Vision-based Simultaneous Localization and Mapping in Changing Outdoor Environments

Michael Milford
Queensland University of Technology, Brisbane, Queensland, Australia 4001
Eleonora Vig, Walter Scheirer, and David Cox
Harvard University, Cambridge, Massachusetts 02138

Received 13 July 2013; accepted 16 May 2014

For robots operating in outdoor environments, a number of factors, including weather, time of day, rough terrain, high speeds, and hardware limitations, make performing vision-based simultaneous localization and mapping with current techniques infeasible due to factors such as image blur and/or underexposure, especially on smaller platforms and low-cost hardware. In this paper, we present novel visual place-recognition and odometry techniques that address the challenges posed by low lighting, perceptual change, and low-cost cameras. Our primary contribution is a novel two-step algorithm that combines fast low-resolution whole image matching with a higher-resolution patch-verification step, as well as image saliency methods that simultaneously improve performance and decrease computing time. The algorithms are demonstrated using consumer cameras mounted on a small vehicle in a mixed urban and vegetated environment and a car traversing highway and suburban streets, at different times of day and night and in various weather conditions. The algorithms achieve reliable mapping over the course of a day, both when incrementally incorporating new visual scenes from different times of day into an existing map, and when using a static map comprising visual scenes captured at only one point in time. Using the two-step place-recognition process, we demonstrate for the first time *single*-image, error-free place recognition at recall rates above 50% across a day-night dataset without prior training or utilization of image sequences. This place-recognition performance enables topologically correct mapping across day-night cycles.

1. INTRODUCTION

Visual mapping and navigation on robots has advanced rapidly in the past decade. There are now many vision-based techniques, including FAB-MAP (Cummins & Newman, 2009), MonoSLAM (Davison, Reid, Molton, & Stasse, 2007), FrameSLAM (Konolige & Agrawal, 2008), V-GPS (Burschka & Hager, 2004), Mini-SLAM (Andreasson, Duckett, & Lilienthal, 2007), SeqSLAM (Milford, 2013; Milford & Wyeth, 2012), and others (Andreasson, Duckett, & Lilienthal, 2008; Konolige et al., 2008; Paz, Pinies, Tardos, & Neira, 2008; Royer et al., 2005; Zhang & Kleeman, 2009) that are competitive with or superior to range sensor-based algorithms, with routes as long as 1,000 km being mapped (Cummins & Newman, 2009). The majority of these systems have been developed and demonstrated largely under certain conditions: high-quality imaging sensors have been used, on relatively stable vehicle platforms and in bright illumination conditions, minimizing problems such as motion blur and changes in appearance. However, these are restrictive constraints, especially as robots are expected to operate over longer periods of time and with lower hardware costs.

Direct correspondence to: michael.milford@qut.edu.au

Recent attempts to develop vision-based place-recognition algorithms that work despite environmental change are likewise restricted by requirements such as the need for prior training (Johns & Yang, 2013; Neubert, Sünderhauf, & Protzel, 2013; Sünderhauf, Neubert, & Protzel, 2013) that attempts to teach a statistical description of environmental change, or buffering of long image sequences (Milford, 2013; Milford & Wyeth, 2012) before matching can begin.

In this paper, we describe research toward enabling any-time vision-based simultaneous localization and mapping (SLAM) for outdoor robots in changing environments equipped with cheap consumer-grade cameras. The focus is on scenarios in which, due to the combination of cost limitations, illumination, and weather changes, the usefulness of traditional feature-based techniques such as scale-invariant feature transforms (SIFT) (Lowe, 1999) and speeded-up robust features (SURF) (Bay, Tuytelaars, & Van Gool, 2006) is limited. We present a novel multistep visual recognition algorithm that comprises an initial low-resolution image comparison step followed by a higher-resolution patch-verification step that enables for the first time accurate visual matching of *single* images across challenging perceptual changes such as day-night cycles in visually aliased

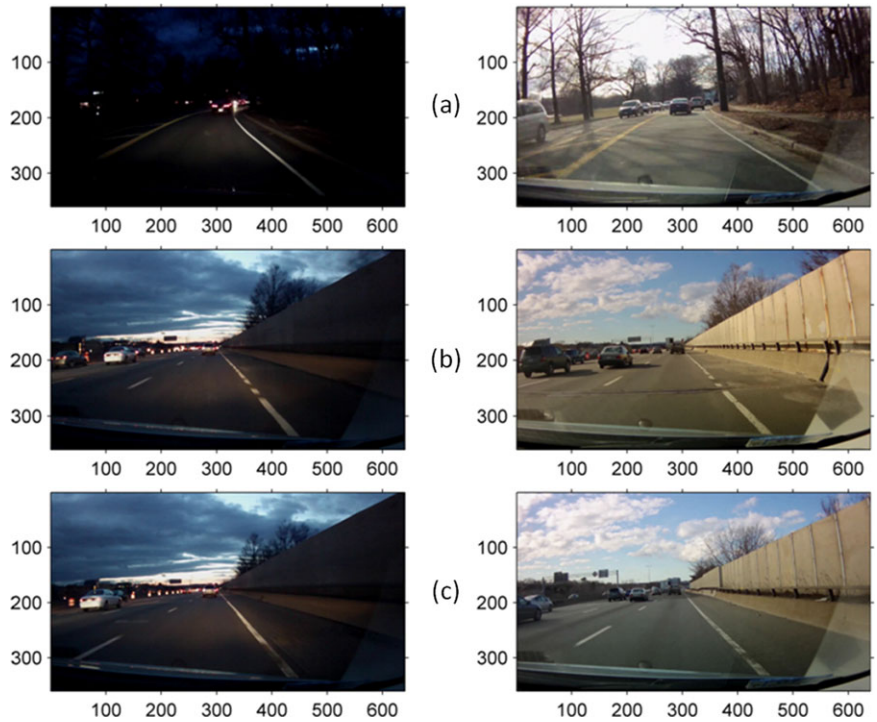

Figure 1. One of the primary contributions of this research is a two-step place recognition algorithm that correctly matches the image pair shown in (a) despite extreme perceptual change, as well as the image pair shown in (b), while also rejecting highly aliased image pairs such as shown in (c), which are actually images taken at different locations along the highway (the distant road signs in the middle left of each image are actually different signs and the distant tree clusters in the middle right of each image are from different trees). The method achieves error-free image matching (100% precision) at recall rates above 50% without requiring prior training or sequences of images, across a degree of perceptual change that is too challenging for existing state-of-the-art feature-based methods (Milford & Wyeth, 2012).

environments (Figure 1) without prior training, as well as a new method for obtaining coarse visual odometry information in these same challenging conditions. We evaluate the algorithms working at real-time speed in both a mixed offroad and urban environment at four times of day with varying environmental conditions: at dawn, during the morning, during a rain shower, and in fading light at dusk, and across a day-night cycle on a highway and suburban road network.

The research presented here builds on previous work, including mapping of a suburban road network at different times of day (Glover, Maddern, Milford, & Wyeth, 2010; Milford & Wyeth, 2010a), sequence-based localization on road networks (Milford, 2011; Milford & Wyeth, 2012), and static image region saliency studies (Milford, 2013). The degree of perceptual change encountered in the datasets presented here is qualitatively larger than in Glover et al. (2010). In contrast to Milford (2011) and Milford and Wyeth (2012), which were only place-recognition studies, we implement a full SLAM solution that calculates and uses motion information to build a map and localize within that map. And finally, we extend upon the work first presented in Milford and George (2012) with the following new contributions:

- A patch-verification algorithm that evaluates the correctness of the image matches proposed by a whole image matcher, leading to fivefold improvements in single-image recall levels achievable at 100% precision.
- The use of edge-detection-based and human-model-based image saliency masks in the patch-verification process calculated on a per image basis, leading to simultaneous improvements in both maximum precision-recall performance and computational requirements.
- A new lightweight visual odometry method using filtered intensity profile tracking that functions in challenging conditions, based on ground plane and wheel configuration assumptions.
- New GPS-tagged day-night road datasets along highways and suburban streets.
- Extensive analysis of the performance of the new algorithms on the new datasets, including precision-recall graphs, maps, and patch-matching examples.

Figure 2. Visual change in an environment over the course of a day and in varying weather: (a) dawn, (b) morning, (c) rain, and (d) dusk. In addition to changing illumination, other challenges are present, such as motion blur from the jerky motion of the platform when traveling offroad.

The paper proceeds as follows. In Section 2, we review the problem and motivation of visual processing under challenging environmental conditions. Section 3 describes our approach, including the mapping system, visual odometry techniques, image-matching methods including whole image matching and patch verification, and saliency mask generation and application. The setup for two experiments is described in Section 4, including a small vehicle traveling through a mixed park and campus environment over the course of a day, as well as a car-mounted camera traversing a journey along highways and suburban roads during the day and at nightfall. Section 5 presents extensive results from both experiments and analyses on the performance of the place-recognition algorithms, the visual odometry algorithms, and the combined system's ability to generate stable maps of the environment. Finally, we conclude in Section 6 with a discussion including a review of promising areas for future work.

2. BACKGROUND

For less than 100 dollars, one can now purchase a small but mechatronically highly capable ground-based vehicle or quad-rotor along with camera sensors, computing hardware, batteries, and enough electronics to give it at least the potential for a reasonable degree of autonomy. Camera technology in particular has advanced incredibly rapidly over the past 10 years, with compact multi-megapixel cameras now available for only a few dollars. This rapid reduction in the cost of many essential components for producing small, low-cost robots has opened up a tantalizing range of possibilities in robotics, such as having fleets of small, somewhat dispensable low-cost robots performing environmental monitoring, surveying, or surveillance duties at all times of day and night and come rain or shine. One of the major challenges in such scenarios is mapping and navigating the world under diverse environmental conditions, especially when placement of beacons is impractical and continuous reliance on GPS is infeasible. While a range of sensing technologies exists for performing mapping and navigation, it is arguably vision-based technology that has benefited most from recent technological advances: current visual sensors are cheap, can be very small, are passive and hence do not interact with the environment, have low power usage, provide a two-dimensional (2D) rather than a 1D snapshot of the environment, and can be used indoors where GPS generally cannot. However, while megapixel counts have burgeoned and sensor quality has improved, there are still fundamental challenges in creating cheap, vision-based mobile robot systems, including the following:

- the difficulty of obtaining the relatively high-quality images required by most conventional feature-based vision-processing techniques, when using low-cost cameras at speed on offroad terrain and/or in poor lighting,
- the problem of achieving reliable place recognition and odometry in outdoor environments over day-night cycles and during different types of weather.

Figure 2 (and Figure 1) illustrates these two challenges. Large changes in illumination [compare Figures 2(a) and 2(d)] or changes in the weather [see raindrops on the lens in Figure 2(c)] can radically alter the types of features detectable by a state-of-the-art algorithm such as scale-invariant feature transforms (SIFT) (Lowe, 1999) and speeded-up robust features (SURF) (Bay et al., 2006). Furthermore, in poor lighting with low-cost hardware and on offroad terrain, image blur is hard to avoid [Figure 2(c)]. Motion blur affects both the place recognition and odometry components of a mapping system, while a change in appearance over the course of a day primarily affects place recognition.

To some degree, these problems can be reduced by using active illumination, more capable sensing equipment, and implementing techniques such as high dynamic range imaging (Kiyoshi, Tomoaki, & Masahiro, 2011). However, active illumination has the disadvantages of (usually significant) additional power drain and active "interaction" with the environment, which may not be desirable in populated (by humans or animals) environments. In addition, images taken at night using an artificial light source often look very different from those taken during the day with natural lighting. Without active illumination of an environment, even well-exposed long exposure images can look very different from an image obtained in sunlight during the day, and

camera motion during the exposure compounds this problem (Milford, 2013). While motion estimation from blurred images can be achieved by tracking edges, it is unclear how well such an approach would expand to more naturalistic environments without strong lines (Klein & Murray, 2008).

