CAT-SLAM: Probabilistic Localisation and Mapping using a Continuous Appearance-based Trajectory

Will Maddern, Michael Milford and Gordon Wyeth
School of Electrical Engineering and Computer Science, Queensland University of Technology
{w.maddern, michael.milford, gordon.wyeth}@qut.edu.au

Abstract

This paper describes a new system, dubbed Continuous Appearance-based Trajectory SLAM (CAT-SLAM), which augments sequential appearance-based place recognition with local metric pose filtering to improve the frequency and reliability of appearance based loop closure. As in other approaches to appearance-based mapping, loop closure is performed without calculating global feature geometry or performing 3D map construction. Loop closure filtering uses a probabilistic distribution of possible loop closures along the robot's previous trajectory, which is represented by a linked list of previously visited locations linked by odometric information. Sequential appearance-based place recognition and local metric pose filtering are evaluated simultaneously using a Rao-Blackwellised particle filter, which weights particles based on appearance matching over sequential frames and the similarity of robot motion along the trajectory. The particle filter explicitly models both the likelihood of revisiting previous locations and exploring new locations. A modified resampling scheme counters particle deprivation and allows loop closure updates to be performed in constant time for a given environment. We compare the performance of CAT-SLAM to FAB-MAP (a state-of-the-art appearance-only SLAM algorithm) using multiple real-world datasets, demonstrating an increase in the number of correct loop closures detected by CAT-SLAM.

Keywords

Appearance-based SLAM, vision-based robot navigation, CAT-SLAM

1. Introduction

In recent years appearance-based localisation systems have gained in popularity as a method for loop closure detection in metric SLAM systems. So-called 'appearance-based SLAM' systems represent the environment as a series of 'snapshots' from discrete locations and typically calculate image similarity based on extracted feature descriptors, such as those generated by SIFT (Lowe 1999) or SURF (Bay et al. 2006). Since visual appearance-based methods rely only upon the similarity between images from two locations, they can perform loop closure detection regardless of accumulated metric error: a common cause of failure for geometric SLAM systems (Thrun and Leonard 2008). Appearance-based SLAM systems augment visual localisation methods with the ability to determine whether an observation comes from a previously unvisited location, effectively performing SLAM in an 'appearance-space'.

Development of appearance-based SLAM systems has primarily focused on increasing the number of previously visited locations that are recognized (high recall) while maintaining low numbers of false positives (high precision). As false positive loop closures cause corruption in most mapping systems, 100% precision (zero false positives) is a common requirement for appearance-based loop closure detection. The most successful appearance-based SLAM algorithm to date is FAB-MAP (Cummins and Newman 2008). A rigorous probabilistic approach to image matching based on a

This research was supported by an Australian Research Council Special Research Initiative on Thinking Systems, TS0669699, to GW, and an Australian Postgraduate Award, to WM. W. Maddern, M. Milford and G. Wyeth are with the School of Electrical Engineering and Computer Science, Faculty of Science and Engineering, Queensland University of Technology, Brisbane QLD 4001 Australia (e- mail: {w.maddern, michael.milford, gordon.wyeth}@qut.edu.au).

'visual bag-of-words' model has allowed FAB-MAP to perform localisation on trajectories up to 1000km in length (Cummins and Newman 2009).

Attempts to improve the precision-recall performance of appearance-based SLAM algorithms such as FAB-MAP typically require additional information not provided by descriptor-based image similarity alone; (Konolige and Agrawal 2008; Cummins and Newman 2009) use RANSAC to compare feature geometry, while (Konolige et al. 2009; Newman et al. 2009; Sibley et al. 2010) use additional laser or stereo image sensors for 3D geometric verification. These methods still rely on matching two distinct locations using appearance alone – they discard the motion information between locations (provided by vehicle odometry) and the sequence in which the locations were visited.

This paper presents Continuous Appearance-based Trajectory SLAM (CAT-SLAM), a probabilistic approach to sequential appearance-based loop closure detection combining the spatial filtering characteristics of traditional geometric SLAM algorithms with the appearance-based place recognition of FAB-MAP. CAT-SLAM represents the map as a continuous trajectory which traverses all previously visited locations, where appearance is represented continuously along the trajectory, rather than at discrete points. Loop closure hypotheses are developed over a number of observations using a Rao-Blackwellised particle filter, which weights particles based on local trajectory-constrained odometry information and appearance-based observation likelihoods.

Like FAB-MAP, CAT-SLAM requires very few parameters beyond those obtained from properties of the sensor platform. The probabilistic formulation of CAT-SLAM makes it straightforward to characterise and deploy to different platforms in different environments, as minimal tuning is required. Since CAT-SLAM uses a constant number of particles, computation time does not increase linearly with the number of previously visited locations; loop performance is maintained over large environments provided the modified particle resampling scheme ensures sufficient particle diversity.

We evaluate the loop closure performance of CAT-SLAM in comparison to FAB-MAP using three datasets: a 500m indoor dataset, a 2.5km urban dataset and a 15km outdoor dataset. The urban dataset and outdoor dataset have previously been used for various FAB-MAP experiments (Smith et al. 2009; Glover et al. 2010), and the indoor dataset for long term robot navigation experiments in (Milford and Wyeth 2009). The differences in sensor platform, environment and scale of these datasets serve to highlight the consistent high performance of CAT-SLAM across a wide range of applications with minimal changes in algorithm parameters. In contrast to recent larger-scale datasets used for FAB-MAP evaluations in (Cummins and Newman 2009; Cummins and Newman 2010) which typically only revisit locations once, these three datasets feature significant numbers of repeated loop closures (more than 10 in areas of the indoor dataset), representative of long-term operations in a single environment.

The following section outlines the relevant literature in the field of appearance-based SLAM. Section 3 describes the fundamental components of two SLAM systems, particle filter SLAM and FAB-MAP, to provide the relevant background theory for the derivation of CAT-SLAM. Section 4 presents the CAT-SLAM algorithm with an implementation specific to loop closure detection. The datasets and algorithm parameters for FAB-MAP and CAT-SLAM used for evaluation are presented in Section 5, with the results in the following section. We conclude with a discussion of the advantages and disadvantages of the CAT-SLAM system over FAB-MAP for loop closure detection.

Initial results for the 2.5km urban dataset and the 15km outdoor dataset were originally presented in (Maddern et al. 2011) and (Maddern et al. 2010) respectively; in this paper we provide a more thorough explanation of the algorithm and further results on an additional dataset.

2. Related Work

The majority of current state-of-the-art SLAM systems are based on a geometric interpretation of the SLAM problem. Geometric SLAM systems employ probabilistic algorithms such as Kalman filters (Dissanayake et al. 2001), Expectation Maximisation (Thrun and Montemerlo 2006) and Rao-Blackwellised particle filters (Montemerlo et al. 2002). These techniques are now well characterised, but their reliance on geometric consistency causes them to become computationally expensive and fragile when building large maps (Thrun and Leonard 2008).

To avoid computational and scaling limitations, a number of SLAM approaches forsake geometric accuracy for flexibility to form semi-metric or non-metric 'topological' approaches. Instead of attempting to combine all features from the environment in a single Euclidean space, non-geometric approaches typically form loosely-connected sub-maps (Bosse et al. 2003) or reduced topological maps (Konolige and Agrawal 2008). Although the maps generated by these algorithms tend not to provide the metric accuracy of full SLAM systems, they provide the robot with the ability to localise and navigate successfully, which is the fundamental purpose of a map for autonomous robot applications (Milford and Wyeth 2009).

Data association in these algorithms is often performed using a standalone loop closure detection system. Loop closure algorithms typically fall under one of the following categories: loop closure using appearance-only information, loop closure using appearance and geometry, and loop closure using appearance and motion sequences.

A. Loop closure with appearance only

Loop closure detection systems that operate using only appearance information typically make use of methods derived from the visual bag-of-words approach originally presented in (Sivic and Zisserman 2003). Features extracted from images of a particular location are classified according to a database of unique descriptors, known as 'visual words'. The map is represented by a set of binary vectors describing which visual words are present at each location. The visual bag-of-words approach has received considerable attention in the field of computer vision and image retrieval (Nister and Stewenius 2006).

The FAB-MAP system (Cummins and Newman 2008) consists of two major components: the visual word database (or codebook) and a Chow-Liu dependency tree. These components are constructed using prior data from a training environment and a method of comparing the current visual bag-of-words to all previously visited locations using recursive Bayesian estimation. Additionally, FAB-MAP is capable of determining whether the current location corresponds to a previously visited location in the map or a new location. The largest scale appearance-based SLAM experiment performed to date is a 1000km road network successfully mapped in appearance-space by FAB-MAP 2.0 (Cummins and Newman 2009; Cummins and Newman 2010). The FAB-MAP algorithm is described in detail in Section 3.

FAB-MAP has been used as a component of a number of metric SLAM systems to provide loop closure candidates. (Newman et al. 2009) describes a complete mapping and 3D reconstruction system using stereo and omnidirectional vision and trawling laser scanners. FAB-MAP is used to provide a set of loop closure candidates using the omnidirectional camera which are subject to geometric verification using RANSAC on stereo point clouds. The loop closures detected by FAB-MAP are incorporated into the metric map for locally optimal trajectory and map estimation. In a similar vein but on a larger scale, FAB-MAP 2.0 is used to detect loop closure candidates in over 140km of stereo data to correct a metric map formed using adaptive relative bundle adjustment (Sibley et al. 2010).

A number of other algorithms exist that use appearance-only information to detect loop closure. (Angeli et al. 2009) uses BayesianLCD, similar to the naive Bayes approach discussed in (Cummins and Newman 2007), to provide loop closure information for a lightweight topological mapping system. This system is extended in (Angeli et al. 2008) to learn the bag-of-words model online, generating a topological map of an outdoor location of approximately 1 km^2. A similar online bag-of-words method is used in (Eade and Drummond 2008) to provide loop closure candidates to a graphical visual SLAM system, closing loops in small indoor and outdoor environments.

PIRF-Nav (Kawewong et al. 2010) uses a dynamic bag-of-words method to build place representations without requiring training data. Descriptors that are stable over multiple frames are stored as robust scene identifiers, and loop closures are detected if a similarity metric between sets of features in an image pair exceeds a threshold. While PIRF-Nav is robust to dynamic objects in environments, it requires a consistent frame rate to generate robust descriptors and a consistent speed to recognise previously visited locations. A similar dynamic bag-of-words method is presented in (Mei et al. 2010) which clusters landmarks based on co-visibility, avoiding the issue of arbitrary discretisation of space. A tf-idf classifier is used to rank loop closure candidates. While both these methods have been demonstrated to outperform FAB-MAP on medium-scale (1-5km) outdoor datasets, both require a number of parameters to function in a given environment, and to date neither have been used to detect loop closures for a map construction algorithm.

As all the above methods rely on a similarity metric between two static appearance 'snapshots', they are subject to perceptual aliasing; they are unable to distinguish between locations in the environment which are similar in appearance yet occur in different locations. The remainder of this section discusses common approaches to augmenting appearance-based matching with additional information to reduce the likelihood of perceptual aliasing.

B. Loop closure with appearance and local geometry

An alternative approach to matching locations based entirely on appearance information is to include the spatial configuration of features in the comparison step. This is typically performed using a variant of the RANSAC algorithm, with the added advantage of providing relative pose geometry if a successful correspondence is found.

The FrameSLAM system in (Konolige and Agrawal 2008) combines accurate stereo visual odometry with frame-based visual matching and pose graph optimisation. Data association in FrameSLAM is performed with the use of CenSure, a lightweight equivalent of SURF (Agrawal et al. 2008). Loop closure detection in FrameSLAM is performed using 3-point RANSAC on feature locations in the image plane. While FrameSLAM provides real-time mapping performance using only visual information, its metric accuracy primarily depends on robust visual odometry provided by stereo cameras. Additionally, because it discards feature appearance information it is more susceptible to failure in environments with repetitive geometric structures. Further enhancements to FrameSLAM in (Konolige et al. 2009) add a vocabulary-tree based appearance match to provide candidate loop closures to the RANSAC stage.

An interesting special case of geometric-based appearance matching is presented in (Bosse and Zlot 2008). Descriptor-like keypoints are extracted from LIDAR data, and a histogram-based matching system is used to select candidate loop closures. These are evaluated using the map-matching component of the Atlas framework, a sub-mapping based full SLAM algorithm. While the keypoints are processed in the manner of descriptors, they are generated by local structural features of the environment, and thus fall under the category of appearance and geometric matching. This approach has been extended to 3D LIDAR maps in (Bosse and Zlot 2010).

(Paul and Newman 2010) extend FAB-MAP by including both spatial layout and range information in the location model. The implementation, dubbed FAB-MAP 3D, uses a random graph method to

represent location and accelerates comparisons of spatial layout using a Delaunay tessellation. Although FAB-MAP 3D is demonstrated to provide significant advantages over FAB-MAP in terms of recall at 100% precision, it requires additional sensors to detect feature range (in this case a laser rangefinder). A less sophisticated geometric method to augment FAB-MAP is presented in (Cummins and Newman 2009), where a 1-D RANSAC stage is used to verify the best 100 appearance-only matches. This geometric post-verification is especially important in the 1000km road network experiment, significantly increasing the number of frames recalled at 100% precision.

The combination of geometric comparison and appearance matching serves to significantly reduce the likelihood of perceptual aliasing; however, some environments may exhibit repetitive locations that are identical in both structure and appearance. In these cases, which are particularly common in man-made indoor and urban environments, matching the current location to a single previous observation remains insufficient to accurately detect loop closure; more information in the form of motion and frame sequence information is required as described in the following section.

