Towards Visual SLAM with Event-based Cameras

Michael Milford[1], Hanme Kim[2], Stefan Leutenegger[2] and Andrew Davison[2]
[1]Australian Centre for Robotic Vision, Queensland University of Technology
[2]Department of Computing, Imperial College
Corresponding author: michael.milford@qut.edu.au

1 Introduction

Event-based cameras (Figure 1) offer much potential to the fields of robotics and computer vision, in part due to their large dynamic range and extremely high "frame rates". These attributes make them, at least in theory, particularly suitable for enabling tasks like navigation and mapping on high speed agile robotic platforms under challenging lighting conditions, a task which has been particularly challenging for traditional algorithms and camera sensors. Before these tasks become feasible however, progress must be made towards adapting and innovating current RGB-camera-based algorithms to work with event-based cameras. In this paper we present ongoing work towards this goal and an initial milestone – the development of a constrained visual SLAM system that can create semi-metric, topologically correct maps of a 2.7 km traverse through a large environment at real-time speed (Figure 2). Although much more sophistication is yet to be built into the system, we hope this work serves as a baseline for future research using these novel sensors.

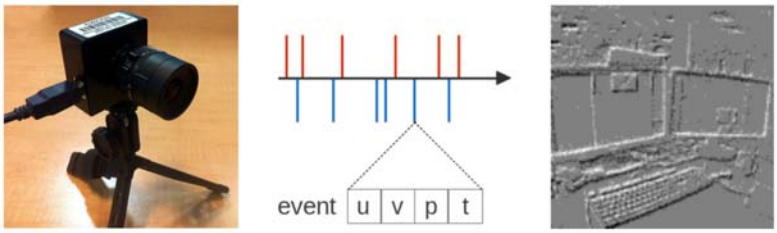

Figure 1: The first commercial event camera: (a) DVS128; (b) a stream of events (upward and downward spikes: positive and negative events); (c) image-like visualisation of accumulated events within a time interval (white and black: positive and negative events). From [1].

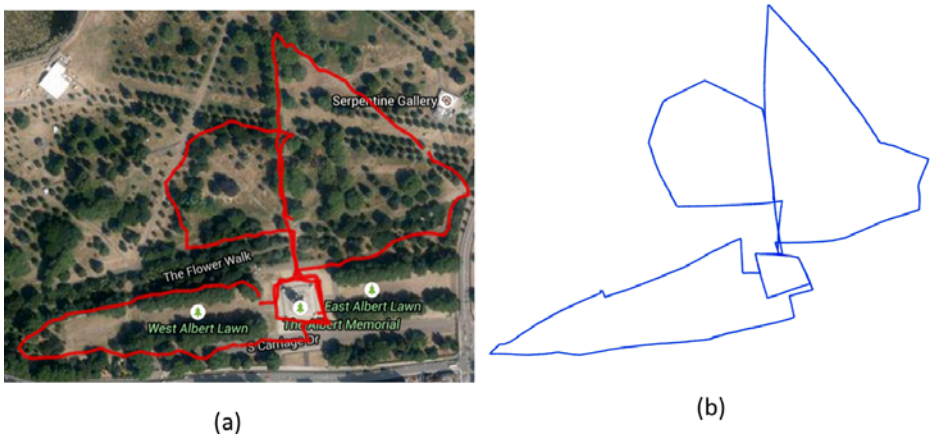

Figure 2: Map of a 2.7 km on foot trajectory through Hyde Park in London created in this research.

2 Approach

The proposed approach consists of three key modules; a loop closure system built on top of the SeqSLAM algorithm; a very limited visual odometry system that assumes a non-holonomic-like platform constrained to a ground plane and performs step counting to provide an approximate translational speed estimate; and a graphical map based on the experience mapping algorithm [2].

3 Place Recognition on Event Data using SeqSLAM

The SeqSLAM algorithm performs place recognition using camera frame sequences [3]. It uses low quality, low resolution imagery from RGB cameras under challenging lighting conditions, rendering it a potentially suitable algorithm for performing loop closure using the sparse, low resolution output from an event camera [4]. To provide place recognition capability, we temporally bin events to create synthetic "event" images, with the rest of the loop closure and its integration with the experience map similar to that described in [5].

3.1 Speed Sensitivity

We first conducted some simple experiments to qualitatively evaluate how sensitive place recognition performance was to camera speed, using a fixed temporal bin window size. We used the open source version OpenSeqSLAM, available from http://openslam.org/, using a sequence length of 100 frames. For these initial experiments, we assumed an approximate camera translational speed measure was available (such that might be obtained from a visual odometry system once implemented), in order to reduce the size of the search space for SeqSLAM.