Although there have been recent advances in relatively cheap range-based sensing hardware such as the Kinect (Whelan, McDonald, Kaess, Johannsson, & Leonard, 2012) and the ranging sensor found on Neato vacuum cleaner robots, their utility is limited in uncontrolled outdoor environments where sunlight can impinge directly on the sensor (in the case of the Kinect) or where rough terrain and jerky vehicle motion render in-plane-only scanning (in the case of other cheap range sensors) ineffective for map creation and place recognition. Their current prices ($50–$200) are still significantly higher than small cameras ($5–$40). High dynamic range techniques (Kiyoshi et al., 2011) become less viable as the speed of the platform increases. More capable sensors and lenses are expensive, usually bulkier and heavier to accommodate larger imaging sensors and lenses, and require more power. Even on large expensive platforms where the sensor cost is relatively small, conventional vision-based processing is not currently feasible in very dark environments. Analysis has shown that even with large sensor pixel sizes, a surprisingly small number of photons hit each pixel in naturally illuminated environments at night (Clark, 2005), at least at the exposure speeds required to produce crisp images (Milford, 2013). Feature-based mapping approaches using thermal cameras also produce poor place-recognition performance in road-based and naturalistic environments (Vidas & Maddern, 2012), since temperature distributions become highly uniform late at night. Thermal cameras are also currently too expensive to be considered in low-cost robotic applications.

It is therefore perhaps not surprising that despite the many on-paper advantages of vision-sensing technology over other sensing modalities, the majority of current practical robot and personal navigation systems rely primarily on GPS, laser/structured light range finders, or external beacons. In this paper, we attempt to rectify that imbalance by presenting and demonstrating techniques for vision-based place recognition and odometry using consumer cameras in perceptually changing and challenging environments.

3. APPROACH

In this section, we describe all the image-recognition and patch-verification algorithms, the patch-based and filtered intensity-profile-based visual odometry techniques, and the RatSLAM system and stand-alone graphical mapping algorithms we used to evaluate the mapping performance achievable using the place-recognition and odometry algorithms. The configuration of the various algorithmic modules for the two main experiments presented in this paper is shown in Figure 3.

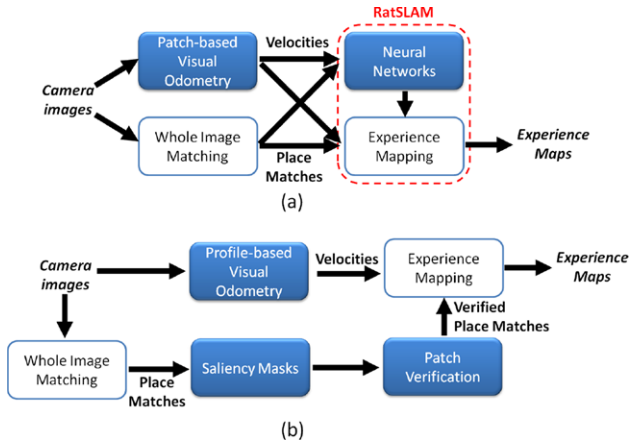

Figure 3. Overview of the algorithmic modules and their configuration for (a) the park and campus experiment and (b) the highway and suburban roads day-night experiment. The common *whole image matching* and *experience mapping* modules are shown as nonshaded shapes.

3.1. RatSLAM System

For the first experiment in the park and campus environment, data output by the visual recognition and visual odometry algorithms were processed by the RatSLAM system. RatSLAM is a robot SLAM system based on models of the navigation processes thought to occur in the rodent brain, specifically the rodent hippocampus (Milford, 2008).

The RatSLAM system consists of three modules, as shown in Figure 4. The local view cells encode visual scenes in the environment, with cells incrementally recruited to represent new distinct visual scenes as they are encountered. The pose cells are a network of highly interconnected neural units connected by both excitatory (positive or reinforcing) and inhibitory (negative) connections. They encode an internal representation of the robot's pose state, and filter both the place-recognition and self-motion information provided by the visual recognition and visual odometry processes. Finally, the experience map is a graphical map made up of nodes called experiences that encode distinct places in the environment, and connected by transitions that encode odometry information. A graph relaxation algorithm (Milford & Wyeth, 2008; Milford, Prasser, & Wyeth, 2005; Olson, Leonard, & Teller, 2006) is run continuously on the experience map, resulting in the continuous map evolution seen in the video accompanying the paper and also shown in Figures 16 and 18. Further information on the RatSLAM system can be found in Milford and Wyeth (2008, 2009).

3.1.1. Stand-alone Experience Mapping

To test the single-image place-recognition performance of the patch verification method presented in this paper, we

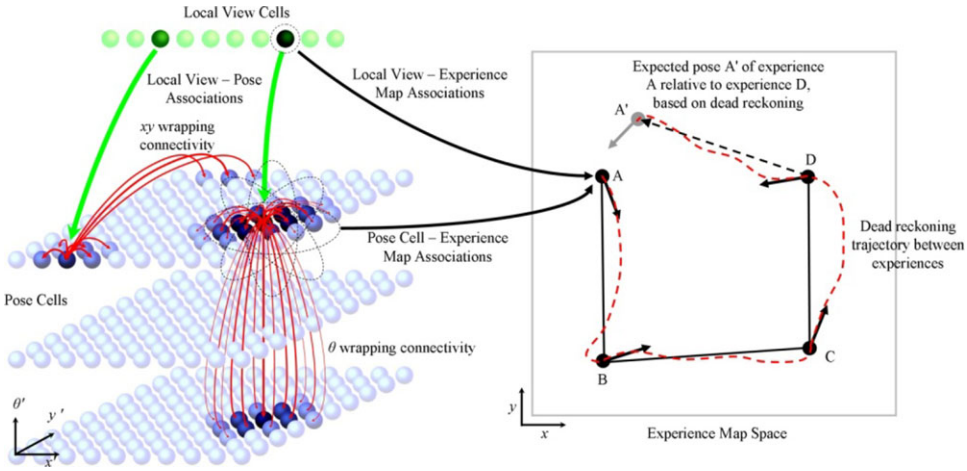

Figure 4. The RatSLAM system. The local view cells encode distinct visual scenes, while the pose cells encode an internal representation of the robot's pose and perform filtering of place recognition estimates and self-motion information. The experience map is a graphical map formed by the combination of the output from the local view cells, pose cells, and self-motion information.

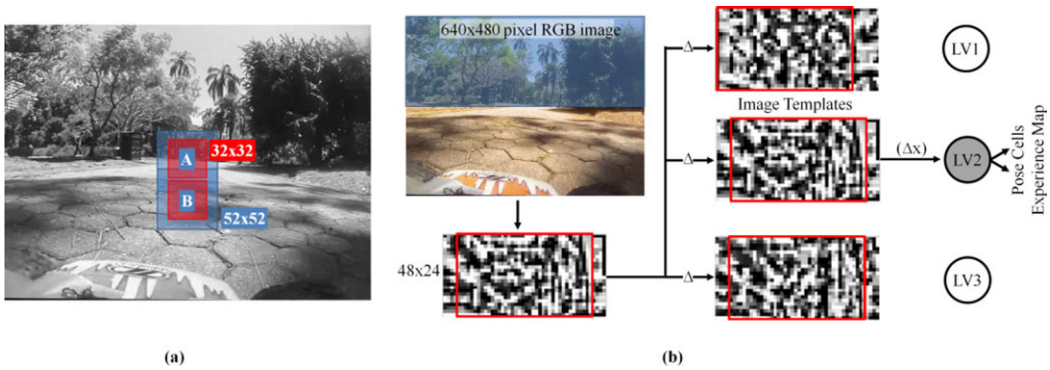

Figure 5. (a) Patch-based visual odometry is performed by tracking rotation using horizontal motion of patch A and vertical motion of patch B. (b) Patch-normalized template matching compares the current visual scene to those stored in a library of learned scenes.

developed a stand-alone implementation of the RatSLAM experience mapping algorithm, creating in essence a graphical mapping algorithm decoupled from any of the biologically inspired neural networks. Consequently, this lightweight mapping algorithm has no filtering capability and hence it provided an additional, map-focused test of the ability of the new place-recognition system to provide sufficient *single-image* recall rates at a 100% precision level (since any false positives would cause a catastrophic map failure), which complements the GPS-based precision-recall studies.

3.2. Patch-based Visual Odometry

The patch-based visual odometry system is a modified version of the system deployed on a quad rotor in Milford, Schill, Corke, Mahony, and Wyeth (2011). The system tracks movement of two image patches to calculate translational speed and yaw of the platform, as shown in Figure 5(a). The primary assumptions are that of a nonholonomic platform at a consistent height above a flat ground surface to give scale, as is done for monocular implementations of some leading visual odometry packages such as LIBVISO2, and relatively distal features in the top half of the image. Frame-to-frame motion of the top patch provides the yaw information, and bottom patch motion provides the translational speed. The odometry gain was calibrated by running the car along a known length of ground and calculating the required gain constant, given in Table II. Patch comparisons were performed by calculating the mean of the intensity difference between each pixel in the patch compared to the corresponding pixel in the previous image. Further implementation details are provided in Milford et al. (2011).

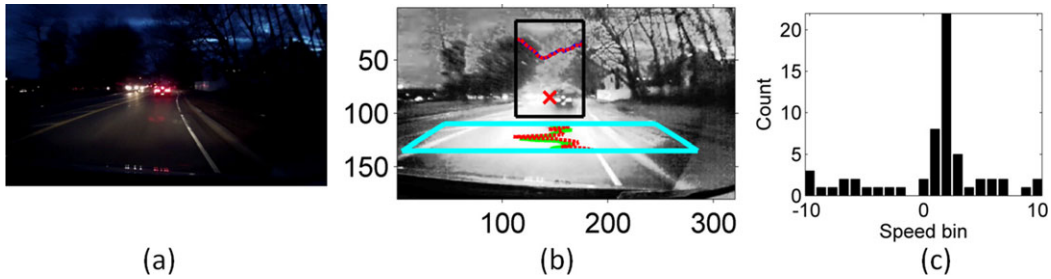

Figure 6. (a) Raw camera frame and (b) brightened, histogram-equalized downsampled frame. The top rectangle outlines the intensity profile area used to calculate rotational changes between consecutive frames, while the bottom trapezoid area is used to calculate translational motion. Actual current and past intensity profiles are shown inside the two areas. A histogram of recent translational motion values enables the identification of the modal speed calculation, in this case 2 units/s, which is converted into an absolute translational speed using a speed gain constant.

3.3. Filtered Intensity Profile Visual Odometry

To enable visual odometry at night with a cheap camera, we developed the improved visual odometry system shown in Figure 6. Assuming an approximately flat ground plane, nonholonomic wheel arrangement, and relatively distal features in the top half of the image, vertical intensity profiles (sums of pixel intensities across rows) were calculated for a region immediately in front of the car known to be the ground plane. To generate a translational speed prediction, a variable β tracked the difference between the current p_t and most recent p_{t-1} intensity profiles:

$$\beta_t = p_t - p_{t-1}. \tag{1}$$

The current and most recent intensity profile differences were then compared over a range of offsets up to value σ to determine the most likely shift s:

$$s = \min_{\Delta x \in [-\sigma,\sigma]} \frac{1}{h - \Delta x} \sum_{x=0} |\beta_t[x + \Delta x] - \beta_{t-1}[x]|, \tag{2}$$

where h is the length of the intensity profile difference vector, β_t is the current vector, and β_{t-1} is the most recent vector. Tracking is performed on intensity profile *differences* rather than just raw intensity values to remove the effect of relatively constant vertical intensity gradients (which cause the profile shift matching to fail) caused by headlights or other visual influences.

To ameliorate the effect of camera jerkiness and the difficulty of tracking movement of the blurry, indistinct ground plane, the u most recent speed values are histogrammed to yield the distribution shown in Figure 6(c.) The shift s associated with the maximal histogram bin is then multiplied by a pixel-to-speed constant δ that can be determined either through knowledge of the camera height, resolution, field of view and pitch, or calibrated on a dataset of known distance, to yield a translational speed in m/s (the approach taken in this work). The upper limit on the speed resolution of this approach is quite coarse—difference profile shifts of 0–10 pixels cover a speed range of 0 to approximately 150 km/h, but this coarseness is required to get a sensible velocity signal from the challenging night-time images.

Rotational velocity is determined in a similar manner by comparing horizontal profiles (sums of pixel columns) in a second image area as shown in Figure 6(b). Rotation is generally much easier to track even at night, and we found there was no need to implement any sort of filtering or histogram voting scheme. In contrast to our original intensity profile odometry system (Milford & Wyeth, 2008), this new approach has several significant improvements, including direct tracking of ground plane movement rather than implicitly through calculating the degree of general perceptual change, and consensus-based filtering of calculated velocities over multiple frames rather than independent frame-to-frame estimates.