C. Loop closure with appearance and motion sequences

A number of recent algorithms in the field of probabilistic topological mapping approach loop closure and map construction as two parts of the same problem. The approach of (Olson 2008) finds the optimal set of local metric and appearance information in the current map that best matches the current set of observations and local motion. This system has been evaluated in an urban environment of approximately 2km in length, and is equally applicable to both vision and laser data. (Blanco et al. 2008) describes a system where both local metric maps and topological position are used to determine the current location within the hybrid map. This approach is extended to a general formulation in (Ranganathan and Dellaert 2011) using a Rao-Blackwellised particle filter for efficient localisation and map construction and evaluated in medium-scale environments.

The recursive Bayesian estimation stage in FAB-MAP includes a location prior, typically implemented by boosting the likelihoods of matches to frames adjacent to the previously matched frame. While this motion prior is described as weak in (Cummins and Newman 2008), it increases recall at 100% precision by up to 50% on the larger datasets in (Cummins and Newman 2009). These results seem to imply that a more sophisticated motion model can provide further improvements to FAB-MAP, a point also raised in (Cummins and Newman 2010).

The RatSLAM system (Milford and Wyeth 2003; Milford et al. 2004) is derived from models of neural mechanisms underlying spatial navigation in the rodent hippocampus; specifically, the use of a three-dimensional competitive attractor network to combine visual and odometric information over a sequence of observations to form a location hypothesis. RatSLAM has successfully mapped many large-scale indoor and outdoor locations, notably the mapping of a 66km suburban road network in (Milford and Wyeth 2008). An online indoor robot delivery experiment was conducted, consisting of over 1100 delivery trials over a period of 2 weeks in 2 different office environments (Milford and Wyeth 2009), to date the longest continuous online demonstration of a SLAM system on a mobile robot (A slightly longer experiment is presented in (Marder-Eppstein et al. 2010), but the map is generated as an initialisation stage and remains static throughout the long term operation). Combinations of FAB-MAP with RatSLAM demonstrated long-term outdoor mapping with no false positive loop closures across multiple times of day and several weeks (Maddern et al. 2009; Glover et al. 2010).

RatSLAM requires a large number of system parameters relating to the underlying neural mechanisms, many of which are unitless and have no physical interpretation, and thus cannot be measured from characteristics of the robot platform. The effect that varying system parameters has on mapping performance is poorly understood due to the complex and dynamic nature of the

continuous attractor network, and the tuning process is largely heuristic and must be performed for each new environment and platform (Milford et al. 2006).

A SLAM system with a similar approach to the novel system presented in this paper is described in (Koenig et al. 2008). A Rao-Blackwellised particle filter is used to weight topological map hypotheses based on local odometric similarities (generated using nearest-neighbour distances between the global map and a fixed-length local odometry history) and visual similarities (generated using HSV histogram distance for panoramic images). An adaptive sensor model is used to compare the current histogram to one generated at an arbitrary distance between two previously visited locations, similar to the continuous appearance model presented in Section 4. Although this system is built on probabilistic principles and has been demonstrated to map routes up to 1.5km in length, it too requires a number of tuning parameters to function on a particular platform in a given environment.

The promising results of the motion model in FAB-MAP at larger scales and the successes of RatSLAM indicate that for large-scale long-term operation, motion and sequence information can provide higher loop closure detection performance than appearance-only or appearance-and-geometry matching alone.

3. Background

The following section describes the essential components of two SLAM systems from which components of CAT-SLAM are derived: Particle Filter SLAM and FAB-MAP.

A. Particle Filter SLAM

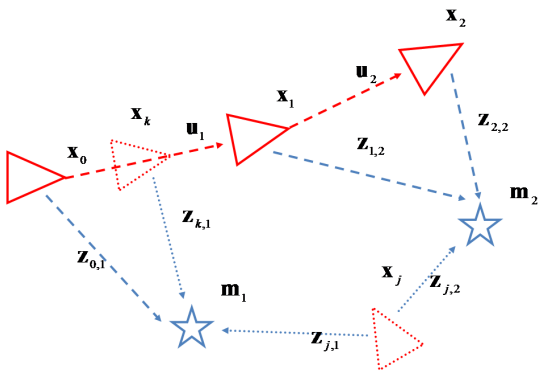


Figure 1 – Geometric SLAM interpretation. A continuous observation and motion model is defined by successive observations of feature geometry.

The majority of Kalman- and Rao-Blackwellised particle filter approaches to the SLAM problem use a geometric interpretation of the observation and motion model, shown in Figure 1. A series of metric measurements $\mathbf{z}_i$ are taken from locations $\mathbf{x}_i$ to features $\mathbf{m}_i$, typically in the form of range, bearing or a combination. The location of the features $\mathbf{m}_i$ with respect to the previously visited discrete locations $\mathbf{x}_i$ can then be determined in continuous geometric space. Additionally, the expected observation for locations between previously visited states (labelled $\mathbf{x}_k$) can be determined using relative geometry, as can the expected observation for any arbitrary location in space (labelled $\mathbf{x}_j$).

A popular SLAM algorithm that makes use of the geometric solution to the SLAM problem is FastSLAM, developed in (Montemerlo et al. 2002), which uses a Rao-Blackwellised particle filter and various schemes for particle resampling. By storing many different location and map hypotheses

as individual particles and assigning weights to those particles based on how well they match observations, particle filter SLAM avoids both the linearization and computational complexity issues of EKF SLAM. The chief innovation in Rao-Blackwellisation is decoupling the process noise from the observation noise. By assuming the map stored by each particle is correct, observations become conditionally independent. The distribution is partitioned as follows:

$$\begin{aligned} &P\left(\mathbf{x}_{0:k}, \mathbf{m} \mid \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}, \mathbf{x}_0\right) \\ &= P\left(\mathbf{m} \mid \mathbf{x}_{0:k}, \mathbf{Z}_{0:k}\right) P\left(\mathbf{x}_{0:k} \mid \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}, \mathbf{x}_0\right) \end{aligned} \tag{1}$$

The joint state is represented by N particles, each with pose history $\mathbf{X}$, weight w and distribution as follows:

$$\left\{w_k^{(i)}, \mathbf{X}_{0:k}^{(i)}, P\left(\mathbf{m} \mid \mathbf{X}_{0:k}^{(i)}, \mathbf{Z}_{0:k}\right)\right\}_i^N \tag{2}$$

The motion-update is performed by directly sampling from the distribution for each particle:

$$\mathbf{x}_k^{(i)} \sim P\left(\mathbf{x}_k \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) \tag{3}$$

Each particle is then assigned a weight based on the importance function:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{P\left(\mathbf{z}_k \mid \mathbf{X}_{0:k}^{(i)}, \mathbf{Z}_{0:k-1}\right) P\left(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right)}{\pi\left(\mathbf{x}_k^{(i)} \mid \mathbf{X}_{0:k-1}^{(i)}, \mathbf{Z}_{0:k}, \mathbf{u}_k\right)} \tag{4}$$

All weights are normalised to sum to unity. The particles are then resampled with replacement, where the probability of selection is proportional to the weight w. Remaining particles are then updated using the EKF or similar While this process is effective in allowing particle filter SLAM to store multiple hypotheses and switch between them as required, it can suffer from "particle deprivation" if there are no particles near the correct hypothesis (Van der Merwe et al. 2001).

Many extensions have been made to the FastSLAM algorithm: FastSLAM 2.0 (Montemerlo et al. 2003), which includes the current observation in the proposal distribution for locally optimal sampling; GridSLAM (Hähnel et al. 2003), which extends the environment representation to an occupancy grid reducing the complications of data association in feature-based representations; and Distributed Particle SLAM (DP-SLAM) (Eliazar and Parr 2003), which further reduces the computational complexity of FastSLAM by storing the particles in an ancestry tree and recording map divergences rather than storing an entire map for each particle. Rao-Blackwellised particle filters are also used in a number of vision-based SLAM systems, notably the monocular SLAM based approach in (Eade 2008).

B. Appearance-based Place Recognition

Appearance-based SLAM is primarily used for detecting loop closures in large unknown environments, which it performs by determining whether the current location matches any previously visited locations or is sufficiently different as to be classified as a new location.

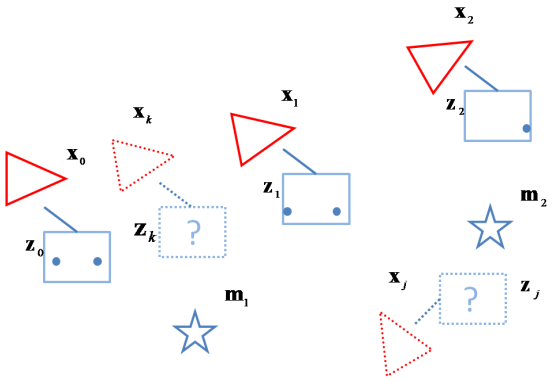


Figure 2 – Appearance-based SLAM interpretation. Expected observations are only available at discrete locations where an observation was previously made. Motion information is not used, allowing loop closures regardless of accumulated metric error.

Figure 2 illustrates the appearance-based approach to the SLAM observation and motion model. Each state $\mathbf{x}_i$ has an associated observation $\mathbf{z}_i$, which stores features $\mathbf{m}_i$ that are visible from that location. The map is represented by the history of states $\mathbf{X}_{0:k}$. However, motion information is typically discarded, since there is no method of generating the expected appearance neither between locations (labelled $\mathbf{x}_k$) nor at arbitrary locations (labelled $\mathbf{x}_j$). Appearance-based SLAM systems can therefore close loops of any size, regardless of accumulated odometry error, but rely entirely on the data association between the current observation and a previous observation.

The current state-of-the-art appearance-based SLAM system is FAB-MAP (Cummins and Newman 2008), which uses a Chow-Liu dependency tree and recursive Bayes estimation within a rigid probabilistic framework to provide robust loop closure detection.

Each image is converted into the visual bag-of-words representation described in (Sivic and Zisserman 2003). It is therefore necessary to create a database of common features from a set of training data in a similar environment to the test environment prior to performing localisation (Cummins and Newman 2007). Every feature extracted from the image is converted to the closest visual word, reducing each image to a binary vector of which words are present in the image.

$$Z_k = \{z_1, \ldots, z_{|v|}\} \tag{5}$$

Each unique location $L_k$is represented by the probability that the object e_i(that creates observation z_i) is present in the scene.

$$\{P(e_i = 1 \mid L_k), \ldots, P(e_{|v|} = 1 \mid L_k)\} \tag{6}$$

The probability of a new image coming from the same location as a previous image is estimated using recursive Bayes:

$$P(L_i \mid Z_{0:k}) = \frac{P(Z_k \mid L_i, Z_{0:k-1}) P(L_i \mid Z_{0:k-1})}{P(Z_k \mid Z_{0:k-1})} \tag{7}$$

where $Z_{0:k-1}$is a collection of previous observations up to time k. $P(Z_k \mid L_i, Z^{k-1})$ is assumed to be independent from all past observations and is calculated using a Chow Liu approximation (Chow and Liu 1968). The Chow Liu tree is constructed once as an offline process based on training data. Observation likelihoods are determined using the Chow Liu tree as follows:

$$P(Z_k \mid L_i) \approx P(z_r \mid L_i) \prod_{q=1}^{|v|} P(z_q \mid z_{p_q}, L_i) \tag{8}$$

where r is the root node of the Chow Liu tree and $p_q$is the parent of node q. The prior probability of matching a location $P(L_i \mid Z_{0:k-1})$ is estimated using a naïve motion model, where the probability of a new place $P(L_{new} \mid Z^{k-1})$ is set to a constant if the current hypothesised location is within 1 frame of the matched location. In practice this has only a slight effect on the final result for smaller datasets (Cummins and Newman 2008), but provides a marked improvement on recall in larger datasets (Cummins and Newman 2010).

The denominator of equation 7 incorporates the probability of matching to a new location in addition to localisation to a previously visited place. To estimate if a new observation comes from a previously unvisited location the model needs to consider all locations, not just visited locations. This can be split into mapped and unmapped locations:

$$\begin{aligned} P(Z_k \mid Z_{0:k-1}) = & \sum_{m \in M} P(Z_k \mid L_m) P(L_m \mid Z_{0:k-1}) \\ + & \sum_{n \in \overline{M}} P(Z_k \mid L_n) P(L_n \mid Z_{0:k-1}) \end{aligned} \tag{9}$$

where M is the set of mapped locations. Since the second term cannot be evaluated directly (as it would require information on all unknown locations), an estimation must be used. A random selection of scenes from training data is used to evaluate the unmapped location according to:

$$\begin{aligned} & \sum_{n \in \overline{M}} P(Z_k \mid L_n) P(L_n \mid Z_{0:k-1}) \\ & \approx P(L_{new} \mid Z_{0:k-1}) \sum_{u=1}^{n_s} \frac{P(Z_k \mid L_u)}{n_s} \end{aligned} \tag{10}$$

where L_u is a sampled location and n_s is the total number of samples. The sampling technique generally provides superior results to the mean field approximation (Cummins and Newman 2008).

To provide speed increases, the implementation in (Cummins and Newman 2008) presented a probabilistic bail-out condition based on the Bennett Inequality (Boucheron et al. 2004), to rank features based on their information content and to discard unlikely matches without performing the full recursive Bayes calculation. To further reduce the amount of computation required, an inverted index lookup scheme was implemented in FAB-MAP 2.0 (Cummins and Newman 2009), which allows fully sparse evaluation.

A number of attempts to incorporate FAB-MAP into a full mapping system have been made, where it has been used as a first stage to detect loop closure. However, these attempts then rely on geometric matching techniques using either laser scanners (Paul and Newman 2010) or stereo cameras (Newman et al. 2009; Sibley et al. 2010), which do not incorporate odometric information in the manner of a pose filter, and as such still rely entirely on the strength of data association between two discrete locations.

4. Continuous Appearance-based Trajectory SLAM

The proposed SLAM system outlined in this section is derived from a ‘trajectory-based’ interpretation of the SLAM problem. This interpretation lies between the two major SLAM paradigms presented in the previous section; it combines aspects of the geometric motion model and

the localisation distribution of particle filter SLAM with the appearance-based observation model and new place detection of FAB-MAP. A diagram of the trajectory-based interpretation is presented in Figure 3.