We gathered three datasets at 3 different average speeds in an office environment from a forward facing DVS128 event camera being carried by an experimenter (Figure 3):

- **Run A:** running
- **Run B:** walking 66% of Run A speed
- **Run C:** slow walking 25% of Run A speed

Events were accumulated into 10 ms time windows (effective "fps" of 100) to form 128 × 128 event snapshots, which were downsampled to 16 × 16 pixel resolution then input into OpenSeqSLAM. We ran two experiments evaluating place recognition performance from the two slower traverses (Runs B-C) matching back to the fastest traverse (Run A). Approximate frame correspondence ground truth was obtained by manually inspecting the frames and is shown in Figure 3b. Each traverse contained internal loops.

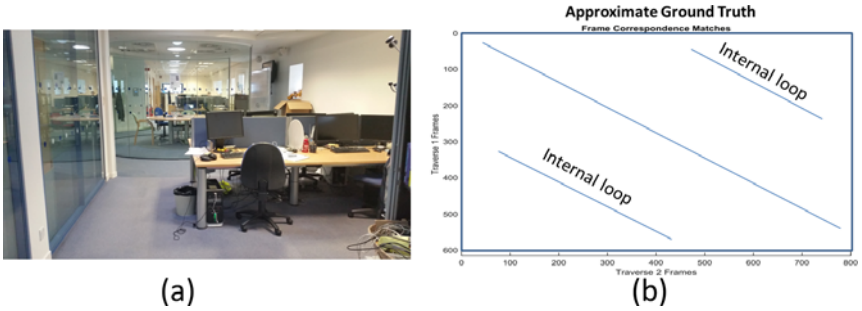

Figure 3: (a) Office testing environment and (b) approximate frame correspondence ground truth for matches between two traverses, obtained by manual inspection of video frames. Note the internal loop closures within each dataset.

Figure 4 shows the frame correspondence matches between Runs A and B, overlaid on the SeqSLAM confusion matrix. Almost 100% match coverage is obtainable with no significant localization errors.

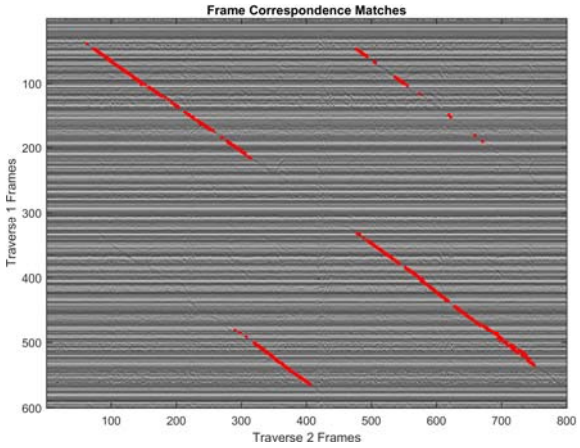

Figure 4: Frame matching for Runs A & B, with a 1.5 × speed differential.

Figure 5 shows the frame matches obtained between Runs A and C, for a speed differential of approximately 4 times. Matching coverage is reduced but still significant, with correct frame matches obtained over approximately 30% of the dataset.

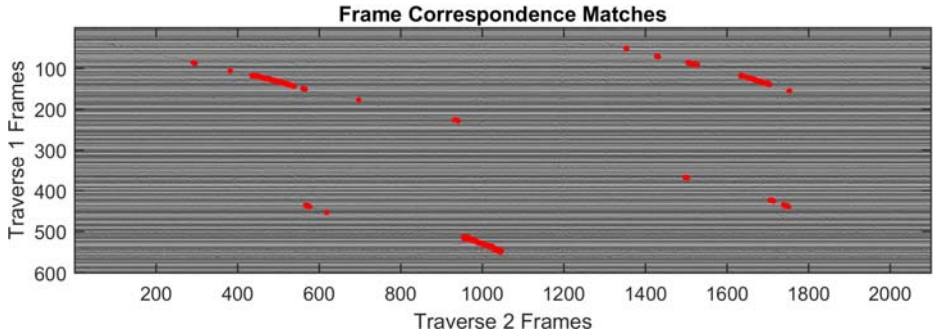

Figure 5: Frame matching for Runs A & C, with an approximately 4× speed differential.