3.4. Patch-normalized Whole Image Place Recognition

The low-resolution, whole image place-recognition process is illustrated in Figure 5(b), and it was used for all experiments in this paper. Camera images are captured and, in the case of the RC car, the ground plane is cropped. While the ground is useful for visual odometry, its proximity means that its appearance, when using a "whole of image" based recognition process, is sensitive to slight changes in vehicle pose when closing the loop, which tends to make place recognition brittle. The bottom 20% of the day-night camera images was cropped to remove the car dashboard.

Once cropped, the resolution of the image is reduced to 48 × 24 pixels (64 × 32 for the day-night experiment). Patch normalization is applied to the image in discrete 8 × 8 pixel square patches (rather than continuously over the image). Patch normalized pixel intensities, I', are given by

$$I'_{xy} = \frac{I_{xy} - \mu_{xy}}{\sigma_{xy}}, \tag{3}$$

where μ_{xy} and σ_{xy} are the mean and standard deviation of pixel values in the patch of size P_{size} within which (x,

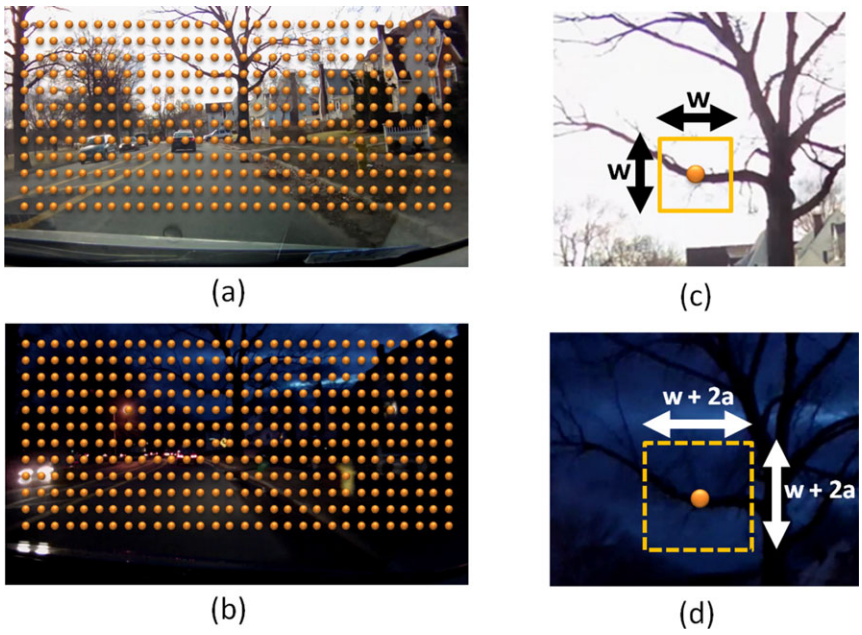

Figure 7. (a) Patch verification involves comparing small patches (c) at corresponding locations (a) and (b) in the proposed pair of images over a local sliding window (d).

y) is located. Mean image differences between the current visual scene and all the learned visual templates (previously stored scenes) are calculated using a normalized sum of intensity differences, performed over a range of horizontal and vertical offsets:

$$D_j = \min_{\Delta x, \Delta y \in [-\eta, \eta]} g\left(\Delta x, \Delta y, i, j\right), \tag{4}$$

where η is the template offset range, and $g(\,)$ is given by

$$g\left(\Delta x, \Delta y, i, j\right) = \frac{1}{s} \sum_{x=0} \sum_{y=0} \left| p^i\left[x + \Delta x,\ y + \Delta y\right] - p^j\left[x,\ y\right] \right|, \tag{5}$$

where s is the area in pixels of the template subframe, i is the index of the current frame, and j is the index of a previously learned frame. If the minimum difference D_j across all existing templates and relative offsets is larger than a threshold D_t, a new template is learned. Otherwise an existing template is matched, leading to (when integrated with the RatSLAM system) activation of pose cells associated with that visual scene and a possible loop closure event. The range of horizontal and vertical offsets provides (assuming the majority of objects in the image are relatively distal) some invariance to camera pose. This invariance enables loop closure even when routes are repeated at slightly different lateral offsets or at different orientations. This capability is important for offroad motion (in contrast to movement along a road network) where repeated paths vary due to environmental change or variation in the path executed by the human or autonomous navigation system.

3.5. Patch Verification

Whole image matching performance on low-resolution images becomes poor when perceptual change is large, such as over day-night cycles. The previous state-of-the-art approach SeqSLAM solved this challenge by matching over long sequences of images (Milford & Wyeth, 2012), rather than single images. The novel patch-verification process presented here is performed on image matches proposed by the whole image matcher described in the previous section. The concept is similar to that used in parallel tracking and mapping (PTAM) (Klein & Murray, 2007), which searches for subimage patch matches in a fixed range around the predicted matching location, although this information is used to update camera pose rather than perform place recognition.

Small image patches at corresponding locations in the two images [indicated by the dots in Figures 7(a) and 7(b)] are compared using a sum of absolute differences comparison, similar to that described in Eq. (5). Comparisons are performed over a sliding window centered on the patch location but extending in both vertical and horizontal directions (currently the whole image offsets are not carried through to the patch-verification stage). However, rather than just finding the maximal patch match and its associated offsets, the entire set of difference scores for each

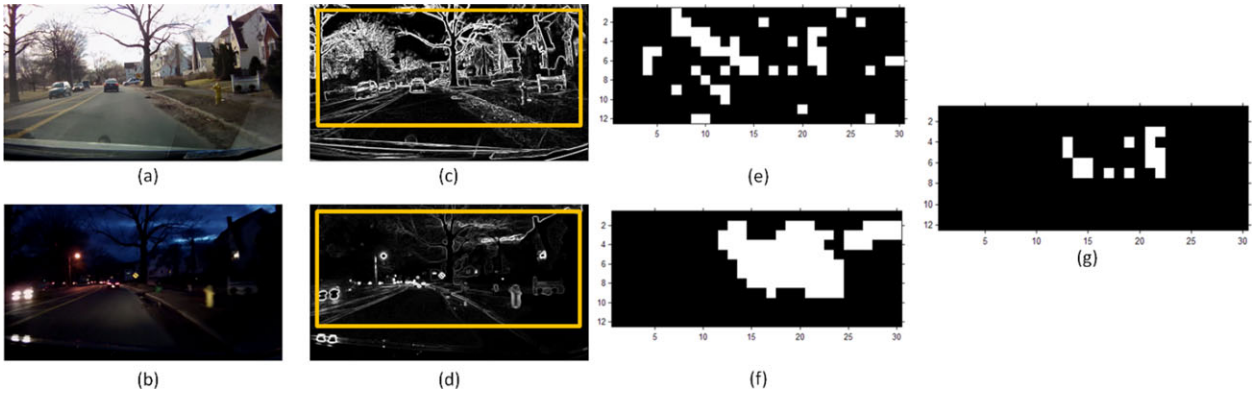

Figure 8. A simple contrast-based saliency mask was created by performing standard edge detection on (a) daytime and (b) nighttime frames to produce (c) and (d) edge-detected images. Note the images and salience masks in (a)–(d) all correspond to the same location. High-quality patch matches are shown by white squares in (e). A 20% edge-based saliency mask is shown in (f), which, when applied to the patch verification process, results in the high-quality patch matches shown in (g).

comparison is used to create a patch match quality score q:

$$q = \frac{g_1}{g_2} > g_{\text{rat}} \cap g_1 < g_{\max}, \tag{6}$$

where g_1 is the difference score for the best matching patch offset and g_2 is the score for the next best matching offset located outside a range of r_{peak} from the first score, g_{rat} is a minimum difference score uniqueness threshold, and $g_{\max}$ is a maximum absolute difference score (values given in Table IV). This approach is similar to that used in Lowe (2004). The number of patches meeting the match quality requirements is summed for a proposed pair of matching images to yield a matching patch count for that image. Patch matches meeting the quality requirements for various correct and incorrect patch pairs are shown in Section 5.2.4.

3.6. Saliency Mask Generation and Application

In the context of this work, a saliency mask is an 8-bit image the same size as the camera images, which is intended to provide a measure of the saliency or usefulness of each part of the image for the place-recognition task. A "good" saliency mask enables the patch-verification process to only process the most salient parts of the image, leading to decreased computation and potentially improved recognition performance. We generated three types of image saliency masks:

- a randomly generated control mask,
- a mask based on simple edge-detected images, and
- masks produced by a hierarchical model of visual saliency in humans.

Each type of mask was used in the same way. To evaluate which parts of the image upon which to perform patch verification, the saliency mask was sampled over each proposed patch comparison location [see the dots in Figures 7(a) and 7(b)], with each location being assigned a saliency score SS based on the summation of the saliency mask M in that area:

$$\text{SS} = \sum_x \sum_y M_{xy}. \tag{7}$$

The saliency scores were then ranked from strongest to weakest, and patch verification performed only in the top f fraction of locations, where f was a parameter that was swept between 0% and 100% to produce the performance curves shown in the Results section. Figure 8(f) shows a sample thresholded saliency mask including only the top 20% of salient regions. Figure 8(e) shows the patch-verification locations that resulted in patch matches exceeding the required quality score, and Figure 8(g) shows the patch matches that would actually be found with the 20% saliency mask applied.

3.6.1. Contrast-based Saliency

Edge-detected images were created using the default "Find Edge" filter in the Virtualdub PCVideo Image Processor V1.0A plugin.

3.6.2. Humanlike Saliency

Here we used a richly parametrized bio-inspired model family (Pinto, Doukhan, DiCarlo, & Cox, 2009) in conjunction with an efficient feature search paradigm (Bergstra, Bardenet, Bengio, & Kégl, 2011) to predict visually salient locations in the test datasets. Belonging to the class of convolutional neural networks, these hierarchical models mimic organizational principles in the human cortex and previously have been used successfully in object recognition and face identification. The aim here was to test whether a performance gain was possible using a sophisticated saliency model that had not been trained for the specific

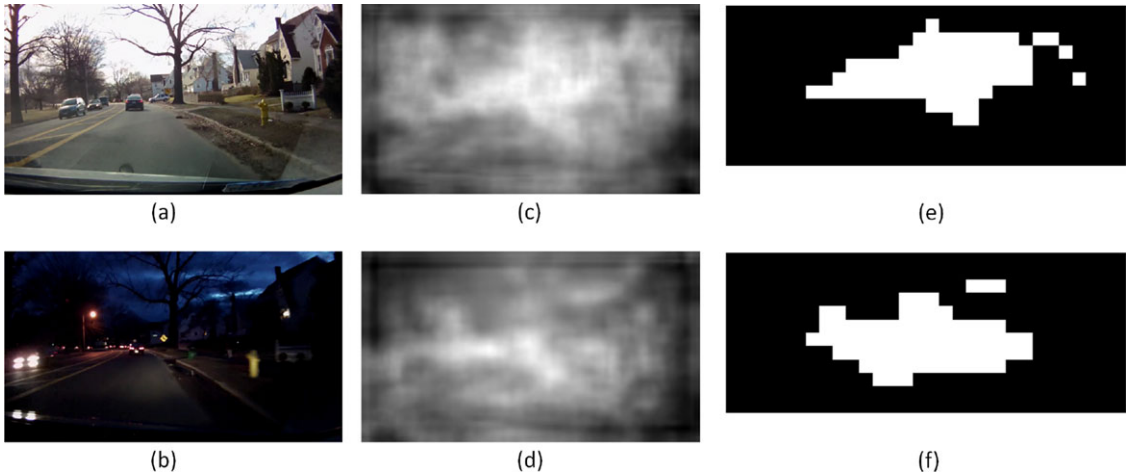

Figure 9. (a) Day and (b) night frames and (c) and (d) their associated saliency maps based on a model trained to mimic visual saliency in humans, also shown at (e) and (f) 20% coverage. All images and salience masks correspond to the same location, which is the same as that shown in Figure 8.

task of place recognition, as had previously been the case for other vision-based tasks. The basic multilayer model architecture is described in more detail in Cox and Pinto (2011) and Pinto et al. (2009). To derive a predictor of visual saliency, we employed an automatic data-driven approach in which efficient algorithms (Bergstra et al., 2011) searched the vast space of bio-inspired models to find those model instances that best predict eye movements in images. The optimal predictor of gaze was then assembled automatically through model blending. Model search and evaluations were carried out on the MIT eye movement benchmark (Judd, Ehinger, & Duran, 2009). In comparison to the edge-detected images, the human model generally produced more evenly distributed saliency masks, as can be seen in Figure 9.