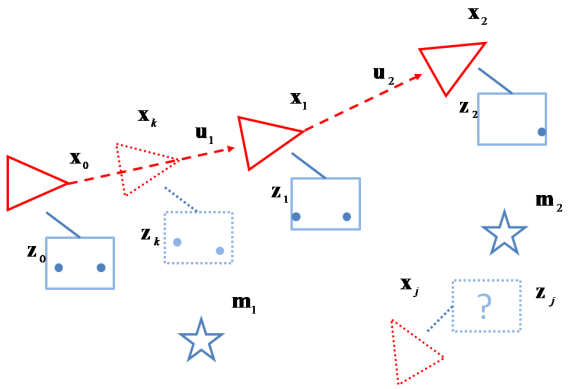


Figure 3 – Trajectory-based SLAM interpretation. A continuous trajectory-based observation model allows the expected appearance to be calculated at any point along a previously visited trajectory. Motion information permits the use of pose filtering without restricting loop closure size.

As with particle filter SLAM, states $\mathbf{x}_i$ are linked by odometry information $\mathbf{u}_i$; however, observations $\mathbf{z}_i$ are formed by appearance representations rather than metric distances. The observation model is formed by a continuous trajectory-based appearance model, which calculates the expected appearance along the trajectory between two nodes. This model allows the calculation of the expected observation $\mathbf{z}_k$ from location $\mathbf{x}_k$ on the trajectory between two previously visited locations. However, unlike the geometric observation model, it does not allow the calculation of the expected observation $\mathbf{z}_j$ at an arbitrary location $\mathbf{x}_j$. This limits the system to localising only to exact trajectories it has previously traversed; however, the utility of other appearance-based SLAM methods indicate that this capability is not required for all applications (Milford and Wyeth 2009).

We define a continuous trajectory T which intersects all previously visited locations $\mathbf{X}_{0:k}$:

$$\mathbf{X}_{0:k} \in T \tag{11}$$

The continuous trajectory T is *not* subject to global geometric correction when loop closures are detected – this is to ensure that multiple traversals of identical locations yield identical odometric sequences regardless of any systematic bias. The map $\mathbf{m}$ is formed by the history of poses as follows:

$$\mathbf{m} = \mathbf{X}_{0:k} \tag{12}$$

To perform localisation and mapping along the trajectory, we require a solution to the following location distribution:

$$P\left(\mathbf{x}_k \,|\, \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}\right) \tag{13}$$

This distribution can be divided into two components: one for all locations along the previously visited trajectory, and one for all previously unvisited locations (as in equation 9). This can be updated recursively as follows:

$$P\left(\mathbf{x}_k \,|\, \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}\right) = \frac{P\left(\mathbf{z}_k \,|\, \mathbf{x}_k\right) P\left(\mathbf{x}_k \,|\, \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k-1}\right)}{\sum_{m \in T} P\left(\mathbf{z}_k \,|\, \mathbf{x}_m\right) P\left(\mathbf{x}_m \,|\, \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k-1}\right) + \sum_{n \in \bar{T}} P\left(\mathbf{z}_k \,|\, \mathbf{x}_n\right) P\left(\mathbf{x}_n \,|\, \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k-1}\right)} \tag{14}$$

In this case the set *m* denotes all poses along the continuous trajectory *T*, and the set *n* denotes all poses not on the trajectory. Localisation is performed by evaluating the distributions for poses *m* along the trajectory detailed in the following section, and the likelihood of visiting a new place is approximated by a sum of distributions for poses *n* not on the trajectory, detailed in Section 4D.

A. Trajectory-based Pose Filtering

The history of discrete poses $\mathbf{X}_{0:k}$ is formed by applying the non-linear motion model naively for each motion update; essentially forming a map using uncorrected odometry information.

$$\hat{\mathbf{x}}_k^{(i)} = f\left(\mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) \tag{15}$$

Although this map will be globally inaccurate as no pose correction is applied at this stage, local sections that are revisited will exhibit similar local odometric sequences. For a 3 degree of freedom model the state and motion update are as follows:

$$\mathbf{x}_k = \begin{bmatrix} x_k & y_k & \theta_k \end{bmatrix}^T, \mathbf{u}_k = \begin{bmatrix} \Delta x_k & \Delta y_k & \Delta\theta_k \end{bmatrix}^T \tag{16}$$

$$f\left(\mathbf{x}_k, \mathbf{u}_k\right) = \begin{bmatrix} x_k + \Delta x_k \cos\left(\theta_k + \Delta\theta_k\right) - \Delta y_k \sin\left(\theta_k + \Delta\theta_k\right) \\ y_k + \Delta x_k \sin\left(\theta_k + \Delta\theta_k\right) + \Delta y_k \cos\left(\theta_k + \Delta\theta_k\right) \\ \theta_k + \Delta\theta_k \end{bmatrix} \tag{17}$$

To extend this representation from observations at discrete locations indexed by an integer *k* to a continuous sequence, we define a continuous pose representation $\mathbf{x}(t)$ with continuous index *t*, where $\mathbf{x}(k) = \mathbf{x}_k$:

$$\mathbf{x}(t) \in T, \quad 0 \le t \le k \tag{18}$$

For the 3 degree of freedom case $\mathbf{x}(t)$ is defined as follows:

$$\mathbf{x}(t) = \begin{bmatrix} x(t) & y(t) & \theta(t) \end{bmatrix}^T \tag{19}$$

The particular form of the trajectory *T* is defined by the continuous motion model of the vehicle. The simplest case of a piecewise-linear interpolated motion model is illustrated as follows:

$$\mathbf{x}(t) = \left(\lceil t \rceil - t\right)\mathbf{X}_{\lfloor t \rfloor} + \left(t - \lfloor t \rfloor\right)\mathbf{X}_{\lceil t \rceil} \tag{20}$$

Figure 4 illustrates the form of the trajectory *T* and the method of continuous indexing for a sample loop closure scenario.

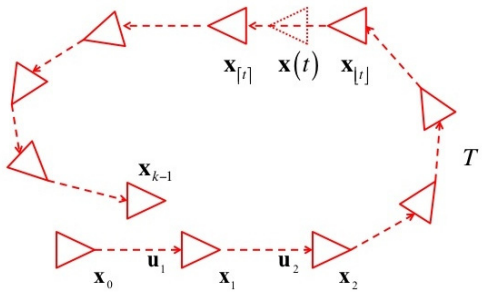


Figure 4 – Sample loop closure scenario for a trajectory-based SLAM system. Previously visited locations are denoted by $\mathbf{x}_k$, and odometry by $\mathbf{u}_k$. The continuous trajectory *T* is defined

by the piecewise linear path formed by uncorrected odometry that intersects all previously visited locations. This ensures that repeated paths through the same location have the same local metric sequence. The continuous index t allows the expected pose $\mathbf{x}(t)$ to be generated at any location on the trajectory by interpolating between two adjacent previously visited locations. For this scenario, we are assuming the robot is revisiting a location near to $\mathbf{x}_0 \ldots \mathbf{x}_1$ at time $\mathbf{x}_{k-1}$, but accumulated odometry error has caused these locations to be distant in global metric space.

Using the trajectory T, the full continuous history of poses can be indexed with the continuous index t, reducing the localisation distribution to a one-dimensional PDF. The distribution conditioned on the continuous trajectory T is as follows:

$$P\left(\mathbf{x}_k \in T \,|\, \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}\right) \tag{21}$$

The distribution above is evaluated using a particle filter using N particles, each with weight w, continuous trajectory index t and boolean trajectory direction d:

$$\left\{w_k^{(i)}, t^{(i)}, d^{(i)}\right\}_i^N \tag{22}$$

The direction component d allows the particles to propagate in either direction along the one-dimensional continuous trajectory. The following sections detail the components of the particle filter required to solve the joint distribution along the continuous trajectory.

B. Trajectory-based Sampling

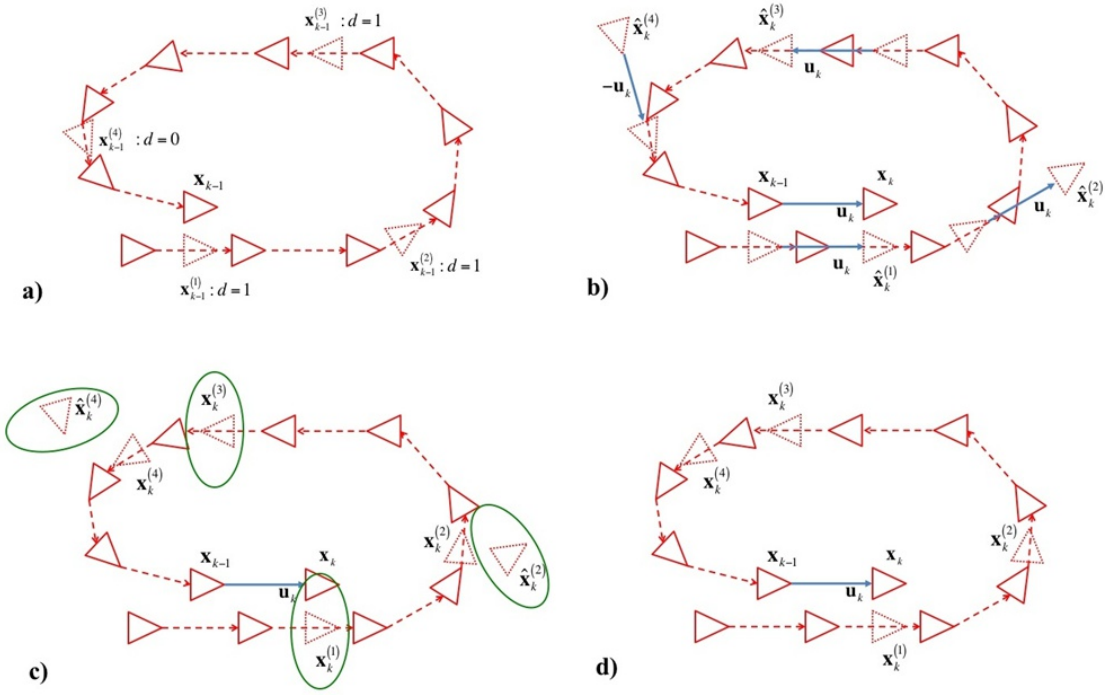


Figure 5 – Trajectory-based sampling for a loop closure scenario. a) illustrates the state of the trajectory at time k-1. Four particles $\mathbf{x}^{(1)} \ldots \mathbf{x}^{(4)}$ are distributed along the trajectory. Particle $\mathbf{x}^{(4)}$ has a reversed direction variable d. b) shows the trajectory update, creating the new location $\mathbf{x}_k$ using odometry $\mathbf{u}_k$. The location of each particle is also updated by $\mathbf{u}_k$ plus Gaussian error $\mathbf{w}_k$ in local metric space, causing most of the particles to leave the trajectory. Particle $\mathbf{x}^{(4)}$ is updated by $-\mathbf{u}_k$ to represent the possibility of traversing the trajectory in the reverse direction. In c) the maximum likelihood location along the trajectory is shown for each particle. Particles located

further from the trajectory, such as $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(4)}$, will have a reduced odometry likelihood term $P\left(\mathbf{x} \mid \hat{\mathbf{x}}_k\right)$. d) illustrates the final locations of each particle after the motion update.

The process of trajectory-based sampling is illustrated in Figure 5. The proposal distribution for the trajectory-based particle filter is given by the vehicle motion model conditioned on the trajectory *T*:

$$\mathbf{x}_k^{(i)} \sim P\left(\mathbf{x}_k \in T \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) \tag{23}$$

This method ensures all particles remain constrained to the trajectory of previously visited locations. The particle update is performed by first generating a proposed pose $\hat{\mathbf{x}}_k$ using the nonlinear vehicle model *f* given control input $\mathbf{u}_k$ with additive Gaussian noise $\mathbf{w}_k$ and direction *d*:

$$\hat{\mathbf{x}}_k^{(i)} = \begin{cases} f\left(\mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) + \mathbf{w}_k, & d = 1 \\ f\left(\mathbf{x}_{k-1}^{(i)}, -\mathbf{u}_k\right) + \mathbf{w}_k, & d = 0 \end{cases} \tag{24}$$

The binary direction variable *d* determines whether the particle is propagated in the forwards or reverse direction along the trajectory. This allows the vehicle to correctly relocalise in locations which it is traversing in the opposite direction to the original route. The proposed state covariance is generated by linearising the motion model at the proposed state location with noise covariance $\mathbf{Q}_k$:

$$\Sigma_k^{(i)} = J_k^{(i)} \mathbf{Q}_k J_k^{(i)\mathrm{T}}, \quad J_k^{(i)} = \frac{\delta f\left(\mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right)}{\delta \mathbf{u}} \tag{25}$$

From this, a distribution over all possible states can be represented using the standard multivariate Gaussian:

$$P\left(\mathbf{x} \mid \hat{\mathbf{x}}_k\right) = \frac{1}{2\pi\sqrt{|\Sigma_k|}} \exp\left[-\frac{1}{2}\left(\mathbf{x} - \hat{\mathbf{x}}_k\right)\Sigma_k^{-1}\left(\mathbf{x} - \hat{\mathbf{x}}_k\right)^{\mathrm{T}}\right] \tag{26}$$

Given the particle can only exist at a location on the continuous trajectory T, the most likely location of the particle is found by locally searching the trajectory for the location for which the above distribution is maximised:

$$t^{(i)} = \underset{t \le k}{\operatorname{argmax}}\, P\left(\mathbf{x}(t) \mid \hat{\mathbf{x}}_k^{(i)}\right) \tag{27}$$

From this index the pose of the particle is set to the maximum likelihood pose on the trajectory relative to the current pose:

$$\mathbf{x}_k^{(i)} = \mathbf{x}\left(t^{(i)}\right) \tag{28}$$

The value of the maximum motion likelihood $P\left(\mathbf{x} \mid \hat{\mathbf{x}}_k\right)$ is stored for use in particle importance weighting.