Figure 6 shows sample frames from sequences that were successfully matched between Runs A and C. The matches were made despite large variations in camera speed and camera jitter that significantly changes both the overall event image and specific details such as polarity (for example polarity is partially switched in (b) due to opposite vertical camera jitter).

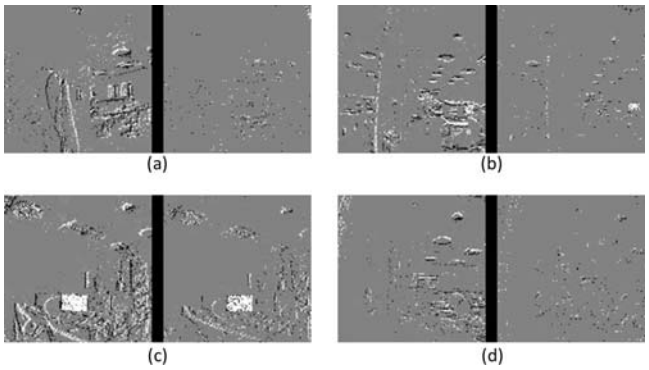

Figure 6: Sample frame matches between the fastest (A) and slowest (C) runs.

Based on our initial investigations, we chose to increase the fixed temporal bin window size to 1000 ms and removed the effect of event polarity (Figure 7), which we found to improve place recognition capability across a range of initial dataset tests, although we do not provide a comprehensive evaluation here.

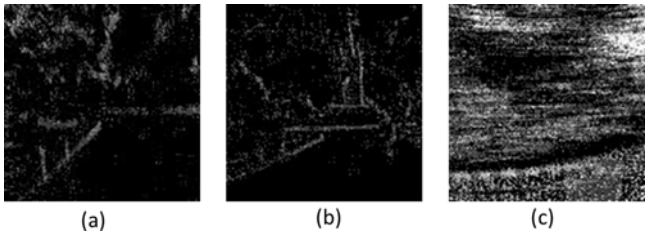

Figure 7: 1000 ms temporally binned windows with polarity removed. (a) Pathway with foliage on the left (b) looking at the Albert monument and (c) an approximate 1 rad/s turn.

4 Limited Visual Odometry

For a 128 × 128 pixel DVS camera moving slowly through a large open environment, there is little to no texture off the ground plane and only occasional readings from distal landmarks. Part of the limitation is due to using a relatively small field of view forward facing camera at a significant distance from the ground plane, resulting in there often being no detectable close range texture, a problem exacerbated by the low resolution of this first generation sensor. For the simple non-holonomic platform assumption used in this work (car-like platform with centrally mounted camera over rotation axis), rotation is fairly straightforward to detect, but translation appears to be extremely difficult. Consequently for this work we used the event camera to estimate step counts – at the end of the paper we discuss several ways in which a more satisfactory solution may be developed in future.

4.1 Patch Tracking

Based on the success of performing place recognition using patch tracking followed by a geometric coherency check [6], we created a similar system that tracked patch movement over consecutive frames (Figure 8a) and then performed a coherency check (Figure 8b-c) using the 2D histogram of horizontal and vertical patch shifts. We tracked patches over a grid of 6 × 6 patches, each 15 × 15 pixels in size, tracked over possible shifts of up to 15 pixels in either direction. For the experiments presented here we used a coherency threshold of 70%: 70% of valid patch shifts had to be consistent in order for the overall image shift to be accepted. Invalid patch shifts occurred in regions of zero texture, a common occurrence with event camera-based data.

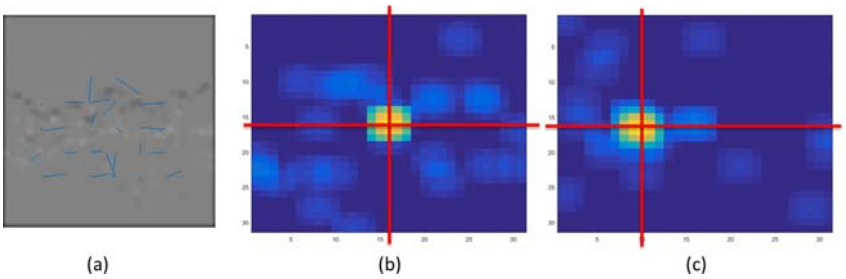

Figure 8: (a) Patch tracking and (b) smoothed 2D patch shift histograms for no rotation and (c) a camera rotation to the right (resulting in a consistent leftward pixel shift).