4. EXPERIMENTAL SETUP

This section describes the testing platforms, cameras, and environments used to evaluate the algorithms.

4.1. Park and Campus Dataset

4.1.1. Testing Platform and Camera

The testing platform was a Team Losi Mini-LST2 remote control car with a Contour+ camera mounted facing forward. The camera has a fisheye wide-angle lens (2.8 mm focal length, approximately 170° field of view) and logged GPS data. Figure 10(a) shows the platform, while Figure 10(b) shows an autonomous version under development. Due to the risk of water damage during the rain dataset and the extreme nature of some of the offroad terrain (small logs, deep leaf litter), the nonautonomous platform was used. The video feed and GPS coordinates were logged onboard and processed offline. To reduce the effect of vibration and jerkiness due to the rough terrain and small size of the vehicle, videos were run through a stabilizing filter [VirtualDub Deshaker filter, available at Thalin (2010), default values used]. The use of a stabilizer introduces a one-frame lag between image capture and the image being available to the localization and odometry routines, equivalent to 33 ms at real-time speed.

4.1.2. Testing Environment and Datasets

Experiments were run over a one-week period in an area including the Queensland University of Technology campus and the City Botanic Gardens in Brisbane, Australia [Figure 10(c)]. The testing area measures approximately 200 m $\times$ 200 m and contains a mixture of open grass, pathways, gravel baths, shrubbery, garden beds, and buildings. The car was remotely driven by an operator following the vehicle.

A set of four datasets were gathered under a range of environmental conditions and at different times of the day (Table I). Each dataset repeated the same route, although minor deviations were inevitable due to pedestrian traffic, construction work, and the difficulty of the terrain in sections. A single traverse of the route was approximately 1,310 m in length (calculated by tracing the route on an aerial map) and took an average of approximately 15 min to complete. The car was jammed twice by branches and leaf litter and was stopped temporarily to remove the obstructing objects. These sections of video were cut, resulting in several discontinuous jumps in the footage. Frames were logged at 30 frames per second and downsampled to a resolution of 640$\times$480 pixels, with every frame processed by the visual odometry system but only every fifth frame processed by the visual template system, due to the high degree

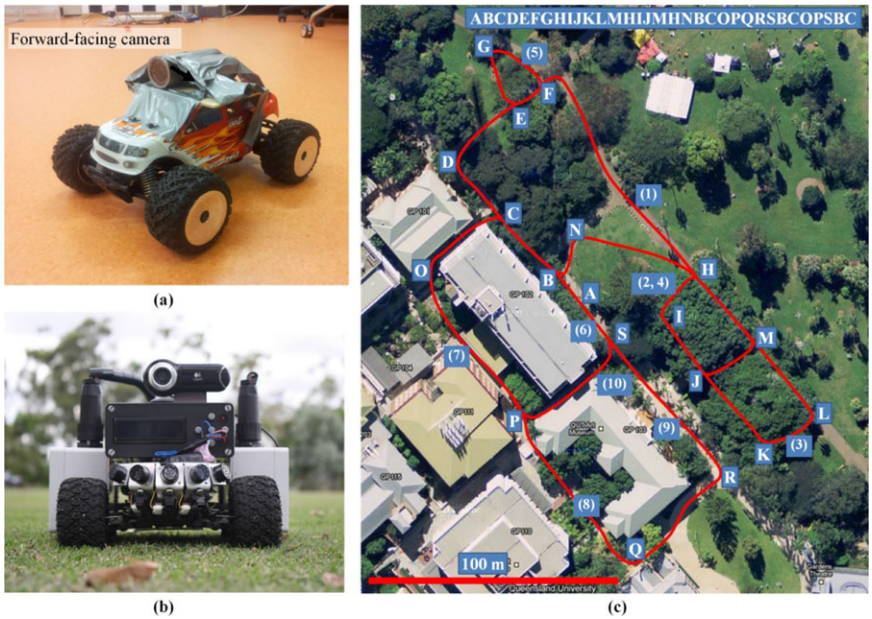

Figure 10. (a) Testing platform, a small but capable offroad enthusiast hobby car with mounted consumer camera, and (b) an autonomous version under development. (c) The vehicle path, with order indicated by the letter sequence. The numbers show the sample frame match locations from Figure 15. Imagery ©2013 DigitalGlobe, Sinclair Knight Merz & Fugro, Map data ©2013 Google.

Table I. Dataset descriptions. Times in Australian Eastern Standard Time (AEST).

Dataset name	Time and comments
Dawn	5:45 a.m. Sun just above local horizon, most areas in shade, excessive sun flare in sections
Morning	10:00 a.m. Sun high up in sky, large ground areas in bright sunlight
Rain	10:30 a.m. Rain drops on lens, wet ground, overcast, and dark
Dusk	6:45 p.m. Sun setting, extremely dark in heavily vegetated areas, significant motion blur, and lack of ground texture

of overlap between neighboring frames. The four datasets are available online.[1]

An attempt was made to use GPS tracking (CEP 10 m) as a ground truth measure. However, due to the heavily vegetated and urban canyon nature of much of the environment, the quality of the GPS tracking was too poor to be useful (far worse than specifications), as shown in Figure 11. The second set of experiments described in this paper partially resolved this GPS limitation by moving to a much larger scale environment.

[1]https://wiki.qut.edu.au/display/cyphy/Michael+Milford+Datasets+and+Downloads

Table II. Parameters.

Parameter	Value	Description
r	32 pixels	Odometry patch size
ς	0.375 °/pixel	Yaw gain constant
ν	0.0125 m/pixel	Translational speed constant
ρ	10 pixels	Odometry patch offset range
s	48 × 24 pixels	Template subframe size
D_t	0.06	Template learning threshold
σ	4 pixels	Template offset range

4.2. Highway and Suburban Roads Day-night Dataset

The second experiment was comprised of two 5.5 km journeys along highway and suburban roads [Figure 12(a), Tables III and IV], performed in the afternoon [Figures 12(b) and 12(c)], and then in the evening as darkness fell [Figures 12(d) and 12(e)]. Parts of the night-time dataset had no streetlights, meaning the only illumination available was from the test vehicle's headlights. Consequently, the degree of perceptual change due to nightfall in the latter part of the dataset in particular was significantly greater than in the previous experiment, as can be seen by sample frames such as those shown in Figure 12(e). The highway component of the datasets [Figures 12(b) and 12(d)] also provided a good test of the ability of the patch-verification algorithm

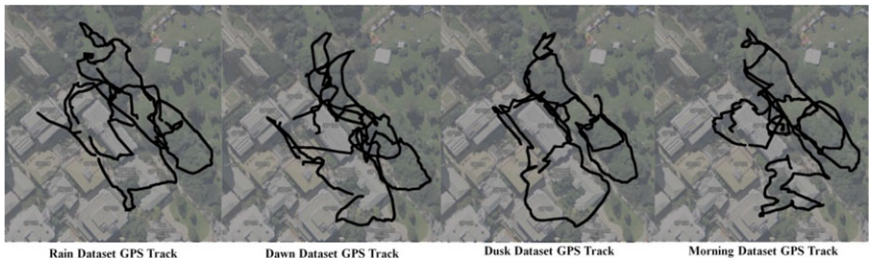

Figure 11. GPS was unreliable, especially under tree cover and around buildings.

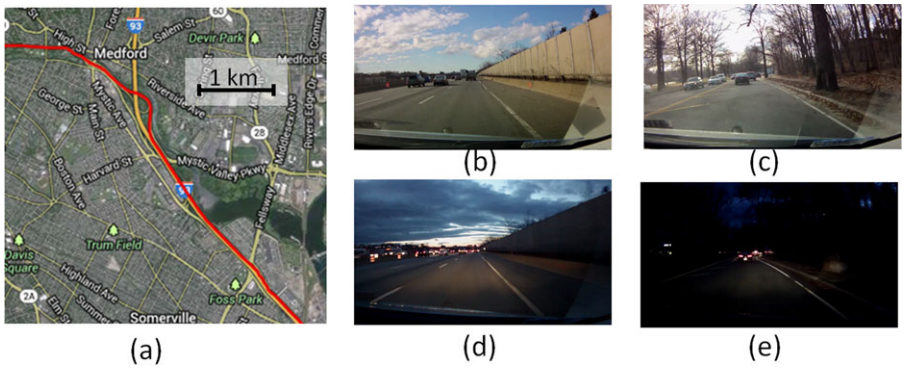

Figure 12. The day-night car dataset, a 5.5 km journey along a variety of (a) and (d) highway and (c) and (e) suburban streets during (b) and (c) an afternoon and (d) and (e) at nightfall. Imagery ©2013 TerraMetrics, Map data ©2013 Google.

Table III. Dataset descriptions.

Dataset Name	Time and Comments
Day	Afternoon, sky glare
Night	Nightfall, start of dataset is as dusk falls and end is very dark

to accurately localize across highly perceptually aliased environments, which has previously been identified as a major challenge for vision-based place-recognition algorithms (Cummins & Newman, 2009). The car's velocity was highly variable over the course of each run, ranging from 0 to just under 100 km/h. A forward-facing consumer bike camera mounted inside the windshield captured 1,080p images at 30 frames per second as well as GPS at 1 Hz.

Image contrast enhancement was performed on the day- and night-time road datasets (although the day dataset did not "need" image enhancement, the same enhancement was applied for the purposes of consistency). Many consumer cameras, including the ones used in this experiment, capture video in a YV12 format (chroma sampling scheme 4:2:0), which provides a useful 12 bits of intensity information per pixel, while sacrificing color representation. Often

Table IV. Parameter values.

Parameter	Value	Description
ς	0.25 °/pixel	Yaw gain constant
v	0.25 m/pixel	Translational speed constant
ρ	10 pixels	Profile matching offset range
s	64 × 32 pixels	Template subframe size
w	40 × 40 pixels	Patch size for patch verification
a	10 pixels	Patch verification local search range
u	60 frames	Velocity history (60 frames = 2 s)
r_{peak}	5 pixels	Patch quality score peak search exclusion zone
$g_{\max}$	0.3	Maximum difference score for an accepted patch match
g_{rat}	1.125	Minimum difference score ratio for an accepted patch match
-	320 × 160 pixels	Odometry image resolution
-	640 × 256 pixels	Patch verification image resolution
-	20 pixels	Patch location spacing (50% overlap with neighbors)

this extra intensity information is lost in a standard processing chain, but we applied brightening and histogram equalization to the original YV12 format images before converting them into standard gray-scale images for use by the place-recognition and visual odometry algorithms.

All images were processed by the visual odometry algorithm, but the frame rate was downsampled to 1 Hz before being input into the place-recognition algorithm. Image resolutions were also reduced to those shown in Table IV. Unlike the previous testing environment, the scale of the environment was large enough that the GPS could be used to semi-automate the analysis of the results by applying an initial 50 m threshold for true positives, which were then evaluated manually by inspection. The day dataset was processed first, meaning the place-recognition algorithm was required to match night-time images back to day-time images.

4.2.1. SeqSLAM Comparison

To provide a comparison to the state-of-the-art in condition-invariant place recognition, we ran the SeqSLAM algorithm on the highway and suburban roads dataset. We used an average sequence-matching length of 270 m and a repeatable velocity tolerance of approximately ±16%, similar to the original SeqSLAM study (Milford & Wyeth, 2012). A performance sweep was performed to generate a precision recall curve.

4.2.2. Parameter Values

Parameter values were tuned heuristically to produce good performance on these datasets. Although it was not feasible to do a complete parameter study, qualitatively we can state that the most significant parameter for the patch-verification process was g_{rat}; changing this parameter value would alter the precision-recall performance curve.

5. RESULTS

Here we present results for the two major experiments conducted in the park and campus environment and the highway and suburban roads day-night environment. For the park and campus environment, results were obtained using the low-resolution-only place recognition and patch-based visual odometry algorithms. Due to the significantly increased degree of perceptual change, the day-night road results were produced using the new combined low-resolution–patch-verification place recognition approach and filtered intensity profile-based visual odometry technique.

5.1. Park and Campus Environment

In this section, we present the visual odometry, place recognition, and mapping results as well as computational statistics.