C. Continuous Appearance Representation

The observation model can take any form, but is only required to determine the existence or non-existence of visible features along the continuous trajectory between two sequential observations. As such, methods that do not require feature correspondence or geometry are preferred.

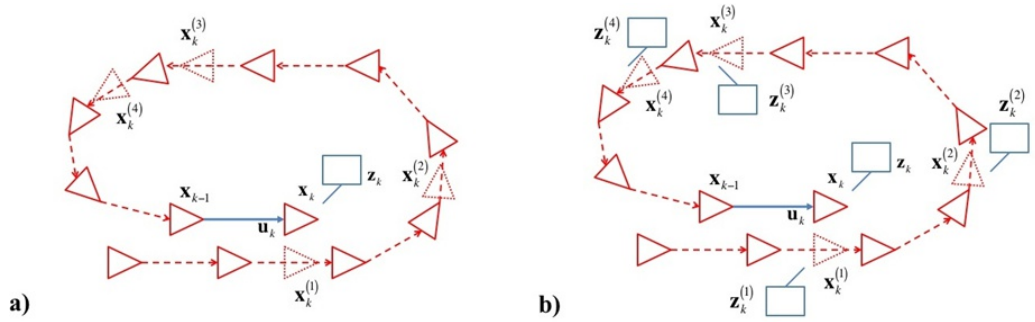


Figure 6 – Observation update for a loop closure scenario. a) shows the corresponding observation $\mathbf{z}_k$ for the new location $\mathbf{x}_k$. Using the continuous appearance representation, expected observations $\mathbf{z}_k^{(1)}$ … $\mathbf{z}_k^{(4)}$ can be generated for each particle by interpolating the observations from the previously visited locations adjacent to each particle, shown in b). Each of these expected observations can be compared to the current observation $\mathbf{z}_k$.

The observation update stage is illustrated in Figure 6. The location representation for each particle is derived from that presented in equation 6, but extended to represent appearance between discrete observations as follows:

$$\left\{P\left(e_i = 1 \mid \mathbf{x}\left(t^{(i)}\right)\right), \ldots, P\left(e_{|v|} = 1 \mid \mathbf{x}\left(t^{(i)}\right)\right)\right\} \tag{29}$$

The method of generating these interpolated appearance representations is dependent on both the continuous vehicle motion model and the camera model. Note that no information on the expected location of features in the image is required; only the likelihood that the feature is visible from the location is necessary. For the simple piecewise-linear case of equation 20, the continuous representation of appearance can be generated by linearly interpolating observation likelihoods between two successive discrete observations:

$$\begin{aligned} P\left(e_i = 1 \mid \mathbf{x}(t)\right) = \left(\lceil t \rceil - t\right) P\left(e_i = 1 \mid \mathbf{Z}_{\lfloor t \rfloor}\right) \\ + \left(t - \lfloor t \rfloor\right) P\left(e_i = 1 \mid \mathbf{Z}_{\lceil t \rceil}\right) \end{aligned} \tag{30}$$

The set of visual words v that form the observation and appearance representation must be derived from training data in a similar environment to the test environment along with the Chow-Liu dependency tree in the same manner as FAB-MAP. The use of the Chow-Liu tree in combination with the trajectory-based particle filter addresses perceptual aliasing in two ways: the Chow-Liu tree assigns likelihoods on a feature-by-feature basis using co-occurrence information from the training environment, and the particle filter assigns location hypotheses on a scene-by-scene basis based on previously visited locations in the test environment.

D. New Place Detection

To determine if the current set of observation and motion information indicates the vehicle is in a previously unvisited location, the following distribution must be evaluated:

$$P\left(\mathbf{x}_k \in \bar{T} \mid \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}\right) \tag{31}$$

As this distribution represents locations not previously visited, it cannot be evaluated directly. To represent the likelihood of a location not on the trajectory, we sample from an ‘unvisited’ location $\mathbf{x}^u$:

$$P\left(\mathbf{x}_k \in \bar{T} \mid \mathbf{z}_k, \mathbf{u}_k\right) = P\left(\mathbf{z}_k \mid \mathbf{x}_k^u\right) P\left(\mathbf{x}_k^u \mid \mathbf{u}_k\right) \tag{32}$$

The observation and motion distributions for an unvisited location can be approximated using information from training data as follows:

$$P\left(\mathbf{z}_k \mid \mathbf{x}_k^u\right) P\left(\mathbf{x}_k^u \mid \mathbf{u}_k\right) \approx P\left(\mathbf{z}_k \mid \mathbf{z}_{\text{avg}}\right) P\left(\mathbf{u}_{\text{avg}} \mid \mathbf{u}_k\right) \tag{33}$$

$\mathbf{z}_{\text{avg}}$ represents an 'average' observation and $\mathbf{u}_{\text{avg}}$ an 'average' control input. These are determined using the mean field approximation (Jordan et al., 1999), or the random sampling method used in (Cummins and Newman, 2008) and presented in equation 10. Without this 'unknown' pose likelihood, the particle distribution represents pure localization, since the probability of a pose not on the trajectory is assumed to be zero.

As localisation is performed by sampling directly from the trajectory distribution, and new place detection is performed by sampling from the training data, both can be combined in the particle filter update stage. The proposed weighting assigned to a location not on the trajectory is given as follows:

$$\hat{w}_k^u = \tfrac{1}{N} P\left(\mathbf{z}_k \mid \mathbf{z}_{\text{avg}}\right) P\left(\mathbf{u}_{\text{avg}} \mid \mathbf{u}_k\right) \tag{34}$$

The new location weight is denoted by w^u. Note that it is not recursively updated; this represents a uniform likelihood of departing the trajectory at any point in time.

E. Particle Weighting and Resampling

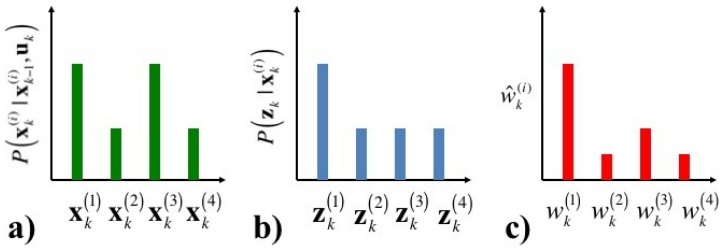


Figure 7 – Motion and observation likelihoods and subsequent particle weighting for a loop closure scenario. a) illustrates the motion likelihood terms generated for the scenario in Figure 5. Since the particles $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(3)}$ do not depart a significant distance from the local trajectory, they are weighted correspondingly higher than particles $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(4)}$. b) illustrates the observation likelihood terms for the scenario in Figure 6. Since particle $\mathbf{x}^{(1)}$ is close to the current location $\mathbf{x}_k$ the expected observation $\mathbf{z}_k^{(1)}$ has a greater matching likelihood to the current observation $\mathbf{z}_k$ than the expected observations at the other particle locations. c) illustrates the proposed weight of each particle incorporating both the motion and observation likelihoods.

The importance weighting of the particles is drawn from the numerators of equation 4 and 7; it combines the observation likelihood of FAB-MAP using the continuous representation of appearance with the motion prior of particle filter SLAM conditioned on the trajectory. By only evaluating the motion and observation model once per particle, updating the weights can be performed in constant time proportional to the number of particles regardless of the number of previously visited locations. The proposed weighting of each particle is as follows:

$$\hat{w}_k^{(i)} = w_{k-1}^{(i)} P\left(\mathbf{z}_k \mid \mathbf{x}_k^{(i)}\right) P\left(\mathbf{x}_k^{(i)} \in T \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) \tag{35}$$

The effect of each of these parameters on the proposed weights is illustrated in Figure 7. The observation likelihood makes use of the Chow Liu distribution as in equation 8 at location t on the trajectory:

$$P\left(\mathbf{z}_k \mid \mathbf{x}_k^{(i)}\right) = P\left(z_r \mid \mathbf{x}\left(t^{(i)}\right)\right) \prod_{q=1}^{|v|} P\left(z_q \mid z_{p_q}, \mathbf{x}\left(t^{(i)}\right)\right) \tag{36}$$

The rightmost part of equation 36 is calculated as follows:

$$\begin{aligned} & P\left(z_q \mid z_{p_q}, \mathbf{x}\left(t^{(i)}\right)\right) \\ & = \sum_{s \in \{0,1\}} P\left(z_q \mid e_q = s, z_{p_q}\right) P\left(e_i = s \mid \mathbf{x}\left(t^{(i)}\right)\right) \end{aligned} \tag{37}$$

where $P\left(z_q \mid e_q = s, z_{p_q}\right)$ is the detector probability and $P\left(e_i = s \mid \mathbf{x}\left(t^{(i)}\right)\right)$ is the continuous appearance representation defined in equation 30. The motion prior is the maximum likelihood point of the motion distribution along the trajectory as found in equation 27:

$$P\left(\mathbf{x}_k^{(i)} \in T \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) = P\left(\mathbf{x}_k^{(i)} \mid \hat{\mathbf{x}}_k^{(i)}\right) \tag{38}$$

The proposed weight of each particle is normalised, such that the sum of all weights of particles on the trajectory plus the new location weight is equal to 1.

$$w_k^{(i)} = \frac{\hat{w}_k^{(i)}}{\sum_j^N \hat{w}_k^{(j)} + \hat{w}_k^u} \tag{39}$$

The particles are resampled when the effective sample size (ESS) falls below a predefined threshold (Liu et al. 2001). This process is illustrated in Figure 8.

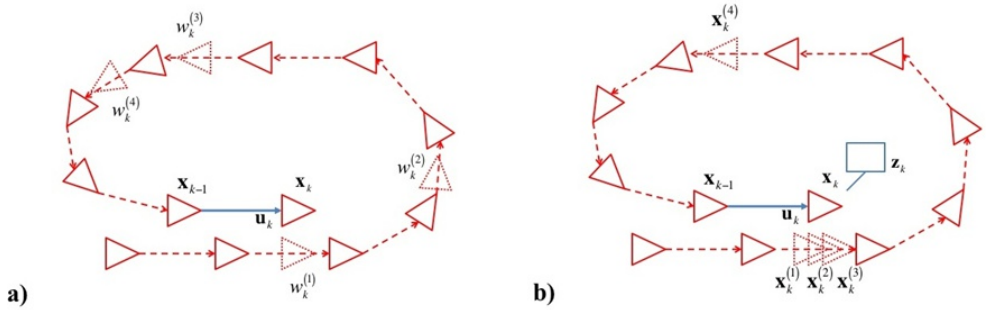


Figure 8 – Particle resampling for a loop closure scenario. Proposed particle weights from Figure 7 are normalized to generate final particle weights, shown in a). If the effective sample size (ESS) metric falls below a threshold, particles are resampled using Select with Replacement. Particles with lower weights are removed while particles with higher weights are duplicated. Not shown in this diagram is the new place weight w^u – particles selected to replace this weight are distributed to a uniform random location on the trajectory to counter particle deprivation in the event that no particles adequately represent the current location.

The ESS is computed as follows:

$$ESS = \frac{N}{1 + \frac{1}{N} \sum_j^{N,u} \left[N w_k^{(j)} - 1 \right]^2} \tag{40}$$

Particles are selected with probability proportional to their weight w_k using the Select with Replacement method (Liu et al. 2001). Any particles selected to replace the new location weight are sampled to a uniform random location on the trajectory (with a random direction) as follows:

$$t^{(i)} \sim U(0,k),\ \mathbf{x}_k^{(i)} = \mathbf{x}\left(t^{(i)}\right),\ d \sim \text{round}\left(U(0,1)\right) \tag{41}$$

This serves to counteract the effects of particle deprivation, since the proportion of particles sampled to diverse locations on the trajectory increases as the new place likelihood increases, thereby increasing the probability of detecting loop closures without requiring evaluation of every previously visited location.

F. Loop Closure Detection

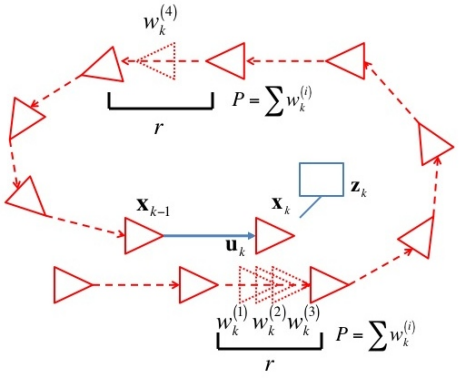


Figure 9 – Location hypothesis generation for a loop closure scenario. The probability of the location distribution at any point along the trajectory is calculated by the sum of particle weights within a fixed distance r. If the distribution probability exceeds a threshold P_{th} a loop closure is detected.

To determine the most likely location hypothesis from the distribution of particles a spatially selective method is used, equivalent to integrating the probability distribution over a short distance along the trajectory, illustrated in Figure 9. The value of the distribution at particle location $\mathbf{x}_k$ is as follows:

$$P\left(\mathbf{x}_k^{(i)}\right) = \sum_j^N h(i,j) \tag{42}$$

The spatially selective function $h(i,j)$ is defined as follows:

$$h(i,j) = \begin{cases} w_k^{(j)} & \left|\mathbf{x}_k^{(j)} - \mathbf{x}_k^{(i)}\right| \le r \\ 0 & \text{otherwise} \end{cases} \tag{43}$$

The distribution will only reach a probability of 1 at a location if all particles are within predefined distance r of that location. The value of r is selected based on the desired resolution of loop closure detection, and as such the location hypothesis is not subject to arbitrary location discretisation due to local visual saliency, a point discussed in (Mei et al. 2010).

At this point the CAT-SLAM algorithm can be used for loop closure detection by specifying a threshold P_{th}. If the maximally likely particle $P^{(i)}$ exceeds the threshold P_{th}, a loop closure exists between the current location and the particle location $\mathbf{x}^{(i)}$.