4.2 Rotation Estimation

We estimated approximate rotation using the detected horizontal pixel shift x_{shift} and effective field of view f:

$$\Delta\theta = f\,\frac{x_{shift}}{128} \tag{1}$$

Although this approach does not explicitly decouple apparent rotation due to translation versus pure rotation on the spot, it has been shown to be acceptable for producing topological maps on constrained wheel-based platforms, such as in [2].

4.3 Step Counting

Step counting was simply performed by counting transitions between positive and negative vertical image shifts, with an assumed fixed step size. We calibrated the step size using a short straight path dataset of known length.

5 Experimental Setup

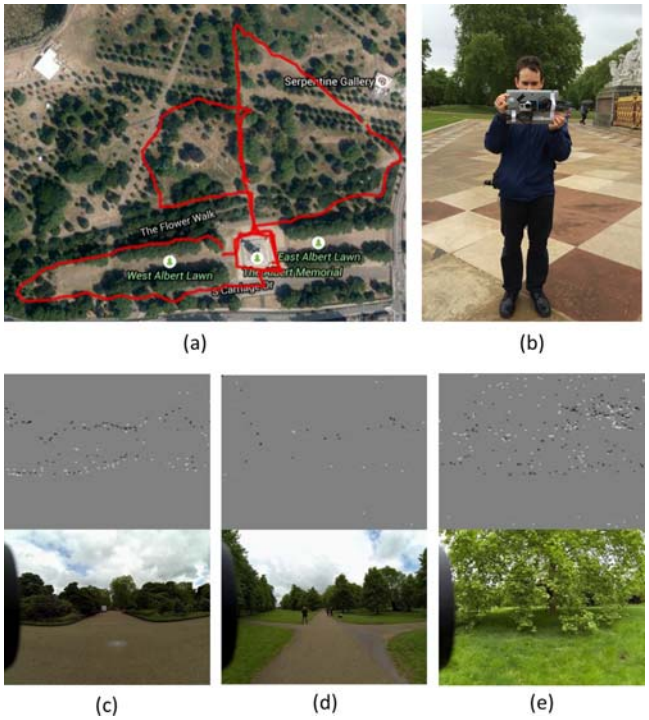

Figure 9: (a) Dataset route, (b) acquisition rig, and (c-e) event frames and sample corresponding GoPro frames from the dataset.

We acquired a 2.7 km dataset through Hyde Park traversing both pathways and grassland at varying walking speeds (Figure 9, although the "platform" in this situation was a human our initial platform target is non-holonomic ground-based vehicles). Acquisition occurred on an overcast day with many dynamic obstacles e.g. pedestrians and maintenance vehicles. Because of the open nature of the environment, events were relatively sparse compared to in an enclosed indoor environment (compare Figure 9c-e with Figure 6).

6 Results

The video accompanying this paper (https://youtu.be/FPZzcKA5LZ0) shows the entire mapping experiment including real-time evolution of the SLAM map over the course of the experiment.

6.1 Visual Odometry

Figure 10 shows the rotational velocity calculated for the entire dataset, with a zoomed in section shown for the first 7 turns at the beginning of the dataset. Although quantitative analysis is not possible (only approximate GPS data was obtained), all the significant turns in the dataset were approximately captured with minimal spurious readings.

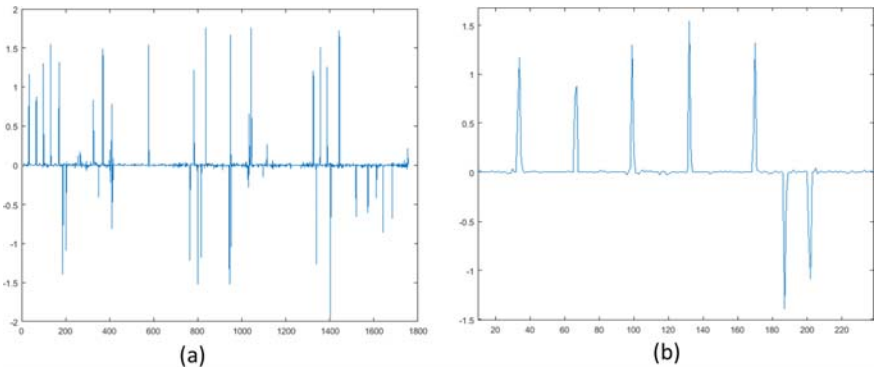

Figure 10: Rotational velocity over the entire dataset.