5.1.1. Visual Odometry

Figure 13 shows the trajectory output by the patch-based visual odometry system for all four datasets, for the common starting pose of $(x, y, \theta) = (0\text{ m}, 0\text{ m}, 0°)$. Although the trajectories clearly do not match on a global scale, subsections of the route are similar for all four datasets, such as the small loop (sequence *EFGEF*) in Figure 10. The differences in the odometry-only trajectories were primarily caused by underestimation of yaw angles and translational speeds in the rain dataset, probably due to reflections in the water lying on the ground, and underestimation of the translational speed in the dusk dataset, due to the poor illumination and consequent lack of ground textures. The differences in translational speed calculations are most easily seen by looking at the length of the first section of each trajectory starting at (0,0) leading up to the first right turn.

5.1.2. Visual Place Recognition

Figure 14 displays a graph of the active (recognized or just learned) visual template versus frame number over all four datasets in the order they were processed, starting with the dawn dataset. The area of the graph below the dashed line is the area in which visual templates learned during the first dawn traverse of the environment were recognized during the subsequent datasets. The system was able to recognize places from the dawn dataset at regular intervals throughout the other three datasets. However, the graph also shows additional templates representative of the subsequent datasets being learned in parallel and bound to those locations in the map. Learning of new templates was due to the zigzag nature of much of the robot's movement through the environment, resulting in different image sequences each time a section was traversed.

5.1.3. Matched Frames

Figure 15 shows a selection of ten pairs of frames that were matched by the visual template system for locations throughout the entire route. The original video frames are shown for clarity purposes, although the actual processed images were 48×24 pixel patch-normalized images. The corresponding locations are shown in Figure 10. The visual system was able to match frames with significantly varying appearance due to (1, 3) sun flare, (2) obscuring leaf litter, (4) motion blur, (5–7) major shadow change, (3, 6, 9–10) large overall illumination change, and (10) water on the camera lens. The frames also show the challenge faced by the visual

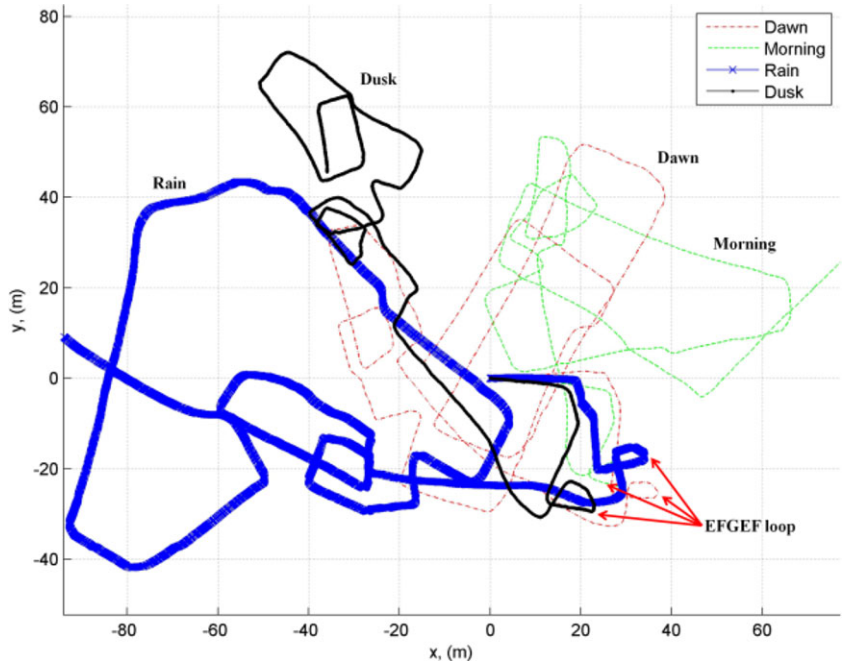

Figure 13. Vehicle trajectories calculated by the patch-based visual odometry system for the four datasets.

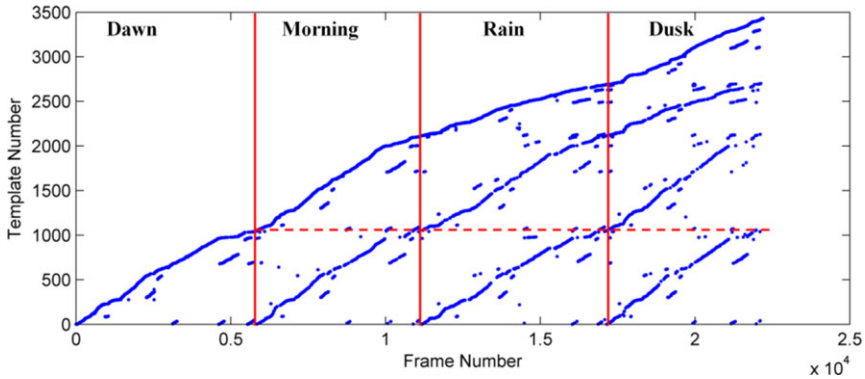

Figure 14. Visual template learning and recognition over the four datasets.

odometry system due to jerky vehicle motion (4) and lack of ground texture in low light (1, 3, 6–10).

5.1.4. Experience Maps

The final test of the system was to create a map of all four datasets. Figure 16 shows the evolution of the experience map after running through each dataset in order. The map is topologically correct after the dawn and morning datasets, although globally it is warped. The map shrinks slightly, primarily due to the underreporting of translational velocity in the dusk dataset and to a lesser extent the rain dataset. However, the constant loop closure within and across datasets ensures the map topology remains correct. The final map layout, although not metrically precise, has the correct topology. A video of the experience map and frame-matching processes is available online.[1]

5.1.5. SLAM with Only Visual Templates from a Single Time

To test the ability of the system to map and localize with only the visual templates learned at one particular time of day, we conducted an additional experiment in which template learning was disabled after the first dawn dataset. From that point onward, the visual template system either recognized a familiar template or reported no match, but it did

Figure 15. Matched visual templates over the four datasets. Corresponding locations are shown in Figure 10.

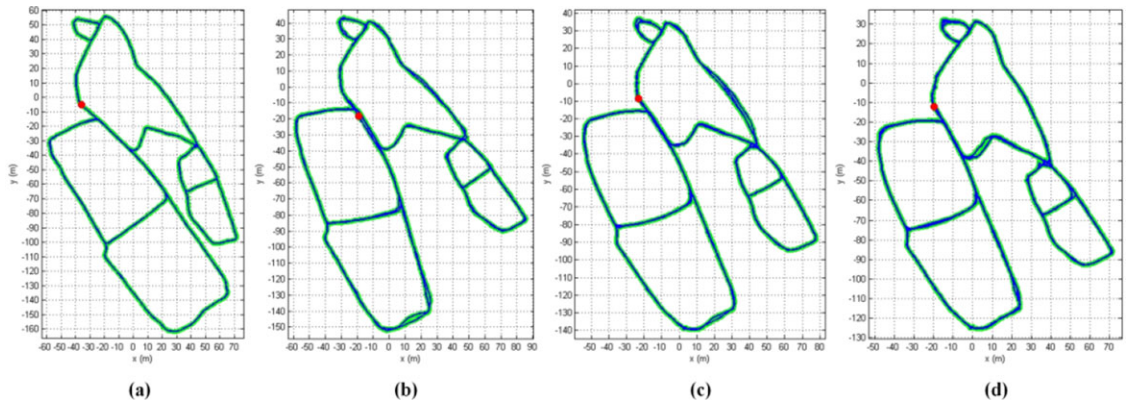

Figure 16. Experience map evolution over time. Experience maps are from after the (a) dawn, (b) morning, (c) rain, and (d) dusk datasets.

not learn any additional templates (Figure 17). Figure 18 shows the evolution of the experience map under these conditions. There are three locations where place recognition failed briefly (shown by dashed circles), all at places where the vehicle was turning corners and actual physical paths varied significantly. Although successful loop closures were achieved surrounding these points, the variation in visual odometry meant that the graph relaxation process was not able to draw these trajectories together to correctly overlap. The local topology in these areas is incomplete but correct, meaning navigation could still be achieved but might be suboptimal.

5.1.6. Computing and Storage

To demonstrate the feasibility of real-time performance on low-cost hardware, we present some pertinent computational statistics. The primary storage requirements come from the visual template library. Over all four datasets, a total of 3,353 templates were learned, taking up 5.8 MB of storage. With regard to computing, the system performs all computation on a fixed time basis, except for visual template comparison and experience map graph relaxation, which are both order O(N) [experience map graph relaxation approximates to order O(N) in a typical sparsely interconnected map]. Each of these two processes was run

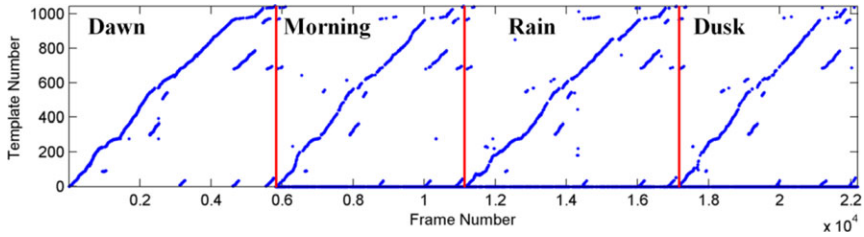

Figure 17. Visual template recognition performance with learning only enabled for the dawn dataset. Nonmatches where a template would normally be learned appear as number zero templates.

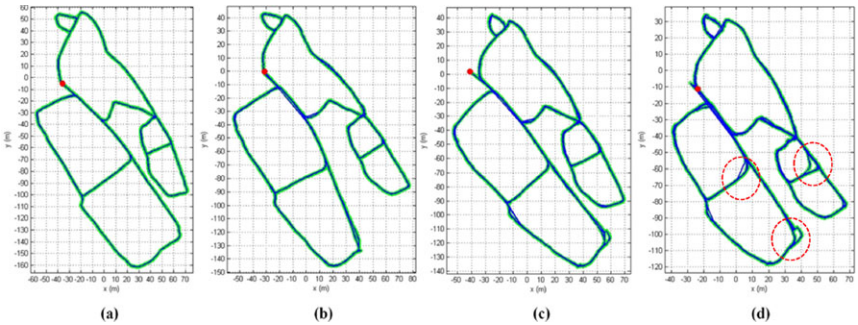

Figure 18. Experience map evolution with template learning disabled after the first dataset. Map shown after the (a) dawn, (b) morning, (c) rain, and (d) dusk datasets. Dashed circles indicate the three short-term localization failures.

on a separate CPU on a standard desktop PC. At the end of the dusk dataset when the system was at maximum load, the visual template system was performing 104 million pixel-to-pixel comparisons per second of data, which ran at real-time speed in unoptimized Matlab code. Experience map graph relaxation is performed with leftover computing cycles. At the end of the experiment, an average of 156 global graph relaxation iterations were performed per second of real-time. This figure can be compared with the eight iterations per second performed at the end of a previous indoor mapping experiment (Milford & Wyeth, 2010b), which was still sufficient to maintain a map that was used for robot navigation. A low-power onboard CPU [such as the 1 GHz processor on the robot shown in Figure 10(b)] should be capable of running the entire system in real-time for an environment of this size. The RatSLAM system used as the mapping backend has had lightweight versions implemented on a Lego Mindstorms NXT ("RatSLAM NXT, http://code.google.com/p/rsnxt08/," 2008) and a small mobile robot called the *iRat* (Ball, Heath, Milford, Wyeth, & Wiles, 2010), demonstrating the feasibility of running the system on a cheap platform.

5.2. Highway and Suburban Roads Day-Night Environment

In this section, we present visual odometry results, place-recognition precision-recall curves with and without patch verification and with the original SeqSLAM algorithm, recall curves at 100% and 99% precision for each of the three salience techniques over a range of salience mask fractions from 1% to 100%, stand-alone experience maps after the day run and then after the night run, and four patch verification examples demonstrating the ability of the system to correctly match places with a very challenging degree of perceptual change and to reject false positives created by the whole image-matching algorithm due to perceptual aliasing.

5.2.1. Filtered Intensity Profile Visual Odometry

The trajectories calculated by visual odometry for the day- and night-time runs are shown in Figure 19, with the two GPS tracks also shown. The starting points of all four tracks have been zeroed and aligned in orientation. Though the visual odometry tracks are not a perfect metric representation of the actual path, they capture the broad shape of the route and are reasonably consistent in terms of the relative length of sections between turns.

