These can then be used to aid localiser initialisation and restarting. We now present four methods for discovering the structure of G. The final two are extensions of our previous work. They are processes which run between vehicle outings. Their aim is to take any new experiences created from the previous outing, introduce the sub-graph formed from their nodes to G, and then look for opportunities to create edges between the newly inserted nodes and all other nodes (experiences).

C. Graph Structure Discovery

1) No Discovery: To demonstrate the importance of G's connectivity levels, in this approach we only allow inter-experience edges to be created which include the start of an experience. This enables experiences to be started if they are not active, but prevents the re-initialisation of lost experiences.

2) Live Discovery: This approach is the one presented in our previous work [1], where it is referred to as "places". Here edges are created when the live stereo stream is localised in more than one experience. This means G can only be changed when the vehicle is driving, and only in the regions covered by the vehicle on that trip. For example if the previous sortie included the car park, but it is not included on the next outing, the car park section cannot be connected to other experiences of that area. This dependancy on the live system to create edges is undesirable.

3) GPS Discovery: In this variant, for every experience created we also store the GPS position of every node. When an experience is created, for each new node, we find the nearest position in all other experiences via GPS. If the distance is less than some threshold[2], we introduce an edge connecting them.

4) Refined Discovery: In this final version we attempt to negate the drift that GPS suffers from over time. As our experiences are collected over several months, our data is affected by this drift. We achieve an initial match to other experience nodes using the same approach as GPS

[2]Different experiences refer to different areas in the world, and so may not be relevant to each other

Fig. 3. An overhead of the Begbroke site, including the two loops driven over the three month period. The outer loop was driven 47 times (thicker line). The inner loop (thinner line) was driven 6 times. The colour of the plot at each point indicates how many experiences are stored there. This figure is reproduced from [1].

Discovery. We then look to refine this estimate by matching the new node against a sequence of nodes surrounding the GPS suggestion, based on the same stereo frame to frame estimation techniques used in the VO system.

Given the new node n^* and a candidate node proposed from GPS, n_c, we take a small window of nodes either side of n_c, denoted $\{n_c\}_w$ and compute the transform from n^* to each node in the window. Next, assuming all transforms are valid, we compute the translation from n^* to each $\{n_c\}_w$. If we find a local minima which is not on the bounds, i.e. n^* really did pass by the window $\{n_c\}_w$, we take the candidate node in $\{n_c\}_w$ with the smallest translation to n^* to be the same place. The appropriate edge is then created in G.

IV. Implementation

Our Visual Odometry system used in this work operates on image frames from a stereo camera. It extracts points of interest from the images using the FAST corner extractor [5]. Robust feature matching between frames is achieved using Binary Robust Independent Elementary Features (BRIEF) descriptors [6]. Efficient second-order matching [7] is then used to refined successful matches to sub-pixel precision. The ego-motion estimation between two frames is achieved using a RANSAC [8] step followed by a least-squares optimisation to refine the solution. We assume the vehicle INS provides a ground truth for motion estimation. Given this, the relative transforms between frames, t_k, produced by the VO system have a mean error of $[-0.0093, -0.0041, -0.0420]$ meters and $[-0.0239, 0.0021, 0.0040]$ degrees and standard deviation of $[0.0225, 0.0245, 0.0155]$ meters and $[0.0918, 0.0400, 0.0383]$ degrees.

One benefit of keeping experiences independent, is that localisation of each one to the current live frame is also independent. This means localisation across multiple experiences can be performed in parallel. By using BRIEF descriptors to compute feature matches, we only require a CPU to achieve frame rate performance, while achieving a performance similar to Speeded Up Robust Features (SURF) [9]. SURF implementations can be made to run at frame rate but require a GPU to achieve this [10]. As each localiser requires access to fast feature matching, running multiple independent localisation modules in parallel would be difficult on a single GPU (the maximum on most computer systems), compared an implementation only requiring the CPU and using a multi-core machine.