The calculated translational velocity, is, unsurprisingly, quite poor. The system does correctly capture several stoppage locations in the dataset.

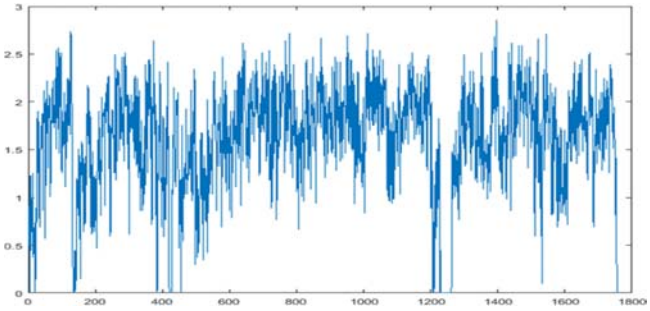

Figure 11: Calculated translational velocity over the entire dataset.

6.2 SLAM Results

Figure 12 shows the evolution of the graphical map over the duration of the experiment. Each loop closure is correctly detected, as also shown by the place recognition / loop closure graph shown in Figure 13. Visual odometry performance is "good enough" to avoid catastrophic failures of the graph relaxation algorithm. The final map is approximately 85% of real size.

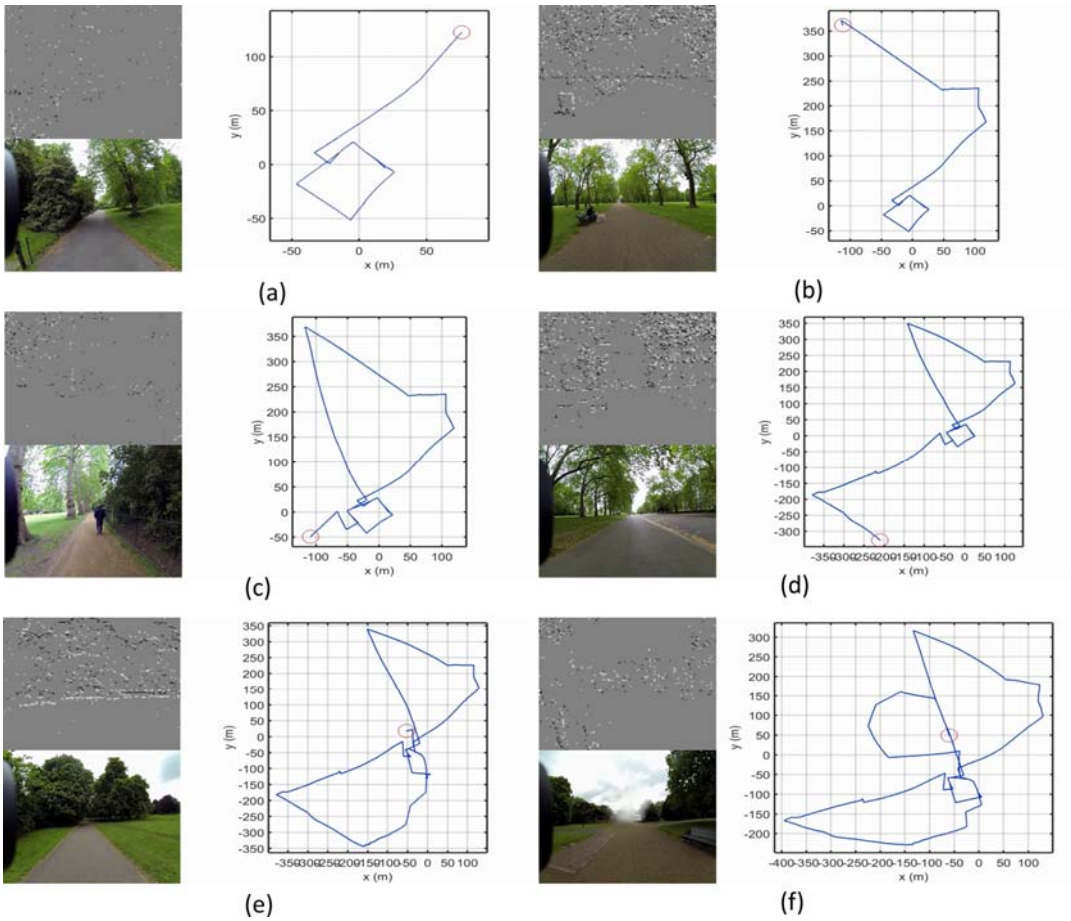

Figure 12: SLAM map over the duration of the experiment. The slight map shrinkage after the last loop closure is a result of the graph relaxation technique used (see [7])