5.2.2. Single-frame Matching Recall and Precision

Figure 20 shows the precision-recall curves with (solid blue line) and without (dashed red line) patch verification, as well as using the original SeqSLAM algorithm (dot-dashed line). Due to the perceptual difficulty of the road dataset, the recall level at 100% precision using only low-resolution image matching is approximately 10%, and increasing the

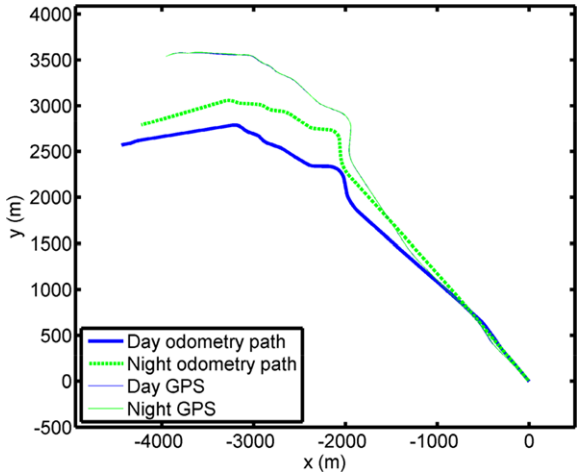

Figure 19. Vehicle trajectories calculated using the intensity profile visual odometry system compared to GPS.

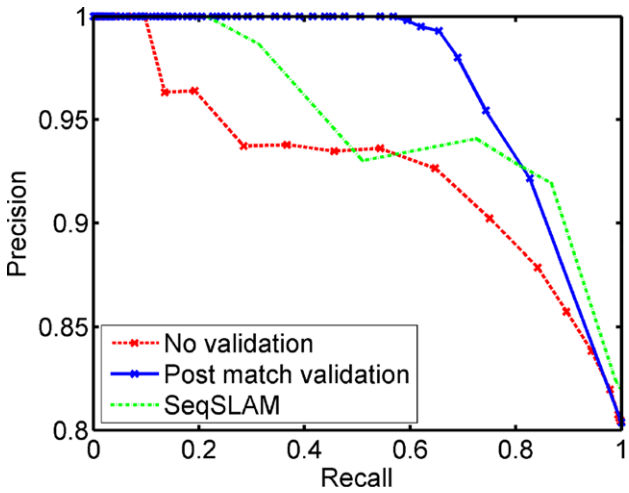

Figure 20. Precision recall curve with and without patch verification and with the original SeqSLAM algorithm. Note the range on the vertical precision axis.

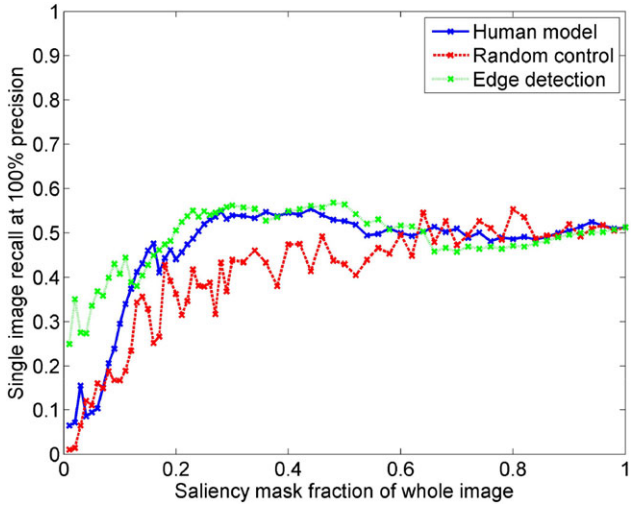

Figure 21. Single-frame recall at 100% precision with varying saliency mask fractions for the human, random, and edge-based masks.

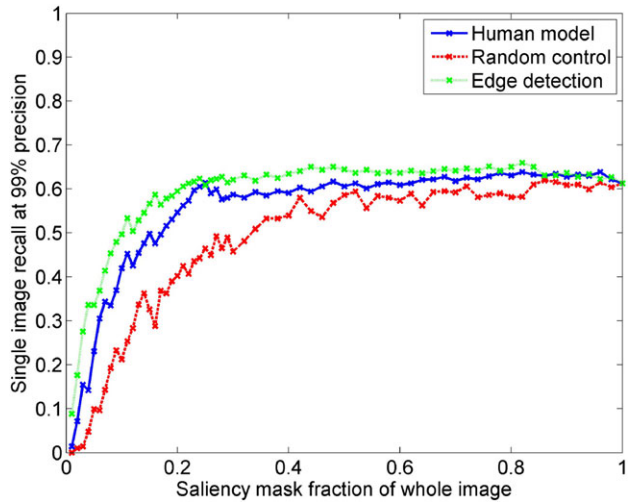

Figure 22. Single-frame recall at 99% precision with varying saliency mask fractions for the human, random, and edge-based masks.

recall rate leads to rapidly decreasing precision. Comparing recall rates at 100% precision, the addition of patch verification improves the recall rate from 10% to 56%, a more than fivefold improvement. Patch verification continues to makes a significant improvement to the recall rate achievable at any recall level up to 100%, where the two curves almost meet. SeqSLAM recall performance at 100% precision lies between single-image matching and the patch verification method, although at lower precision levels it matches the recall rate achieved by the patch-verification method.

Figures 21 and 22 show the recall rates achieved at 100% and 99% precision, respectively, when the edge-detection, human model, and randomized saliency masks are used to selectively perform patch verification over fractions of the image ranging from 1% to 100%.

At 100% precision, using a saliency mask fraction of only 20–40% for either the edge or human model masks leads to the best recall performance. Recall rates actually drop by about 10% (from an absolute recall rate of 56–51%) as a larger fraction of the mask is applied. Applying a fractional saliency mask, therefore, leads to an improvement in recall performance at 100% precision, but also to a significant reduction in post-validation computing, by reducing the number of patch verifications that need to be performed.

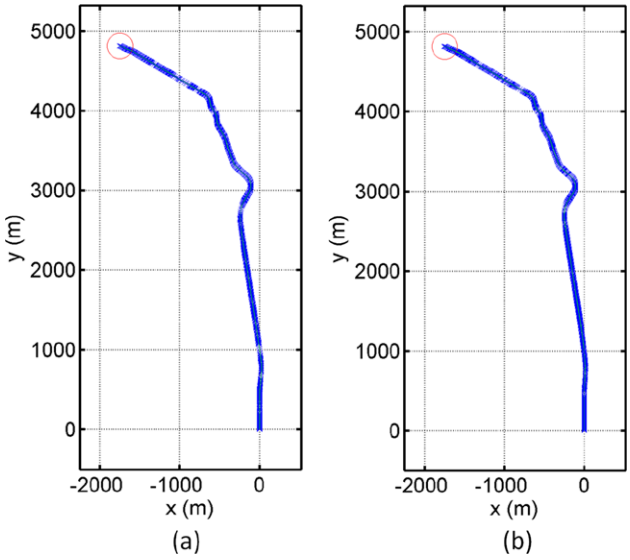

Figure 23. Experience maps after (a) the daytime run and (b) the nighttime run.

For example, applying a 20% edge-based saliency mask enables a fivefold speed increase in the patch-verification process and an improved recall level of 57%, compared to 51% when patch verification is performed over the entire image. Even with a random mask, performance almost plateaus when patch verification is calculated over 50% of the image. Interestingly, the performance of the edge-based mask at image fractions below 20% is superior to even the human-based mask, perhaps because an edge-detection calculation is more closely aligned with the method by which the patch-matching quality scores are calculated. Especially notable is the ability to recall 25–35% of all images with no errors when performing patch validation on only 1% or 2% of the total image, which corresponds to comparing only four to seven (out of 360) patch locations in the image.

At 99% precision, maximum recall performance is effectively reached when a 25% human or edge-based saliency mask is applied, although there is no drop off in recall rate as the mask fraction is increased beyond this size. Recall plateaus at approximately 64%.

5.2.3. Experience Maps

Figure 23 shows the graphical map created using the standalone experience mapping module, without any benefit of the RatSLAM neural networks and hence no false-positive rejection ability, using a 23% edge-based saliency mask at the 100% precision operating point. Figure 23(a) shows the map after the car completed the first day-time run, while Figure 23(b) shows the combined map after both day- and night-time runs. The map is consistent with the raw odometry signal and GPS trajectories. During the second night-time run, loop closures are continuously found back to the day-time run (as evidenced by the nicely overlapping paths) without any false-positive closures (as evidenced by the consistency in map layout and size during the night-time run).

5.2.4. Post-validation Match Samples

In this section, we show four examples of accepted and rejected image matches based on the patch-verification process. A video of 555 image matches confirmed by the patch-verification process is provided at the link given in footnote 1.

Figure 24(a) shows a frame from the night-time dataset along with the frame from the day-time dataset deemed to be the closest match by the low-resolution, whole image matcher [Figure 24(b)]. This match triggered the patch-verification process, which performed patch comparisons between the two images. A total of 57 patch matches that met the quality requirements described in Section 3.6 are shown by hollow black squares in a contrast-enhanced version of both images [Figures 24(c) and 24(d)], with the larger squares with dashed lines as borders in Figure 24(d) indicating the search space for that particular patch. The corresponding patch pairs are shown in Figure 24(e). The "loose" patch-matching requirements enable the verification process to propose patch matches that look quite different due to lighting changes, motion blur, and generally poor image quality. While inspection of each patch match suggests they are not all correct, for the patch-verification process to be useful it must only produce more patch matches (correct or not) for true image match pairs than for incorrect pairs.

Figure 25 shows the same details for a pair of images that were matched by the whole image matcher but then successfully rejected by the patch-verification process. These frames—both captured along a featureless highway—are a classical failure point for a low-resolution, whole image-matching approach because the low-resolution images appear identical. At low resolution the road surface, barrier wall, and even trees appear similar in appearance and in similar image locations across both images. However, the patch-verification process is only able to find 17 patch matches that meet the quality requirements, in contrast to the 57 found for the correctly matching image pair in Figure 24 and the 37 found for the correctly matching image pair found in a nearby part of the highway (Figure 26). Beyond the confirmation by GPS, the image matches in Figure 26 can be confirmed by the reader by examining the sign in the distant left of the image, the dashed white line in the right foreground, and the fine details in the tree at the top right of the frame.

Finally, Figure 27 demonstrates the patch-verification process enabling more spatially accurate matching than a low-resolution-only approach, by rejecting matches from images taken at nearby but not identical locations.

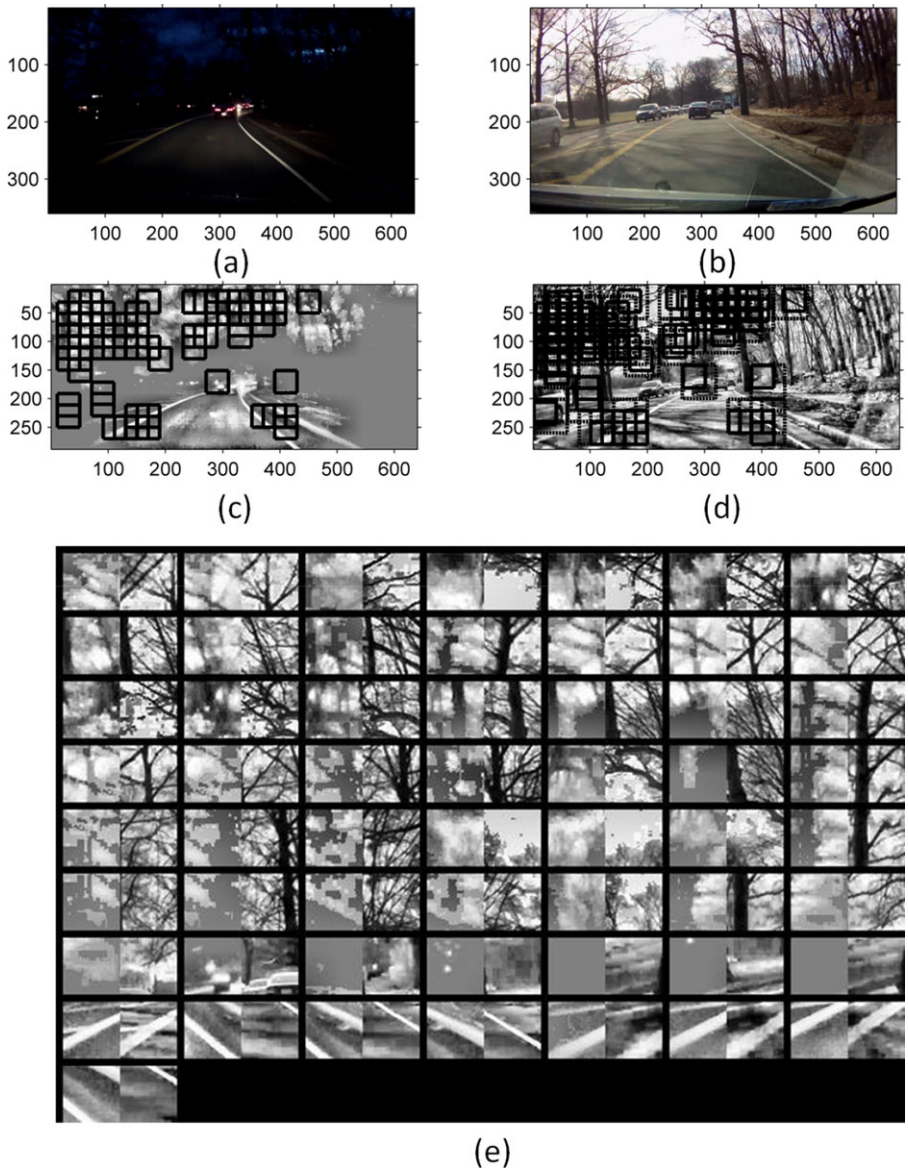

Figure 24. True positive match between images (a) and (b), confirmed by patch verification, which found patch matches at the square locations shown in (c) and (d). The corresponding patch pairs are shown in (e).