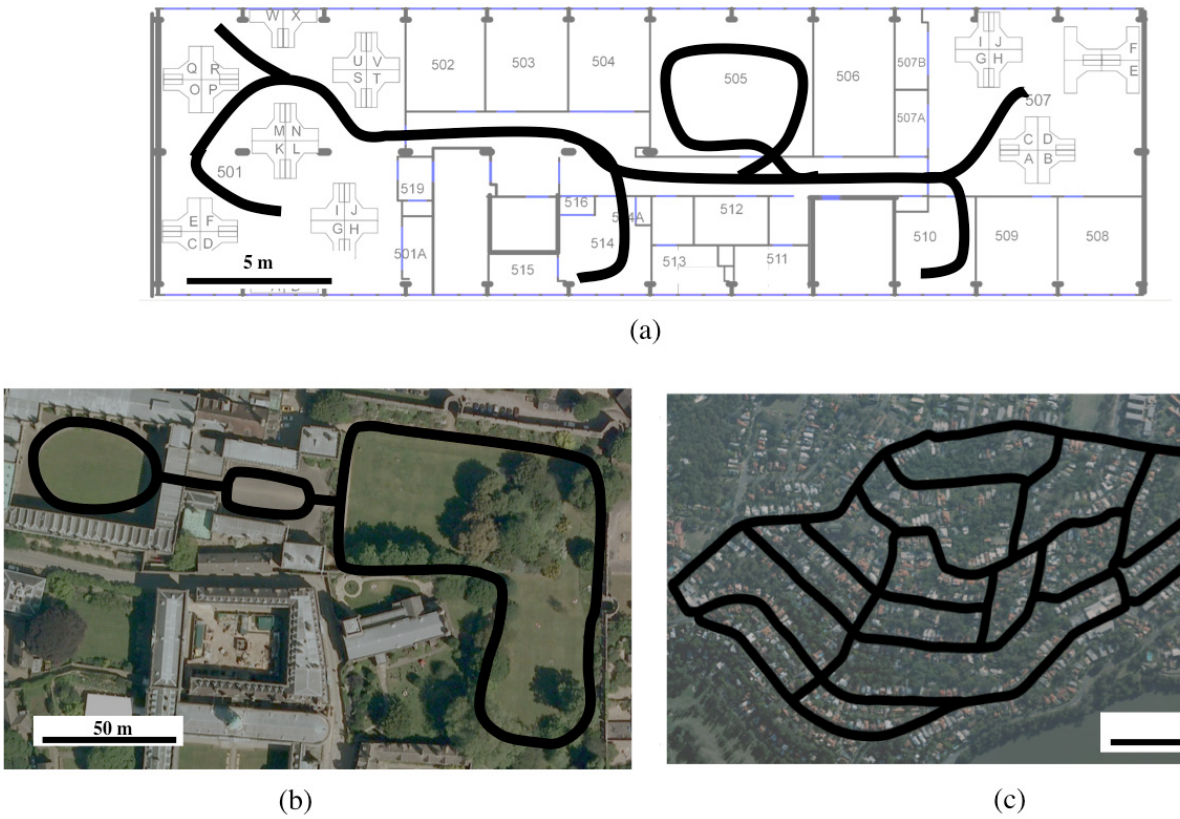
5. Experimental Procedure

This section details the steps taken to evaluate the CAT-SLAM algorithm. The experiments compare CAT-SLAM to FAB-MAP in a large outdoor environment, an urban environment and an indoor environment. These datasets were chosen to be representative of possible environments for autonomous robot navigation. While features such as corridors, pathways and road lanes ensure the

platforms traverse similar trajectories when revisiting locations, no particular effort was made to ensure the trajectories were perfectly overlapping in a metric sense.

The variation in scale, appearance and platform characteristics between the datasets demonstrate the loop closure performance of CAT-SLAM in a wide range of environments with minimal tuning requirements.

A. Experimental Setup



(a)

(b)

(c)

Figure 10 – (a) Indoor test environment consisting of a 500m traversal of an office building. (b) Urban test environment consisting of a 2.5km traversal of a university campus. (c) Outdoor test environment consisting of a 15km traversal of a suburban road network. Note the difference in scale of each environment. Satellite images © Google Maps 2011.

The indoor dataset is a 500m section of the robot delivery trial presented in (Milford and Wyeth 2009), pictured in Figure 10 (a). It consists of 15000 frames captured at 7fps from a Basler firewire camera and panoramic mirror combination aboard a Pioneer 3DX robot. The route taken is a continuous tour of the floor of a university building, with over 10 traversals of the main corridor and multiple visits to each room. Odometry information is provided by the shaft encoders on the wheels of the robot. Since GPS ground truth is not available indoors, a hand-corrected trajectory based on vehicle odometry and manually-tagged loop closures was used for the purposes of evaluation.

The urban dataset used for this evaluation is presented in (Smith et al. 2009). It comprises over 7000 panoramic images from a Point Grey Ladybug2 camera with accompanying wheel odometry (from shaft encoders on the Segway RMP) and GPS data logged at 5Hz. The route taken is a 2.5km tour of the grounds of New College, Oxford, pictured in Figure 10 (b), with multiple traversals of each location in both forward and reverse directions (a total of 5 traversals of the quadrangle area). Ground truth is provided by GPS locations; however, the signal is degraded in many locations throughout the urban dataset (particularly through a tunnel between courtyards). Approximately 45%

of the panoramic images have an associated valid GPS position; recall data for the precision recall curves is based on a hand-corrected trajectory, which provides a valid location for every frame. The results for this dataset differ slightly from those presented in (Maddern et al. 2011) due to the use of the corrected ground truth.

The outdoor experiment uses a dataset previously gathered for the full-day mapping experiment presented in (Glover et al. 2010). The dataset consists of 19000 video frames captured at 15 frames per second from a forward-facing Logitech QuickCam Pro 9000 webcam mounted on the roof of a car, as well as GPS data gathered at 1 Hz for ground truth. The GPS lock remained consistent throughout the entire route. The route taken by the car is a 15km tour of a surburban road network in St Lucia, Queensland, pictured in Figure 10 (c) with multiple repeated loops (up to 6 traversals of the same location) and wide variation in the types of roads traversed; from wide 4-lane main roads to single lane roads bordered by dense foliage. The dataset was gathered at midday to reduce the likelihood of image saturation due to direct sunlight. Because a forward-facing camera was used, repeated paths were always traversed in the same direction. The results for this dataset differ from those presented in (Maddern et al. 2010); for this evaluation the full 15km trajectory was used rather than a shorter section.

Table 1 presents a summary of the key properties of the three datasets.

Table 1 – Key dataset properties

Parameter	Indoor	Urban	Outdoor
Platform	Pioneer 3DX	Segway RMP	Automobile
Distance travelled	500m	2.5km	15km
Vision Hardware	Basler A310fc	Point Grey Ladybug2	Quickcam Pro 9000
Image Resolution	480x80 (panoramic)	2048x536 (panoramic)	640x480 (forward facing)
Avg. features per frame	70	1100	650

B. Algorithm Details

As both FAB-MAP and CAT-SLAM only require appearance information, no camera calibration or image registration is required. This reduces the setup in comparison to appearance-and-geometry methods; no special setup beyond codebook and Chow Liu tree generation was required for each dataset despite the considerable differences in camera model and image size. Feature descriptors are extracted using SURF (Bay et al. 2006), and a fast approximate nearest neighbour algorithm (Arya et al. 1998) was used to find the corresponding visual word for each descriptor.

The FAB-MAP implementation used for comparison is developed in-house and derived from (Cummins and Newman 2008). Enhancements presented in (Cummins and Newman 2009) primarily reduce computation time and increase scalability, and are not required for the comparatively small datasets used for this experiment. The geometric post-verification presented in (Cummins and Newman 2009) is not used for either algorithm. Parameters for the detector functions of FAB-MAP were taken from (Cummins and Newman 2008).

Training data for the codebook and Chow Liu tree for the urban and indoor dataset were provided by a downsampled 1000 image version of each main dataset respectively with repeated sections removed (as different datasets from similar environments were not available). The codebook was generated using modified sequential clustering (Teynor and Burkhardt 2007) yielding 6856 visual words for the urban dataset and 5320 words for the indoor dataset.

The codebook and Chow Liu tree used for the outdoor dataset are derived from the large-scale St Lucia full suburb experiments presented in (Milford and Wyeth 2008). SURF features were extracted from 7000 non-overlapping images sampled from a larger dataset of the same suburb, resulting in a codebook containing 5730 visual words. The average observation for each dataset was derived from training data using the mean field approximation in (Cummins and Newman 2008). The mean of the odometry in each axis from training data was used as the average motion vector.

Odometry information for the outdoor dataset was generated from visual information using the method presented in (Milford and Wyeth 2008). While this semi-metric method is not as precise as feature-based approaches, it does not require the calculation of feature correspondence and produces repeatable odometry in successive traversals of the same location. Odometry information for the urban and indoor dataset was provided by wheel odometry. The translational and rotational uncertainties were estimated based on typical performance for each platform.

The list of constants used in both algorithms for each dataset is presented in Table 2.

Table 2 – Dataset parameters for FAB-MAP and CAT-SLAM

FAB-MAP	**Indoor**	**Urban**	**Outdoor**
$p(z_i = 1 \mid e_i = 0)$	0	0	0
$p(z_i = 0 \mid e_i = 1)$	0.61	0.61	0.61
$p(L_{new} \mid \mathrm{Z}^{k-1})$	0.9	0.9	0.9
CAT-SLAM	**Indoor**	**Urban**	**Outdoor**
$p(z_i = 1 \mid e_i = 0)$	0	0	0
$p(z_i = 0 \mid e_i = 1)$	0.61	0.61	0.61
Translation Uncertainty σ_y	0.01	0.05	0.05
Rotation Uncertainty σ_θ	0.01	0.05	0.05
Number of Particles N	1000	1000	1000
ESS Threshold	0.3	0.3	0.3
Distribution Radius r	0.5 metres	2.5 metres	5 metres

To highlight the perfomance of CAT-SLAM due to the combination of odometry and appearance-based match information, we present additional results for two modified versions. The first version weights particles based on odometry match alone; the appearance-based observation likelihood is removed from equation 35 to produce the following weighting scheme:

$$\hat{w}_k^{(i)} = w_{k-1}^{(i)} P\left(\mathbf{x}_k^{(i)} \in T \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{u}_k\right) \tag{44}$$

We also present results for the opposite weighting scenario; particles are weighted using only appearance-based match likelihoods.

$$\hat{w}_k^{(i)} = w_{k-1}^{(i)} P\left(\mathbf{z}_k \mid \mathbf{x}_k^{(i)}\right) \tag{45}$$

Note that although this version of CAT-SLAM uses only visual match information to form the pdf (and therefore appears superficially similar to FAB-MAP), the positions of the particles along the trajectory are still updated using odometry information.

Because CAT-SLAM consists of a particle filter with a random sampling component, it will produce slightly different results in successive experiments with the same parameters. The CAT-SLAM results presented for the precision-recall and loop closure distribution is the median result obtained using 25 trials with the parameters listed in Table 2. The results presented for the parameter variation analysis were obtained with 25 successive trials for each combination of parameters.

6. Results

This section describes the mapping results of both FAB-MAP and the three CAT-SLAM variants on the three datasets. The primary performance metric is the precision-recall curve, where precision is defined as the number of correct matches divided by the total number of matches, and recall as the number of correct matches divided by the total number of expected matches. Both correct and expected correct matches are defined as previously visited locations within a set distance, equal to 2.5 metres for the indoor dataset, 7.5 metres for the urban dataset and 15 metres for the outdoor dataset.

Precision-recall curves have become the dominant metric for loop closure detection systems (Cummins and Newman 2009; Kawewong et al. 2010, Mei et al. 2010), and allow direct comparison between algorithms for a given dataset. However, depending upon the application, lower recall may be tolerated in favour of a more topologically distributed set of loop closures. For this reason we also present qualitative results illustrating the distribution of loop closures in Section 6B.

A. Precision-Recall

To use FAB-MAP or CAT-SLAM to detect loop closure for a semi-metric or metric SLAM system a precision of 100% is typically required, since false positive matches can cause catastrophic failure during mapping for geometric systems (Thrun and Leonard 2008). In this respect, the desired outcome is a high recall rate at 100% precision. However, many metric SLAM systems only require candidate loop closures, which are then subject to geometric verification. For these systems a small number of false positives are tolerated, and therefore we show results for both 100% and 99% precision. Additionally, analysis of the false positives reported by both systems is important to determine the likely failure modes in other environments.

Figure 11 shows the precision-recall curve for both FAB-MAP and CAT-SLAM for each dataset. It is apparent that for all three datasets the combined CAT-SLAM variant provides the best overall recall performance at high precision.

Predictably, the CAT-SLAM variant with odometry-only weighting performs poorly on all three datasets, as no absolute observations of location from vision are available. However, it does seem to indicate that environments with more complex trajectories (such as the indoor dataset) provide more information to disambiguate location than environments with simple trajectories (such as the outdoor dataset, which consists primarily of straight sections and left turns).

The CAT-SLAM variant with vision only weighting provides comparable performance to FAB-MAP for all three datasets. Although it does not directly use odometry information in the particle weighting process, it benefits from the trajectory-following properties of the particles, which allows it to track an appearance-based match across a length of the trajectory. This variant produces the highest precision results at 100% recall across all the datasets, which may be useful for proposing frequent loop closure candidates to a metric SLAM system. However, it is clear that the combination of odometry and appearance-based match information in the combined CAT-SLAM system provides superior results at 100% precision across all the test environments.