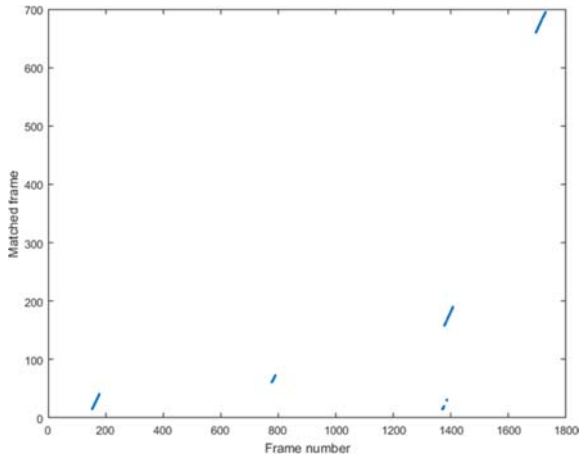

Figure 13: Place recognition / loop closures reported by OpenSeqSLAM.

7 Summary

We have presented current ongoing research and results towards the eventual goal of a complete and generally applicable visual SLAM system using event cameras. As a first step towards this goal, we have investigated whether current techniques can be adapted to replicate

some of the initial large scale (but limited in scope) visual SLAM results [2]. Our initial findings are these:

1. Performing visual place recognition or loop closure with event cameras is of comparable difficulty to performing it with normal RGB cameras, albeit being subject to the same challenges of achieving viewpoint invariance or reporting metric viewpoint change. We were able to achieve reliable place recognition reporting indoors and outdoors and at varied speeds.
2. Performing general 6DOF visual odometry with the current first generation event cameras is difficult, due in part no doubt to their very low resolution, although rotation detection is reasonable given a non-holonomic platform assumption. Soon to be released cameras will ease this challenge somewhat with QVGA or perhaps even VGA resolutions. Recent related work is also promising for providing motion estimation [1, 8].

Ultimately the appeal of this novel sensing modality is twofold – their extremely low latency for reporting events and low power consumption suggests applications on high speed, highly agile platforms such as small quadrotors or perhaps wearable devices [9], while the recent development of neural computing architectures may more readily suit event-based rather than traditional camera vision. Whether these projected benefits will eventuate is yet to be seen – we hope with this work to have established an initial baseline upon which future work can build upon. In future work we will seek to push these cameras to the limit by deploying them on rapidly manoeuvring platforms in cluttered, poorly lit environments.

8 Acknowledgements

This research was partially supported by the ARC Centre of Excellence in Robotic Vision CE140100016, an ARC Future Fellowship FT140101229 to MM and a Microsoft Research Faculty Fellowship to MM.

9 References

1. Kim, H., et al., *Simultaneous Mosaicing and Tracking with an Event Camera*, in *British Machine Vision Conference*. 2014. p. 566-576.
2. Milford, M. and G. Wyeth, *Mapping a Suburb with a Single Camera using a Biologically Inspired SLAM System.* IEEE Transactions on Robotics, 2008. **24**(5): p. 1038-1053.
3. Milford, M. and G. Wyeth. *SeqSLAM: Visual Route-Based Navigation for Sunny Summer Days and Stormy Winter Nights*. in *IEEE International Conference on Robotics and Automation*. 2012. St Paul, United States: IEEE.
4. Milford, M., *Vision-based place recognition: how low can you go?* International Journal of Robotics Research, 2013. **32**(7): p. 766-789.
5. Milford, M., et al., *Featureless Visual Processing for SLAM in Changing Outdoor Environments.* Journal of Field Robotics, 2014. **31**(5).
6. Milford, M., et al. *Condition-Invariant, Top-Down Visual Place Recognition*. in *IEEE International Conference in Robotics and Automation (ICRA)*. 2014.
7. Milford, M.J., D. Prasser, and G. Wyeth. *Experience Mapping: Producing Spatially Continuous Environment Representations using RatSLAM*. in *Australasian Conference on Robotics and Automation*. 2005. Sydney, Australia: ARAA.
8. Mueggler, E., et al., *Lifetime Estimation of Events from Dynamic Vision Sensors*, in *IEEE International Conference on Robotics and Automation*. 2015, IEEE: Seattle, United States.
9. Mayol, W.W., et al., *Applying active vision and SLAM to wearablese.* Robotics Research, 2005. **15**: p. 325-334.