The image pair shown in Figures 27(a) and 27(b) was matched using the low-resolution image matcher, but the patch-verification process is only able to find 11 matching pairs. It is clear from a quick glance that the images are from the same general location, but closer examination reveals that the day-time image is from a location 10 or 20 m further along the route than the night-time image (look for the fire hydrant on the right side of the road and the building's relative location). In subsequent frames, the global image matcher did pick the correct match in this section of road, and the fire hydrant was picked as one of the patch-verification matches, although for purposes of brevity we merely describe the result.

5.2.5. Computation and Storage

The combined set of algorithms when using edge-based saliency masks ran at real-time speed or better. Here we briefly discuss the computational load of components beyond those already detailed for the previous experiment, which ran at real-time speed or better and involved a larger map and hence larger computational load.

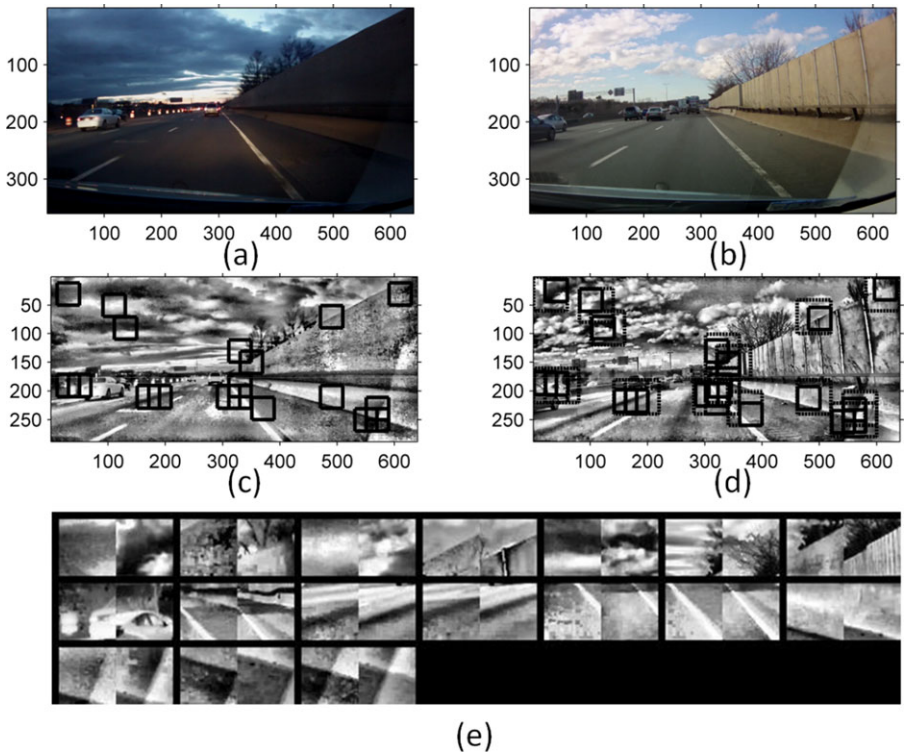

Figure 25. False positive match (as output by the whole image matcher) between images (a) and (b) correctly detected by patch verification, which only found 17 patch matches at the square locations shown in (c) and (d). The corresponding patch pairs are shown in (e).

The saliency mask calculation and patch-verification processes add varying degrees of computational complexity. The edge-based mask calculation was performed offline but is a trivial calculation and could easily be performed as each new image was obtained. The human-based saliency mask is currently not optimized for online operation and was performed offline at less than real-time speed. The patch-verification process scales linearly with the number of images stored by the system, and it is performable at real-time speed even without the use of a saliency mask to improve computation. The calculation below gives the approximate computational requirements for real-time patch-verification operation in the experiment presented in this paper:

$$\begin{aligned} &40\ \textit{pixels} \times 40\ \textit{pixels} \times 20\ \textit{xshift} \times 20\ \textit{yshift} \\ &\quad \times 30\ \textit{patches across}\ 12\ \textit{patches down} \times 1\ \textit{frame/s} \\ &\quad = 230 \times 10^6\ \textit{pixel comparisons per second}. \end{aligned}$$

Although the code running the algorithms is not fully optimized, generally speaking at least a one-to-one ratio between 8-bit pixel comparisons and nominal computer clock speed is achievable, suggesting the current patch-verification method might scale to performing patch verification in real-time on the 10 top match candidates output by the whole image-matching algorithm without resorting to GPU-based computation.

Finally, the computational load of calculating low-resolution image matches is detailed in Milford (2013). Computations scale linearly with the number of images stored and the square of the degree of pose invariance that is required by the matching process. Using low-resolution imagery means that image databases storing enough imagery to map a city can be searched in real-time on a standard PC (Milford, 2013).

6. DISCUSSION

This paper has presented a range of approaches to enabling vision-based place recognition and odometry in challenging or changing perceptual conditions using relatively cheap consumer-grade camera equipment. The visual processing techniques require no prior training,[2] and they were

[2]No training is required to generate a topological map. To obtain a map with absolute scale, a short calibration of the translational gain constant is required if camera height and parameters are not known.

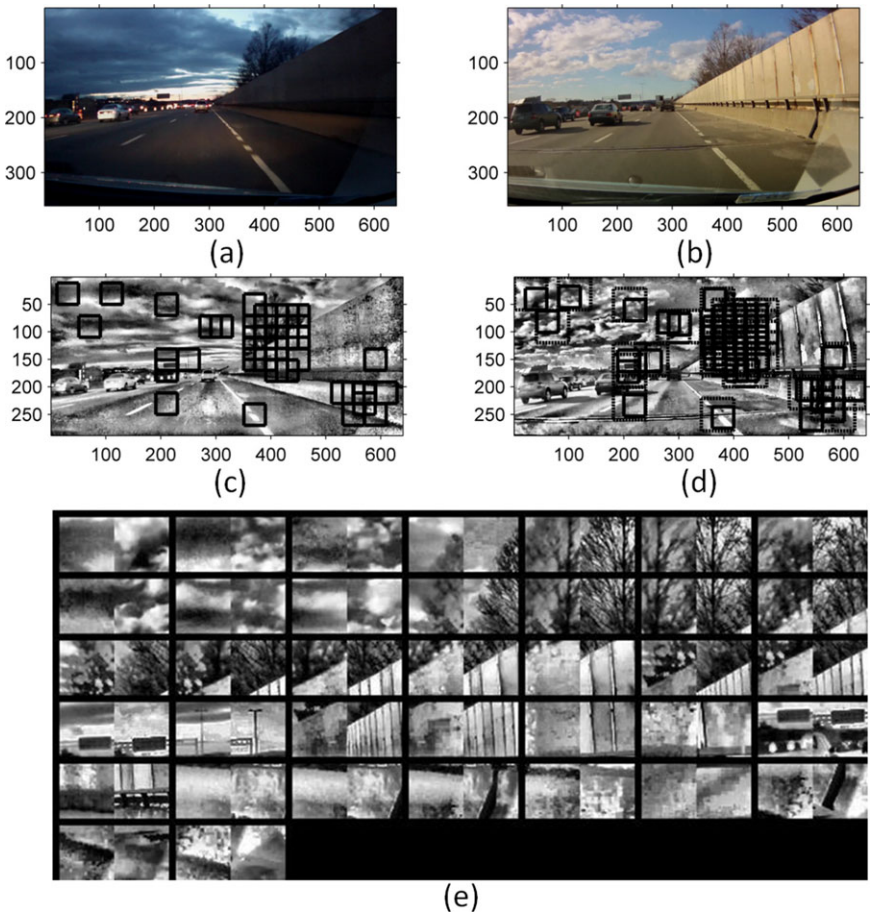

Figure 26. True positive match between images (a) and (b) confirmed by patch verification, which found 37 patch matches at the square locations shown in (c) and (d). The corresponding patch pairs are shown in (e).

demonstrated to enable topological mapping with consumer cameras in a varied vegetated and urban environment as well as across a day-night cycle on a road-based dataset.

The introduction of a patch verification routine to assess the matches proposed by a low-resolution whole image matcher led to very significant improvements in place recognition performance, with further improvement in performance and reductions in computing achieved by utilizing salience masks calculated on a per image basis. The patch verification approach is successful because, perhaps like other verification techniques such as geometric verification (Cummins & Newman, 2009), it is able to reliably detect the small number of false-positive matches reported by the low-resolution whole image matcher without significantly reducing the number of true positives. In addition, the high recall at 100% precision provided by patch verification removes the restrictive sequence requirements of past low-resolution image-matching methods (Milford & Wyeth, 2008, 2012), enabling the possibility of instantaneous localization when crossing a previously traversed path, rather than just when following along it.

Here we discuss some limitations of the presented approach and areas for future work. We used a forward-facing camera only, and hence we had no ability to close the loop when retracing a route in the opposite direction. However, past work has demonstrated that such similar place recognition algorithms can be adapted to utilize omnidirectional imagery (Milford & Wyeth, 2010b; Prasser, Milford, & Wyeth, 2005). The ability of the system to function with relatively low-resolution imagery would also be likely to enable the combination of cheap and compact panoramic imaging rigs with a low-cost camera (the mirror/lens could be mass-produced with loose specifications, because camera calibration is not a concern, unlike for many traditional vision-based algorithms). For example, a $5 panoramic lens

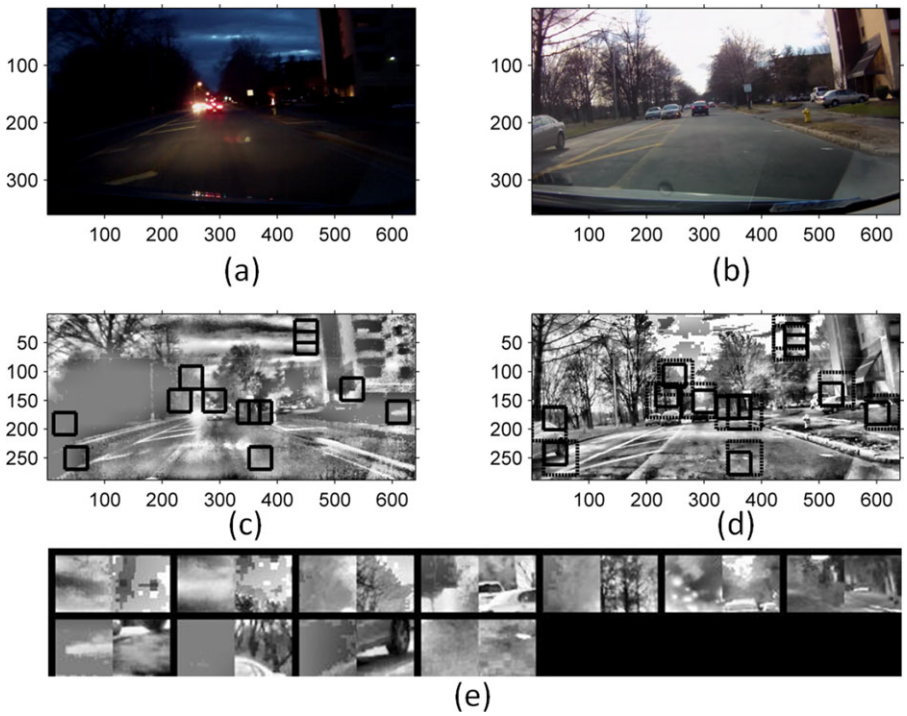

Figure 27. "Close" false positive match between images (a) and (b) correctly discarded by patch verification, which found patch matches at the square locations shown in (c) and (d). The corresponding patch pairs are shown in (e).

when combined with a standard mass-produced phone sensor produces panoramic video at an effective resolution of approximately 960×160 pixels, similar in total dimensions to the images used in these studies. In contrast, much current robot research makes use of high-end panoramic imaging setups such as the Point Grey Ladybug 2 (~10,000 USD). Alternatively, two perspective cameras mounted in opposite directions along the primary vehicle axis would provide forward-backward recognition capability.