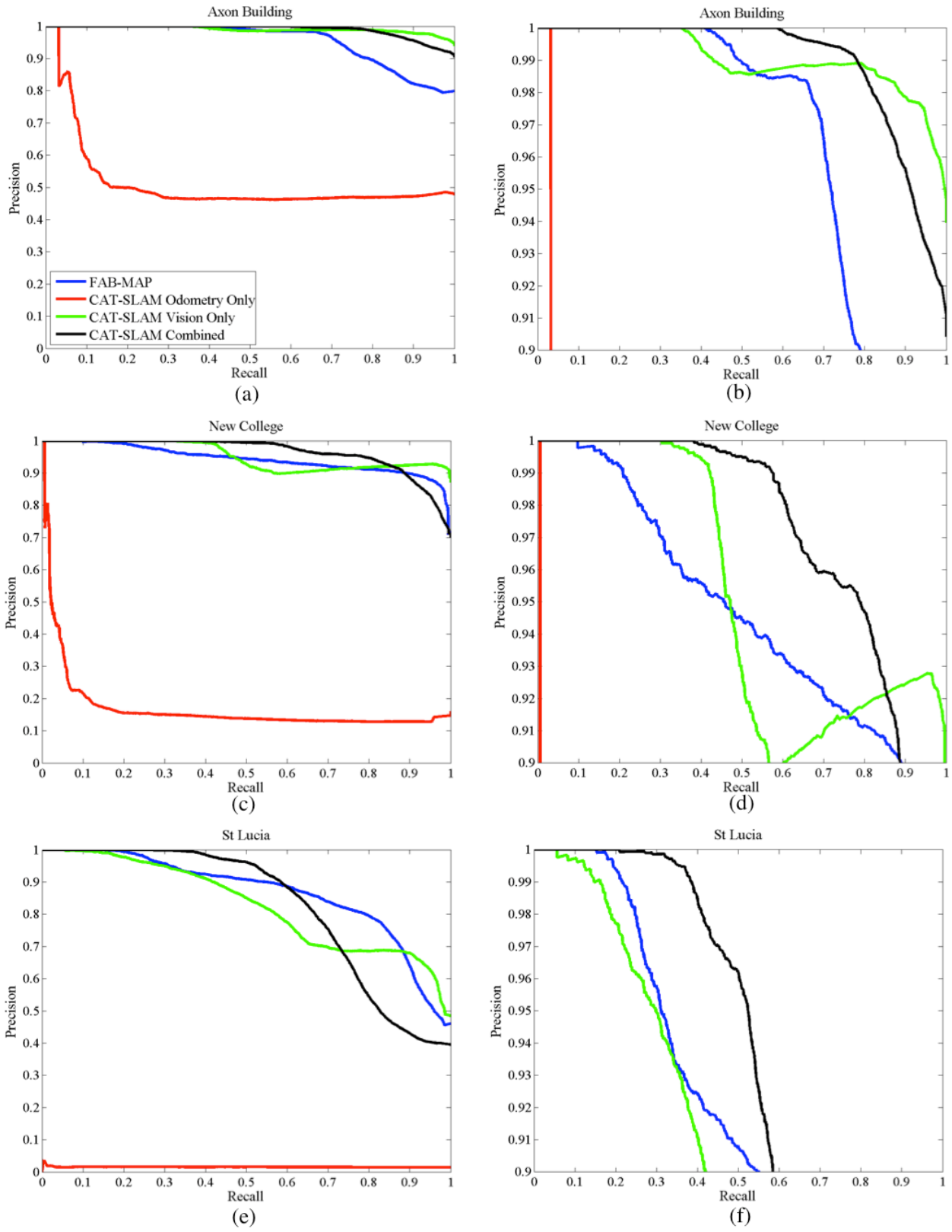


Figure 11 – Precision-recall curves for (a-b) the indoor dataset, (c-d) the urban dataset and (e-f) the outdoor dataset. (b), (d) and (f) illustrate the recall at precision values from 0.9 to 1. CAT-SLAM provides consistently higher recall than FAB-MAP at 100% precision. However, for lower precisions FAB-MAP outperforms CAT-SLAM on both the urban and the outdoor dataset.

Table 3 lists the recall values for 99% and 100% precision on each dataset for both algorithms.

Table 3 – Recall rates for FAB-MAP and CAT-SLAM variants at 99% and 100% precision

	Recall					
Dataset	**Indoor**		**Urban**		**Outdoor**	
Precision	**100%**	**99%**	**100%**	**99%**	**100%**	**99%**
FAB-MAP	41.05%	49.31%	9.78%	20.84%	15.32%	21.58%
CAT-SLAM Odometry Only	3.12%	3.14%	0.56%	0.56%	0.0%	0.0%
CAT-SLAM Vision Only	35.16%	42.93%	30.25%	42.17%	5.63%	16.0%
CAT-SLAM Combined	58.85%	78.13%	38.24%	57.57%	20.91%	38.48%

The indoor dataset is the least challenging of the datasets as it covers the shortest distance and has controlled lighting conditions, and hence both CAT-SLAM and FAB-MAP exhibit high recall rates at high precision. The trajectory-matching characteristics of CAT-SLAM provide a higher recall rate at 100% and 99% precision in comparison to FAB-MAP. The urban dataset is more challenging than the indoor dataset, as it traverses a number of indoor and outdoor locations with dynamic features and significant variation in lighting conditions. The results for FAB-MAP on the urban dataset are consistent with those presented in (Newman et al. 2006), providing approximately 10% recall at 100% precision and 20% recall at 99% precision. For this dataset, CAT-SLAM provides almost four times the recall at 100% precision, and even out-performs FAB-MAP at 90% precision.

Because of the illumination variation in the urban dataset, FAB-MAP often reports partial matches to previously visited locations without a sufficient probability to report a loop closure. Since CAT-SLAM builds up a location hypothesis over a number of observations rather than a single isolated frame, a location hypothesis can be formed from a sequence of partial visual matches in the correct order with the correct local metric sequence. This allows CAT-SLAM to maintain loop closure hypotheses along sections of the trajectory where strong visual information is not necessarily available from each individual frame. For the urban dataset, CAT-SLAM provides higher performance than FAB-MAP at both 100% and 99% precision.

The outdoor dataset provides a similar challenge to the urban dataset; large variations in lighting conditions and a number of dynamic objects (in this case, other vehicles on the road). Additionally, the field of view of the camera is greatly reduced in comparison to the previous two datasets, and the scale is significantly larger. FAB-MAP reports similar recall rates to the urban dataset at 100% precision with slightly over 10% recall. CAT-SLAM provides slightly higher recall at 100% precision, and almost double the recall of FAB-MAP at 100% precision.

B. Loop Closure Distribution

The following section presents an analysis of the loop closure distribution provided by both algorithms and the implications for using either algorithm for map construction in each environment.

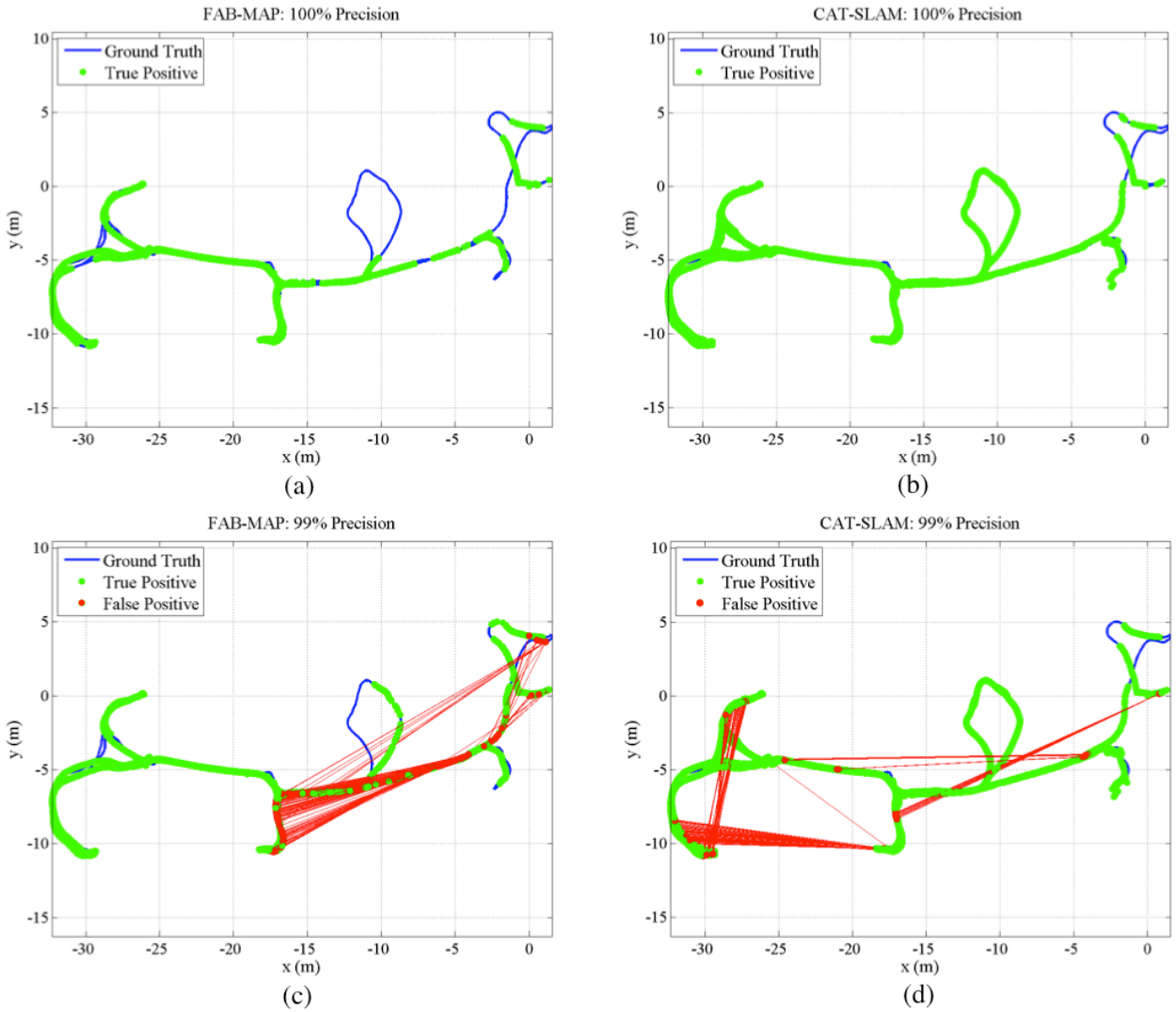


Figure 12 – Loop closure distribution for the indoor Axon Building dataset at 100% and 99% precision for FAB-MAP and CAT-SLAM. (a) and (b) show the distribution of true positives at 100% precision. (c) and (d) show both true positives and false positives at 99% precision.

Figure 12 shows loop closures detected by both FAB-MAP and CAT-SLAM on the indoor Axon Building dataset projected over ground truth locations. The high recall rate of CAT-SLAM at high precision is evident in the uniform distribution of correct loop closures in (b). FAB-MAP provides a similarly even distribution, however it fails to localise in a few visually indistinct areas such as the room at (-10, 0).

The distribution of true and false positives for each algorithm is presented in Figure 12 (c) and (d). For topological mapping and other applications requiring 100% precisions each false positive causes map corruption, however if the false positive links to a nearby location the corruption may not be catastrophic. The false positives reported by FAB-MAP in Figure 12 (c) appear to mostly originate from erroneous matches to locations around (0, 0) and (0, 5); false positives for CAT-SLAM in (d) are more evenly distributed throughout the environment.

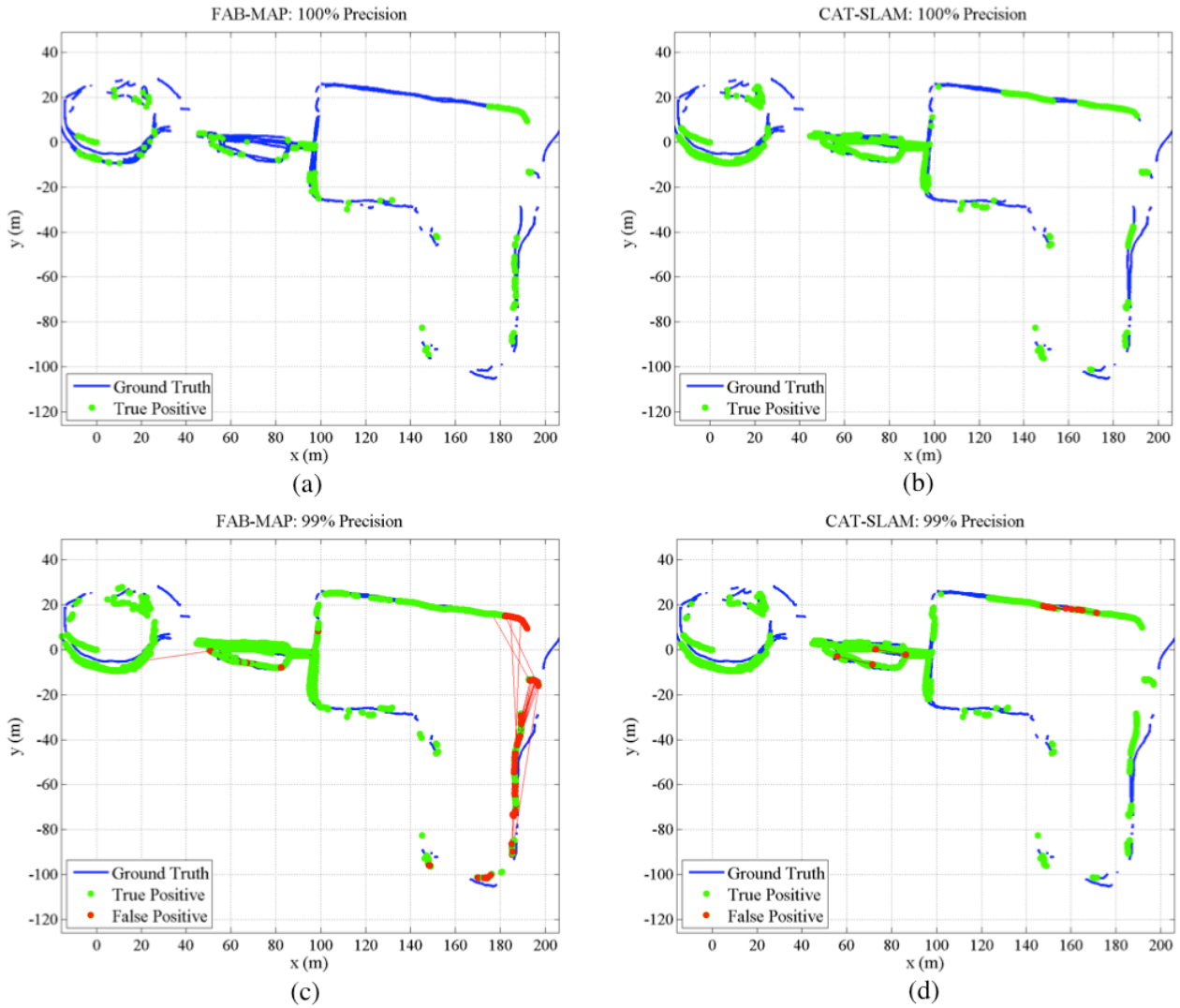


Figure 13 – Loop closure distribution for the urban New College dataset at 100% and 99% precision for FAB-MAP and CAT-SLAM. (a) and (b) show the distribution of true positives at 100% precision. (c) and (d) show both true positives and false positives at 99% precision.