V. Results

To evaluate our proposed localisation algorithm we repeatedly drove two partially overlapping loops around Begbroke Science Park. Over a three month period we collected 53 data sets, each one approximately 0.7km long. We drove the vehicle, Fig. 4, in a variety of weather conditions and times of day to capture the visual variation of the route.

Fig. 3 displays the two routes that were repeatedly driven along with the typical performance of the system. Predominately the outer loop (shown with a thicker line) was traversed, while the inner loop (thinner line) was completed on the final 6 runs. The number of experiences that are laid down at each point along the route are indicated by the intensity of the plot. Note that different regions require different number of experiences. The regions to the north and east require relatively few experiences for localisation as these areas are visually stable. In contrast, the area to the north west and south west corner demand significantly more experiences. The region to the north west over looks a car park, which exhibits daily fluctuation. The section of road in the south west corner is covered by overhanging tress. These cause strong shadowing effects, making localisation against previous experiences difficult.

To test how the connectivity of G affects localisation performance we evaluate the four variants of our approach outlined in Section III-C.

- *No Discovery* - inter-experiences edges only contain experience beginnings. Allows experiences to be started but not re-started once lost.
- *Live Discovery* - inter-experience edges created when the live VO system successfully localises in more than one experience. Implementation used in our previous work [1].
- *GPS Discovery* - inter-experience edges between nodes generated from closest GPS points.
- *Refined Discovery* - candidate inter-experience edges suggested from GPS and then refined using robust image matching.

We present results for the minimum number of localisers N = 1 and 2 in Fig. 5 and Fig. 6. These are plots of outing number versus how much of the VO output needed to be saved for that run. This can also be thought of as how often the system was unable to localise against prior experiences, i.e. got lost. Note that the features in the graphs follow the nature of the data we collected. On runs 35-38 we drive the vehicle at dusk for the first time, with large pools of water

Fig. 4. 37km of visual data was collected over a three month period using the Wildcat, the group's vehicle.

on the road and light drizzle. The environment under these conditions had not been experienced previously, explaining the large jump in experiences saved. After this the vehicle makes several sorties where little or no experiences are created as we have a relatively complete representation of the workspace. We start to drive the inner loop of the route on visit 47, which causes a large and sustained spike in the number of experiences saved from this point onwards as the inner section of the loop has not previously been visited and is novel to the system.

The total percent of saved frames for each variant is shown in Table I. These numbers are the fraction of the maximum amount of experience data that could be saved, i.e. if we stored all VO output. As experience creation is driven by localisation failure, a lower number here indicates the system is performing better, i.e. it is localised for longer and is lost less. Performing Refined Discovery results in a 27% and and 20% improvement for N = 1 and 2 respectively, when compared to our previous work, Live Discovery. Interestingly GPS Discovery does not performed as well as Refined Discovery. This is likely to be caused by the quality of the inter-experience edges in G being substandard due to the drift present in GPS data.

Note how in both Fig. 5 and Fig. 6 GPS and Refined Discovery do not always out perform the original Live Discovery variant. This is because performance on a particular run is directly tied to what has been saved previously. As the Live Discovery stores more than is necessary in earlier runs, on some later outings it has more experiences to draw from and sometimes stays localised for longer than the other systems. However overall Table I shows that Refined Discovery performs best, followed by GPS Discovery and then Live Discovery. No Discovery always performs worst and saves significantly more experiences that the other variants.

In Fig. 7 we show, for each visit, the average number of successful localisers while the system is not lost. We see that both variants that process newly saved experiences between outings generally record a higher number of successful localisers for each run. Given they also store less experiences, this implies they are making much better use of the information they already have stored. By increasing the quantity and quality of edges in G, we have shown that we can stay localised for longer and are required to save less, as we making better use of the information we currently have.