The current place recognition system is not suited to open-field operation in large open environments where movement is unrestricted and paths are not necessarily repeated by a robot. However, this restriction is also present in many vision-based SLAM systems developed to date. One approach to overcoming this limitation is to combine a SLAM system with absolute positioning information provided by GPS, when available (Thrun & Montemerlo, 2006). It is interesting to note that GPS availability and the presented visual SLAM method's viability tend to be complementary, at least for the park campus dataset presented in this paper. When the GPS signal was most degraded, the vehicle was usually traveling along urban canyons or under trees on offroad paths where paths are constrained, situations in which the presented approach works well. In addition, visual homing methods tend to work well in large open spaces with distal visual cues (Sturzl & Zeil, 2007).

Future work will pursue a number of research directions beyond those already mentioned above. The first will be to further optimize the patch verification algorithm, which is predicted to be the computationally limiting factor as environments get larger. One seemingly obvious speedup would be to deploy the algorithm on a GPU, but the simplicity of the primary operation (sum of absolute differences) means that gaining performance is critically dependent on memory bandwidth management.

Secondly, we will investigate how to provide a higher degree of pose invariance. One of the key advantages of traditional feature-based, "bottom up" approaches is that scenes can be recognized from a range of significantly different camera poses, at least in ideal conditions. In contrast, the presented approach is only able to achieve a limited degree of pose invariance by performing low-resolution, whole image matching over a range of image offsets, a method that works well until camera pose changes result in significantly nonaffine image transformations. However, even then image comparison has been shown to degrade gracefully, especially when using panoramic images (Sturzl & Zeil, 2007). By expanding the patch verification process to include more of the top-ranked image matches, and

introducing some form of deformable graph over which patch matching is performed, it may be possible to achieve significantly greater degrees of pose invariance, at the cost of increased computational load. Expanding the patch verification search to more proposed image match pairs also has the advantage of enabling the system to correctly match false negatives missed by the initial whole image matcher.

Thirdly, we have only addressed perceptual change due to day-night and weather cycles in this paper—there are other challenges, such as seasonal change (Sünderhauf et al., 2013), that we will need to investigate, especially phenomena including snow and defoliation of vegetation in winter.

Lastly, the quality of the maps presented here is comparable to or better than those previously used successfully for robot navigation (M Milford & Wyeth, 2010b), suggesting that with the addition of a local obstacle-avoidance module, navigation using these maps is feasible. We will investigate combining state-of-the-art local obstacle avoidance techniques with existing global path-planning algorithms (M Milford & Wyeth, 2010b) in order to conduct active navigation experiments under challenging and changing environmental conditions. In particular, we will examine the utility of applying these algorithms on small platforms such as automated RC vehicles equipped with low-cost visual sensors. The current characteristics of the algorithm make it relatively well-suited to activities such as teach and repeat (Furgale & Barfoot, 2010) along similar repeated paths, with the eventual aim of achieving fully autonomous SLAM and navigation, day or night, and in fine or rainy weather.

ACKNOWLEDGMENTS

This work was supported by an Australian Research Council Discovery Early Career Researcher Award DE120100995 to M.M. and partially supported by a fellowship in the Postdoc-Programme of the German Academic Exchange Service (DAAD, D/11/41189) to E.V. We thank Ashley George for his help in gathering the park and campus datasets, and Mike Burns for the day-night road datasets.

REFERENCES

Andreasson, H., Duckett, T., & Lilienthal, A. (2007). Mini-SLAM: Minimalistic visual SLAM in large-scale environments based on a new interpretation of image similarity. Paper presented at the International Conference on Robotics and Automation, Rome.

Andreasson, H., Duckett, T., & Lilienthal, A. (2008). A minimalistic approach to appearance-based visual SLAM. IEEE Transactions on Robotics, 24(5), 1–11.

Ball, D., Heath, S., Milford, M., Wyeth, G., & Wiles, J. (2010). A navigating rat animat (pp. 804–811). Cambridge, MA: MIT Press.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features Computer Vision–ECCV (pp. 404–417).

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. Paper presented at the Advances in Neural Information Processing Systems, Montreal, Quebec, Canada.

Burschka, D., & Hager, G. D. (2004). V-GPS (SLAM): Vision-based inertial system for mobile robots. In Robotics and Automation, Proceedings ICRA'04. 2004 IEEE International Conference, Vol. 1. IEEE, Washington, DC.

Clark, R. N. (2005). Digital cameras: Does pixel size matter? Factors in choosing a digital camera (does sensor size matter?). http://www.clarkvision.com/articles/does.pixel.size.matter/.

Cox, D., & Pinto, N. (2011). Beyond simple features: A large-scale feature search approach to unconstrained face recognition. Paper presented at the IEEE International Conference on Automatic Face & Gesture Recognition and Workshops, Santa Barbara, CA.

Cummins, M., & Newman, P. (2009). Highly scalable appearance-only SLAM–FAB-MAP 2.0. Paper presented at the Robotics: Science and Systems, Seattle.

Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6), 1052–1067.

Furgale, P., & Barfoot, T. D. (2010). Visual teach and repeat for long-range rover autonomy. Journal of Field Robotics, 27(5), 534–560.

Glover, A. J., Maddern, W. P., Milford, M. J., & Wyeth, G. F. (2010). FAB-MAP + RatSLAM: Appearance-based SLAM for multiple times of day. Paper presented at the International Conference on Robotics and Automation, Anchorage, AK.

Johns, E., & Yang, G. Z. (2013). Feature co-occurrence maps: Appearance-based localisation throughout the day. Paper presented at the International Conference on Robotics and Automation, Karlsruhe, Germany.

Judd, T., Ehinger, K., & Duran, F. D. (2009). Learning to predict where humans look. Paper presented at the International Conference on Computer Vision, Kyoto, Japan.

Kiyoshi, I., Tomoaki, Y., & Masahiro, T. (2011). A high dynamic range vision approach to outdoor localization. Paper presented at the International Conference on Robotics and Automation, Shanghai, China.

Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small AR workspaces. Paper presented at the International Symposium on Mixed and Augmented Reality, Nara, Japan.

Klein, G., & Murray, D. (2008). Improving the agility of keyframe-based SLAM. European Conference on Computer Vision (pp. 802–815).

Konolige, K., & Agrawal, M. (2008). FrameSLAM: From bundle adjustment to real-time visual mapping. IEEE Transactions on Robotics, 24(5), 1066–1077.

Konolige, K., Agrawal, M., Bolles, R., Cowan, C., Fischler, M., & Gerkey, B. (2008). Outdoor mapping and navigation using stereo vision. In Khatib, O., Kumar, V., & Rus, D. (eds.), Experimental robotics (pp. 179–190) Springer: Berlin-Heidelberg.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. Paper presented at the International Conference on Computer Vision, Kerkyra, Greece.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), 91–110.

Milford, M. (2011). Towards condition-invariant sequence-based route recognition. Paper presented at the Australasian Conference on Robotics and Automation, Melbourne, Australia.

Milford, M. (2013). Vision-based place recognition: How low can you go? International Journal of Robotics Research, 32(7), 766–789.

Milford, M., & George, A. (2012). Featureless visual processing for SLAM in changing outdoor environments. Paper presented at the International Conference on Field and Service Robotics, Matsushima, Japan.

Milford, M., Schill, F., Corke, P., Mahony, R., & Wyeth, G. (2011). Aerial SLAM with a single camera using visual expectation. Paper presented at the International Conference on Robotics and Automation, Shanghai, China.

Milford, M., & Wyeth, G. (2008). Mapping a suburb with a single camera using a biologically inspired SLAM system. IEEE Transactions on Robotics, 24(5), 1038–1053.

Milford, M., & Wyeth, G. (2009). Persistent navigation and mapping using a biologically inspired SLAM system. The International Journal of Robotics Research, 29(9), 1–23.

Milford, M., & Wyeth, G. (2010a). Improving recall in appearance-based visual SLAM using visual expectation. Paper presented at the Australasian Conference on Robotics and Automation, Brisbane, Australia.

Milford, M., & Wyeth, G. (2010b). Persistent navigation and mapping using a biologically inspired SLAM system. International Journal of Robotics Research, 29(9), 1131–1153.

Milford, M., & Wyeth, G. (2012). SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. Paper presented at the IEEE International Conference on Robotics and Automation, St. Paul, MN.

Milford, M. J. (2008). Robot navigation from nature: Simultaneous localisation, mapping, and path planning based on hippocampal models (Vol. 41). Berlin-Heidelberg: Springer-Verlag.

Milford, M. J., Prasser, D., & Wyeth, G. (2005). Experience mapping: Producing spatially continuous environment representations using RatSLAM. Paper presented at the Australasian Conference on Robotics and Automation, Sydney, Australia.

Neubert, P., Sünderhauf, N., & Protzel, P. (2013). Appearance change prediction for long-term navigation across seasons. Paper presented at the European Conference on Mobile Robots.

Olson, E., Leonard, J., & Teller, S. (2006). Fast iterative alignment of pose graphs with poor initial estimates. Paper presented at the International Conference on Robotics and Automation, Orlando, FL.

Paz, L. M., Pinies, P., Tardos, J. D., & Neira, J. (2008). Large-scale 6-DOF SLAM with stereo-in-hand. IEEE Transactions on Robotics, 24(5), 946–957.

Pinto, N., Doukhan, D., DiCarlo, J., & Cox, D. (2009). High-throughput screening approach to discovering good forms of biologically inspired visual representation. PLoS Computational Biology, 5(11), doi: 10.1371/journal.pcbi.1000579.

Prasser, D., Milford, M., & Wyeth, G. (2005). Outdoor simultaneous localisation and mapping using RatSLAM. Paper presented at the International Conference on Field and Service Robotics, Port Douglas, Australia.

RatSLAM NXT (2008). http://code.google.com/p/rsnxt08/.

Royer, E., Bom, J., Dhome, M., Thuilot, B., Lhuillier, M., & Marmoiton, F. (2005). Outdoor autonomous navigation using monocular vision. Paper presented at the IEEE International Conference on Intelligent Robots and Systems.

Sturzl, W., & Zeil, J. (2007). Depth, contrast and view-based homing in outdoor scenes. Biological Cybernetics, 96(5), 519–531.

Sünderhauf, N., Neubert, P., & Protzel, P. (2013). Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons. Paper presented at the International Conference on Robotics and Automation, Karlsruhe, Germany.

Thalin, G. (2010). Deshaker—Video stabilizer (version 2.5).

Thrun, S., & Montemerlo, M. (2006). The GraphSLAM algorithm with applications to large-scale mapping of urban structures. The International Journal of Robotics Research, 25(5–6), 403–429.

Vidas, S., & Maddern, W. (2012). Towards robust night and day place recognition using visible and thermal imaging. Paper presented at the Beyond Laser and Vision: Alternative Sensing Techniques for Robotic Perception Workshop at RSS2012, Sydney, Australia.

Whelan, T., McDonald, J., Kaess, M., Johannsson, M. F. H., & Leonard, J. J. (2012). Kintinuous: Spatially extended KinectFusion. Paper presented at the RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras, Sydney, Australia.

Zhang, A. M., & Kleeman, L. (2009). Robust appearance based visual route following for navigation in large-scale outdoor environments. The International Journal of Robotics Research, 28(3), 331–356.