Figure 13 shows loop closures detected by both FAB-MAP and CAT-SLAM on the New College dataset projected over the GPS ground truth (at locations where GPS signals were valid). At 100% precision in (a), FAB-MAP recalls only a small fraction of possible loop closures; large visually indistinct areas around (120, 20) are not recognized even when revisited twice. Inconsistent matching in distinct visual areas is also apparent around (70, 0). The advantages of performing trajectory-based matching in CAT-SLAM are particularly evident in (b). Parts of the trajectory that are not visually distinct in isolation are correctly localised given a sufficient number of partial matches in the correct order over a sequence of observations. In areas around (70, 0) almost every location is correctly matched to a previously visited location in the correct order.

Figure 13 (c) and (d) show locations of false positives for both algorithms at 99% precision. Both FAB-MAP and CAT-SLAM exhibit a number of false positives, however the majority of false positives reported by CAT-SLAM appear to be across short distances for this particular dataset.

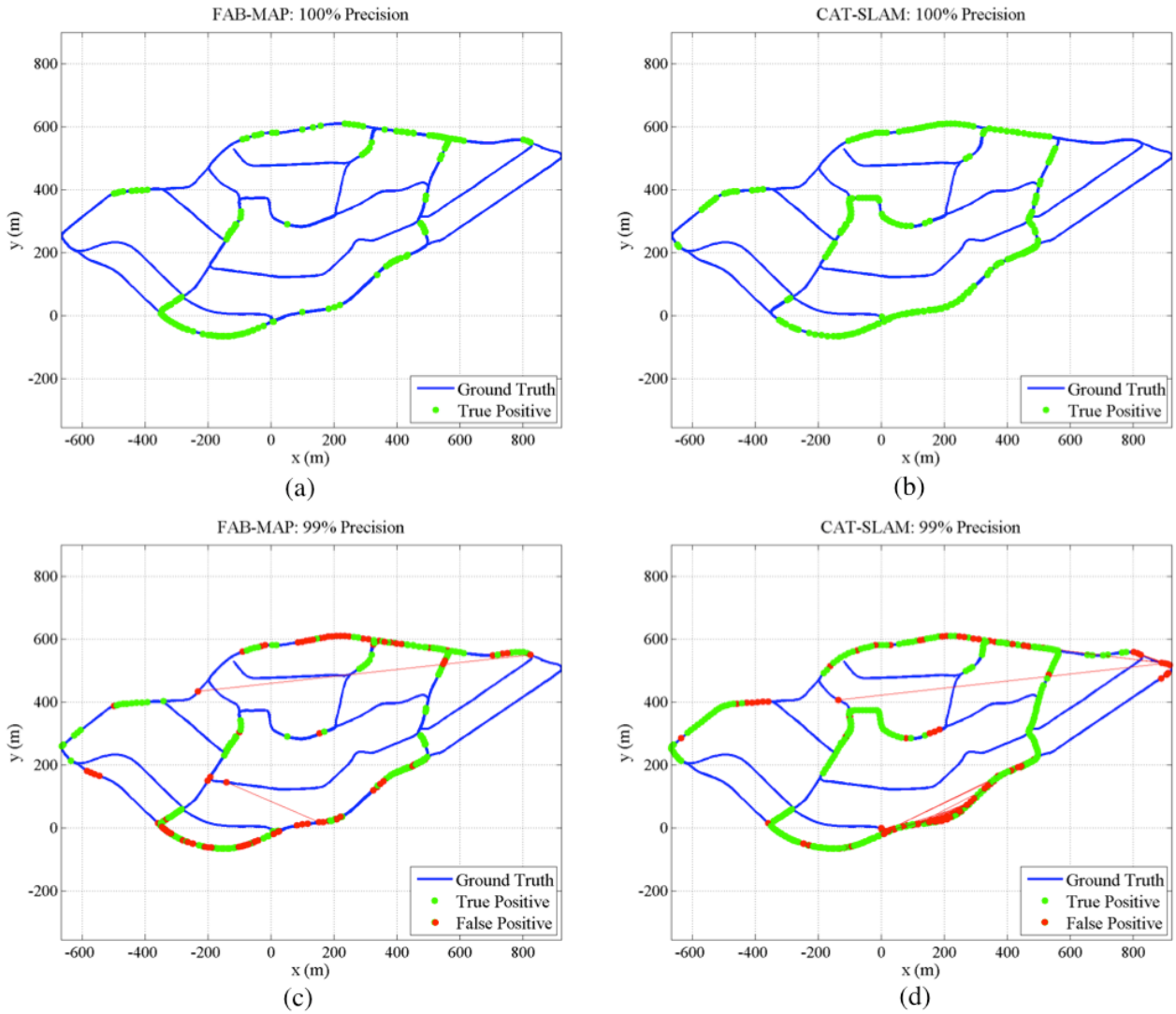


Figure 14 – Loop closure distribution for the outdoor St Lucia dataset at 100% and 99% precision for FAB-MAP and CAT-SLAM. (a) and (b) show the distribution of true positives at 100% precision. (c) and (d) show both true positives and false positives at 99% precision (Note that many sections of the trajectory are only visited once).

Figure 14 shows loop closures detected by both FAB-MAP and CAT-SLAM on the St Lucia dataset projected over the GPS ground truth. The relative distributions of true positive loop closures are of note: FAB-MAP tends to detect intermittent true positives within the space of a few metres, where CAT-SLAM typically has unbroken sequences of 20 metres or more. This reflects the sequential matching nature of CAT-SLAM; it requires a number of correct visual and odometric matches before a sufficiently dominant hypothesis is formed. However, once such a hypothesis is dominant, it is maintained until particles not within the local distribution distance d are sufficiently supported by novel visual and odometric information to suggest an alternate hypothesis. By requiring a sequence of supporting visual information over a number of updates, CAT-SLAM trades isolated loop closure detection for increased false positive rejection.

C. Parameter Variation

The following section presents results obtained over a number of trials for each dataset using different combinations of parameters. Figure 15 details the effect on recall at 99% and 100% precision for a variation in the number of particles, over a series of 25 trials. The ESS threshold is set

to 0.3 for consistency across the results. Across all datasets there is a clear correlation between increasingly reliable high recall rates with increasing number of particles, though the absolute recall rate does not benefit beyond a certain point for each dataset. It is important to note that the recall rate decreases for lower numbers of particles because the absolute number of true positives is lower, not because the absolute number of false positives is higher.

(a) and (b) illustrate the distribution of recall rates for the indoor dataset at 100% and 99% precision respectively. For reliably good performance exceeding that of FAB-MAP at 100% precision at least 1000 particles are required. While the distribution above 1000 particles is narrow, indicating the results are even more reliable, the absolute recall rates do not significantly exceed those provided by experiments with 1000 or 2000 particles. The trend is similar for 99% precision, except fewer particles are required to provide superior results to FAB-MAP.

The results for the urban dataset exhibit a similar trend to that of the indoor dataset, although in this case fewer than 100 particles are required for reliable performance exceeding FAB-MAP. In both the 100% and 99% precision cases, the recall benefits obtained for particle numbers in excess of 100 are minimal.

The challenging nature of the large scale of the outdoor dataset is apparent in (d) and (e). Experiments with fewer than 1000 particles fail to provide recall above 10%, and it appears that 1000 particles is still not sufficient to guarantee reliable recall rates. Greater numbers of particles may yet provide better results for this dataset; however, in the case of 99% precision the recall results for 1000 or more particles reliably exceed those of FAB-MAP.

A further experiment to determine recall rates when varying the ESS threshold for a fixed number of particles was performed. 10 trials were performed for each dataset for 1000 particles and ESS values between 0.05 and 0.5. The results did not show any consistent trend between ESS threshold and recall at either 99% or 100% precision for any of the datasets. As the ESS threshold has only a minor impact on computational requirements (a higher threshold will cause more frequent sampling, which adds a minor computational cost to each iteration), increasing or decreasing the ESS threshold provides no significant benefit to the overall performance of the algorithm.

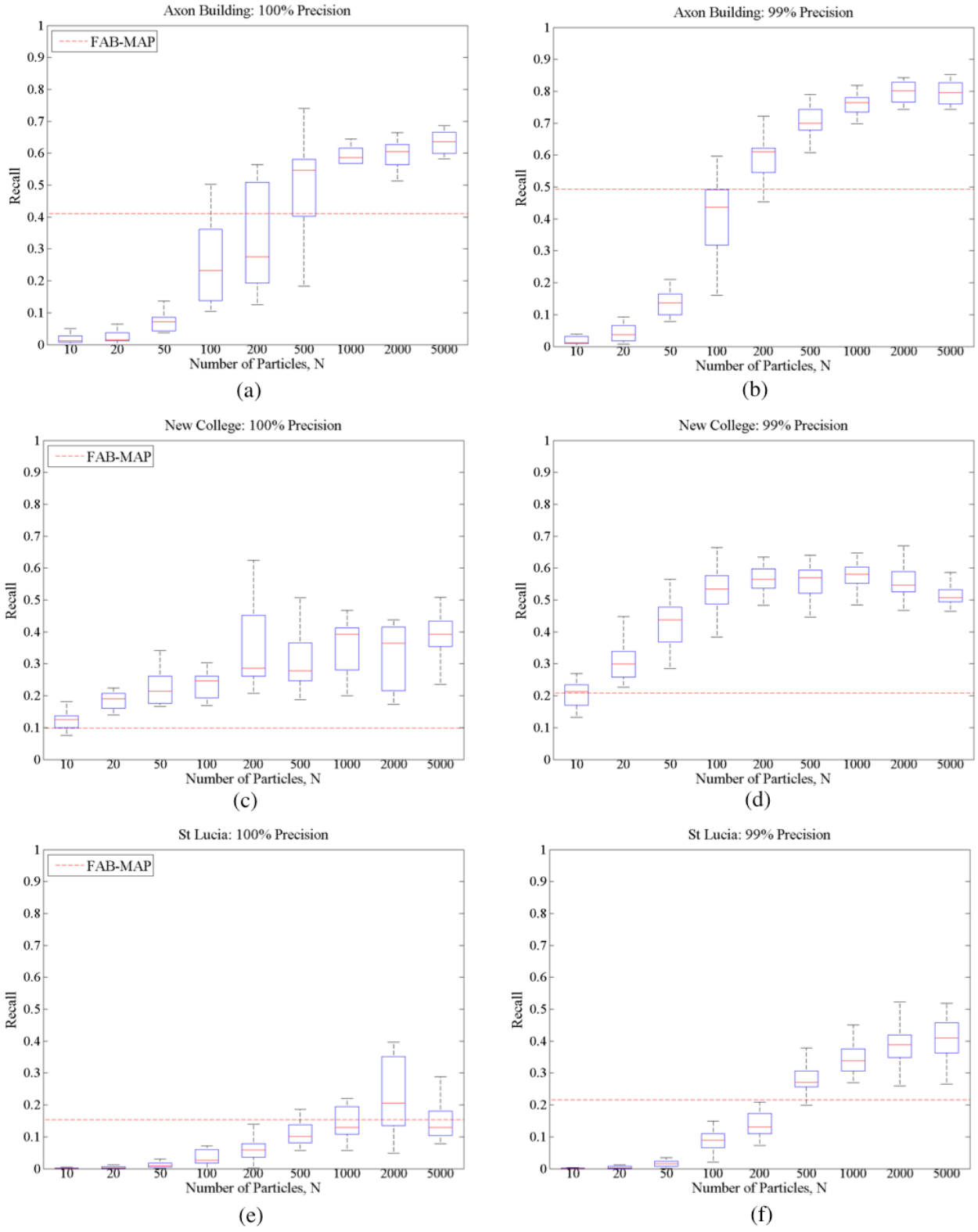


Figure 15 – Recall rates reported over 10 trials for each of the three datasets at both 100% and 99% precision with variation in particle numbers. The ESS threshold was set to 0.3 for consistency with the previous section. The result for FAB-MAP is shown as a baseline for each experiment. Box plots are used to illustrate the spread in recall rates for each combination of parameters.

D. Computational Requirements

The current implementation of CAT-SLAM requires approximately 1 ms per particle update on a single core of a 3GHz Core 2 processor. The majority of this time is spent evaluating the observation likelihood. This performance is comparable to early FAB-MAP implementations (Cummins and Newman 2007), however unlike FAB-MAP the computation requirements do not increase linearly with map size. On all but the largest scale datasets presented in this paper, CAT-SLAM provided equal or better recall performance to FAB-MAP with constant time filter updates. However, as with FAB-MAP, a visual bag-of-words for each past observation must be stored, so the appearance-based map grows in memory requirements in proportion to the number of observations.