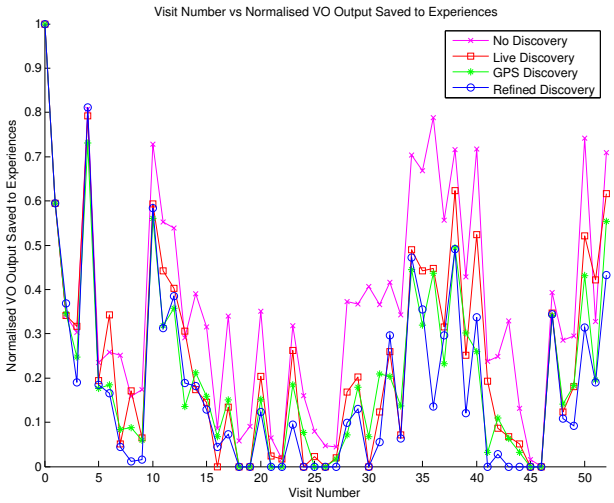

Fig. 5. System performance when using different graph structure discovery approaches. N = 1 for all runs.

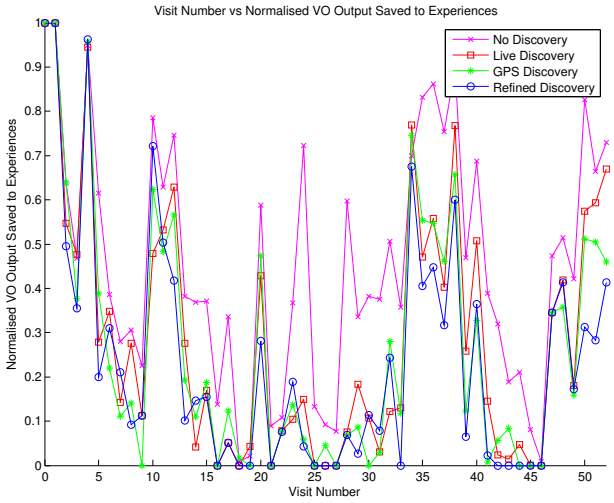

Fig. 6. System performance when using different graph structure discovery approaches. N = 2 for all runs.

VI. Related Work

The previous work that has attempt to tackle the long term navigation problem has generally approached it with a more typical "global metrically correct" approach. Konolige and Bowman's view-based map system [11] was developed to deal with dynamic indoor environments [12]. Milford and Wyeth's RatSLAM system [13] is also able to store different representations of the same place. In both of these systems, different appearances are connected with metric information to mutual nodes, so their different views of the world are already linked together. Biber and Duckett [14] create a map that has both long term and short term features by sampling prior laser maps at a series of time scales. This allows them to deal with long term structural change and short term dynamic objects. Their prior laser maps exist in a single global frame of reference, so fusing them to create a new map is trivial.

In contrast to these previous approaches, we do not adopt a

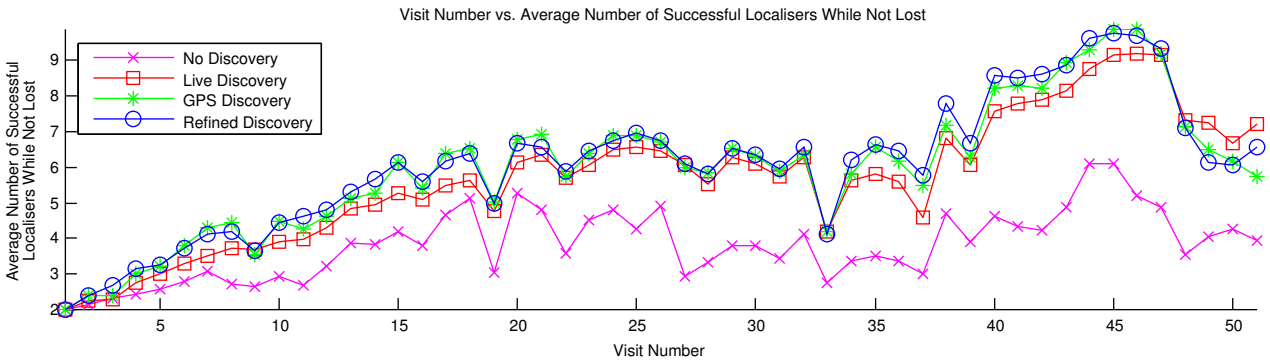

Fig. 7. For each run, the average number of successful localisers while the system is not lost is shown for each system variant. Note that GPS and Refined Discovery, both variants that spend time after a run to perform better matching of newly saved experiences have a higher number of successful localisers. Given both of these variants save less, it shows they are leveraging their experiences better. Results shown for N = 2.

TABLE I

PERCENTAGE OF SAVED OUTPUT (LOWER IS BETTER). GPS AND REFINED DISCOVERY METHODS WORK BEST.

	Discovery Type			
	No Discovery	Live	GPS	Refined
N = 1	34.93%	23.35%	19.89%	17.05%
N = 2	43.95%	26.48%	24.33%	21.16%

single frame of reference. When new experiences are saved, they remain independent. The advantage of this approach is that multiple experiences can cover the same physical space without being forced into the same frame of reference and localisation is trivially parallelised. The difficulty comes in knowing where different experiences cover the same physical space. We therefore require some sort of linking between experiences.

VII. CONCLUSION

In this paper we have shown how our original continuous localisation of a road vehicle can be improved with a new process, run between outings, to enhance the information shared between experiences. Previously we demonstrated that ongoing localisation can be achieved by saving distinct visual experiences, but that leveraging these when needed is not trivial. By placing all experiences in a single graph, we can create topological links between nodes that can be used to aid localisation. In our prior work we only augmented these edges when the system was online. By introducing this new step, we can increase the quality and quantity of these edges. Our results show that doing this, we can achieve a significant improvement in performance over our previous approach. We can stay localised for longer and need to save less information to achieve it.

VIII. ACKNOWLEDGEMENTS

Winston Churchill is supported by an EPSRC Case Studentship with Oxford Technologies Ltd. Paul Newman is supported by EPSRC Leadership Fellowship EP/I005021/1. This work has also been supported by BAE SYSTEMS.

REFERENCES

[1] W. Churchill and P. Newman, "Practice makes perfect? managing and leveraging visual experiences for lifelong navigation," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Minnesota, USA, May 2012.

[2] D. Nister, O. Naroditsky, and J. Bergen, "Visual Odometry for Ground Vehicle Applications," *Journal of Field Robotics*, vol. 23, 2006.

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

[4] M. Cummins and P. Newman, "Highly Scalable Appearance-Only SLAM FAB-MAP 2.0," in *Robotics Science and Systems*, 2009.

[5] E. Rosten, G. Reitmayr, and T. Drummond, "Real-time video annotations for augmented reality," in *Advances in Visual Computing. LNCS 3840*, December 2005, pp. 294–302.

[6] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary Robust Independent Elementary Features," in *European Conference on Computer Vision*, September 2010.

[7] C. Mei, S. Benhimane, E. Malis, and P. Rives, "Efficient homography-based tracking and 3-d reconstruction for single-viewpoint sensors," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1352–1364, Dec. 2008.

[8] 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 (6), p. 381395, 1981.

[9] H. Bay, A. Ess, T. Tuytelaars, and L. V. Goo, "Surf: Speeded up robust features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, pp. 346–359, 2008.

[10] N. Cornelis and L. V. Gool, "Fast Scale Invariant Feature Detection and Matching on Programmable Graphics Hardware," in *Computer Vision and Pattern Recognition*, 2008.

[11] K. Konolige, J. Bowman, J. D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua, "View-based maps," *International Journal of Robotics Research (IJRR)*, vol. 29, no. 10, 2010.

[12] K. Konolige and J. Bowman, "Towards lifelong visual maps," in *IROS*, 2009, pp. 1156–1163.

[13] M. Milford and G. Wyeth, "Persistent navigation and mapping using a biologically inspired slam system," *The International Journal of Robotics Research*, 2009.

[14] P. Biber and T. Duckett, "Dynamic maps for long-term operation of mobile service robots," in *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.