7 Discussion

Due to the trajectory following properties of the particles, CAT-SLAM can maintain a hypothesis across a number of frames when supporting visual information above the hypothesis threshold is not available for all frames, as is required for FAB-MAP. This greatly increases the recall rates as entire sections of trajectories can be matched, even when traversed in the opposite direction to the original path.

However, the requirement for a sequence of familiar visual and odometric information reduces the speed at which CAT-SLAM is able to generate a new location hypothesis. While FAB-MAP can localize using only a single frame, CAT-SLAM requires a number of particle update (and possibly resample) stages; revisiting short sections of a path (such as crossing an intersection from a different approach) may not be detected by CAT-SLAM. Additionally, local motion such as on-the-spot rotation and U-turns will not be adequately represented on the continuous trajectory; rotation and translation are explicitly coupled on the trajectory (although traversing paths in the reverse direction and traversing a path backwards will both be correctly tracked by the particles). For applications that do not require 100% precision FAB-MAP may provide more topologically distributed loop closures (and therefore better results), as it does not require an odometric sequence to detect loop closures.

The recall performance obtained with the combination of odometry and appearance-based match information is evident in the results of Figure 11. While the results for the odometry-only variant are predictably low compared to FAB-MAP and other CAT-SLAM variants, it still obtains 50% precision at 100% recall on the indoor dataset. For a system such as FrameSLAM, which does not use appearance information for loop closure, the odometry-only CAT-SLAM variant could provide a computationally inexpensive method of vastly reducing the set of loop closure candidates presented at each update (as it only requires 10us to update the odometry likelihood for each particle).

The computational advantages of a fixed number of particles representing a distribution could allow CAT-SLAM to scale to much larger environments than other appearance-based SLAM systems, provided sufficient particle diversity is maintained by the modified particle resampling scheme. Since the computational requirements per update are directly proportional to the number of particles, there will be a trade-off between recall reliability and computational costs. However, the available computational resources may be specified as a constraint of the sensor platform used for the mapping experiments, and thus for a given filter update rate the number of particles becomes a measurable system parameter. This further reduces the number of arbitrary parameters required by CAT-SLAM and makes it easy to determine the performance and recall reliability on a given computational platform. Additionally, since each particle is independently evaluated, parallelisation using modern GPU cores could provide significant increases in recall reliability for a given platform.

Recent FAB-MAP publications (Cummins and Newman 2008; Cummins and Newman 2009) introduce a number of techniques to increase the speed of visual matches. However, there are a number of underlying assumptions introduced by these techniques, which make them inapplicable to

the CAT-SLAM particle filter. Primarily among these is the assumption that out of all previous locations, one and only one observation should match to the current location. In contrast, CAT-SLAM only forms location hypotheses when close to 100% of the expected observations (generated by each particle) match the current location. In this case, probabilistic bailout conditions that discard potential matches based on relative scores are detrimental to the particle update stage; it is likely that a large number of particles all represent the correct location and thus should all be equally highly weighted. Unfortunately this implies that existing improvements to FAB-MAP cannot be leveraged to provide computational enhancements to CAT-SLAM.

A point raised in (Cummins and Newman 2010) is the effect of the Chow-Liu tree on recall near 100% precision. The authors state that the use of a Chow-Liu tree (in comparison to a naïve Bayes model) does not give consistent recall improvements at 100% precision on larger datasets for FAB-MAP. However, it does increase the similarity measure between difficult matches. CAT-SLAM benefits from the use of a Chow-Liu tree for matches at 100% precision more so than FAB-MAP; as CAT-SLAM integrates a number of matches over a sequence of observations, it is not as sensitive to spurious false positives, and partial matches under difficult conditions would accumulate to form successful location hypotheses.

An aspect not explored in this paper is the use of loop closure events to inform the construction of the appearance-based map. If the current location has a high match probability with a previously visited location, the appearance of both locations could be combined to form a single representation. Additionally, links could be formed in the continuous trajectory that would allow particles to traverse locations where loop closures have been previously detected, forming a continuous appearance-based trajectory graph. This could allow CAT-SLAM to perform localisation and mapping in appearance-space with both constant computation time and constant memory requirements for a given environment; we are not currently aware of any system that meets these specifications. However, these modifications require non-trivial changes to the structure of the particle filter and require a new SLAM interpretation distinct from the current trajectory-based approach.

8 Conclusions

Appearance based SLAM systems, such as FAB-MAP, represent the map using the appearance observed at discrete locations. The novel algorithm presented in this paper, CAT-SLAM, models the appearance at previously visited locations along a continuous trajectory, which allows odometric information to be used to improve the recall of loop closure events. By making use of local metric motion information that appearance-based SLAM systems typically discard, spurious false positives can be rejected, and location hypotheses can be maintained with only partial visual matches. The minimal setup requirements and consistent high performance across the range of environments presented in this paper illustrate the ease with which CAT-SLAM can be applied to a wide range of mapping scenarios. The results of the mapping experiment on three very different datasets demonstrated that the combination of both appearance and motion information in CAT-SLAM provides a clear advantage over appearance-only SLAM systems in terms of recall at 100% precision.

References

Agrawal, M., Konolige, K. and Blas, M. (2008). Censure: Center surround extremas for realtime feature detection and matching. European Conference on Computer Vision, Marseille, France, Springer.

Angeli, A., Doncieux, S., Meyer, J. and Filliat, D. (2009). Visual topological SLAM and global localization. IEEE International Conference on Robotics and Automation, Kobe, Japan.

Angeli, A., Filliat, D., Doncieux, S. and Meyer, J. (2008). "Fast and incremental method for loop-closure detection using bags of visual words." IEEE Transactions on Robotics 24(5): 1027-1037.
Arya, S., Mount, D., Netanyahu, N., Silverman, R. and Wu, A. (1998). "An optimal algorithm for approximate nearest neighbor searching fixed dimensions." Journal of the ACM 45(6): 891-923.
Bay, H., Tuytelaars, T. and Van Gool, L. (2006). Surf: Speeded up robust features. European Conference on Computer Vision, Graz, Austria, Springer.
Blanco, J., Fernandez-Madrigal, J. and Gonzales, J. (2008). "Toward a Unified Bayesian Approach to Hybrid Metric-Topological SLAM." IEEE Transactions on Robotics 24(2): 259-270.
Bosse, M., Newman, P., Leonard, J., Soika, M., Feiten, W. and Teller, S. (2003). An atlas framework for scalable mapping. IEEE International Conference on Robotics and Automation, Taipei, Taiwan, Citeseer.
Bosse, M. and Zlot, R. (2008). "Map matching and data association for large-scale two-dimensional laser scan-based slam." The International Journal of Robotics Research 27(6): 667.
Bosse, M. and Zlot, R. (2010). "Place recognition using regional point descriptors for 3D mapping." Field and Service Robotics: 195-204.
Boucheron, S., Lugosi, G. and Bousquet, O. (2004). "Concentration inequalities." Advanced Lectures on Machine Learning: 208-240.
Chow, C. and Liu, C. (1968). "Approximating discrete probability distributions with dependence trees." IEEE Transactions on Information Theory 14(3): 462-467.
Cummins, M. and Newman, P. (2007). Probabilistic appearance based navigation and loop closing. IEEE International Conference on Robotics and Automation, Rome, Italy.
Cummins, M. and Newman, P. (2008). Accelerated appearance-only SLAM. IEEE International Conference on Robotics and Automation, Pasadena, California.
Cummins, M. and Newman, P. (2008). "FAB-MAP: Probabilistic localization and mapping in the space of appearance." The International Journal of Robotics Research 27(6): 647.
Cummins, M. and Newman, P. (2009). Highly scalable appearance-only SLAM–FAB-MAP 2.0. Robotics: Science and Systems Conference, Seattle, Washington.
Cummins, M. and Newman, P. (2010). "Appearance-only SLAM at large scale with FAB-MAP 2.0." The International Journal of Robotics Research.
Dissanayake, M., Newman, P., Clark, S., Durrant-Whyte, H. and Csorba, M. (2001). "A solution to the simultaneous localization and map building (SLAM) problem." IEEE Transactions on robotics and automation 17(3): 229-241.
Eade, E. (2008). Monocular Simultaneous Localisation and Mapping, Cambridge University. PhD Thesis.
Eade, E. and Drummond, T. (2008). Unified loop closing and recovery for real time monocular slam. European Conference on Computer Vision, Marseille, France.
Eliazar, A. and Parr, R. (2003). DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks. International Joint Conference on Artificial Intelligence, Acapulco, Mexico, Citeseer.
Glover, A., Maddern, W., Milford, M. and Wyeth, G. (2010). FAB-MAP + RatSLAM: Appearance-based SLAM for Multiple Times of Day. IEEE International Conference of Robotics and Automation. Anchorage, Alaska.
Hähnel, D., Fox, D., Burgard, W. and Thrun, S. (2003). A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. IEEE International Conference on Intelligent Robots and Systems, Las Vegas, NV.
Jordan, M. I., Ghahramani, Z., Jaakkola, T. S. and Saul, L. K. (1999). An introduction to variational methods for graphical models. Machine Learning, 37(2): 183–23
Kawewong, A., Tongprasit, N., Tangruamsub, S. and Hasegawa, O. (2010). "Online and Incremental Appearance-based SLAM in Highly Dynamic Environments." The International Journal of Robotics Research 30(1): 33-55.

Koenig, A., Kessler, J. and Gross, H. (2008). A graph matching technique for an appearance-based, visual slam-approach using rao-blackwellized particle filters. IEEE International Conference on Intelligent Robots and Systems.

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

Konolige, K., Bowman, J., Chen, J., Mihelich, P., Calonder, M., Lepetit, V. and Fua, P. (2009). "View-based maps." The International Journal of Robotics Research 29(8): 1-17.

Liu, J., Chen, R. and Logvinenko, T. (2001). "A theoretical framework for sequential importance sampling and resampling." Sequential Monte Carlo Methods in Practice: 225–246.

Lowe, D. (1999). Object recognition from local scale-invariant features. IEEE International Conference on Computer Vision, Corfu, Greece, Published by the IEEE Computer Society.

Maddern, W., Glover, A., Milford, M. and Wyeth, G. (2009). Augmenting RatSLAM using FAB-MAP-based Visual Data Association. Australasian Conference on Robotics and Automation. Sydney, Australia.

Maddern, W., Milford, M. and Wyeth, G. (2010). Performing Loop Closure Detection on a Suburban Road Network using a Continuous Appearance-based Trajectory. Australasian Conference on Robotics and Automation, Brisbane, Australia.

Maddern, W., Milford, M. and Wyeth, G. (2011). Continuous Appearance-based Trajectory SLAM. IEEE International Conference on Robots and Automation, Shanghai, China.

Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B. and Konolige, K. (2010). The Office Marathon: Robust Navigation in an Indoor Office Environment. IEEE International Conference on Robotics and Automation, Anchorage, Alaska.

Mei, C., Sibley, G. and Newman, P. (2010). Closing loops without places. IEEE International Conference on Intelligent Robots and Systems. Taipei, Taiwan.

Milford, M. and Wyeth, G. (2003). Hippocampal models for simultaneous localisation and mapping on an autonomous robot. Australasian Conference on Robotics and Automation, Brisbane, Australia, Citeseer.

Milford, M. and 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. and Wyeth, G. (2008). Single Camera Vision-Only SLAM on a Suburban Road Network. IEEE International Conference on Robotics and Automation, Pasadena, California.

Milford, M. and Wyeth, G. (2009). "Persistent Navigation and Mapping using a Biologically Inspired SLAM System." The International Journal of Robotics Research: (in press).

Milford, M., Wyeth, G. and Prasser, D. (2004). RatSLAM: a hippocampal model for simultaneous localization and mapping. IEEE International Conference on Robotics and Automation, New Orleans, LA.

Milford, M., Wyeth, G. and Prasser, D. (2006). RatSLAM on the edge: Revealing a coherent representation from an overloaded rat brain. IEEE International Conference on Intelligent Robots and Systems, Beijing, China.

Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B. (2002). FastSLAM: A factored solution to the simultaneous localization and mapping problem. Proceedings of the National conference on Artificial Intelligence, Edmonton, Canada, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B. (2003). FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. International Joint Conference on Artificial Intelligence, Acapulco, Mexico, Citeseer.

Newman, P., Cole, D. and Ho, K. (2006). Outdoor SLAM using visual appearance and laser ranging. IEEE International Conference on Robotics and Automation, Orlando, FL, Citeseer.

Newman, P., Sibley, G., Smith, M., Cummins, M., Harrison, A., Mei, C., Posner, I., Shade, R., Schroeter, D. and Murphy, L. (2009). "Navigating, Recognizing and Describing Urban Spaces With Vision and Lasers." The International Journal of Robotics Research 28(11-12): 1406.

Nister, D. and Stewenius, H. (2006). Scalable recognition with a vocabulary tree. IEEE Conference on Computer Vision and Pattern Recognition, New York, NY.

Olson, E. (2008). Robust and efficient robotic mapping. PhD Thesis, Massachussets Institute of Technology.

Paul, R. and Newman, P. (2010). FAB-MAP 3D: Topological Mapping with Spatial and Visual Appearance. IEEE International Conference on Robotics and Automation, Anchorage, Alaska.

Ranganathan, A. and Dellaert, F. (2011). "Online Probabilistic Topological Mapping." The International Journal of Robotics Research.

Sibley, G., Mei, C., Reid, I. and Newman, P. (2010). "Vast-scale Outdoor Navigation Using Adaptive Relative Bundle Adjustment." The International Journal of Robotics Research: (in press).

Sivic, J. and Zisserman, A. (2003). Video Google: A text retrieval approach to object matching in videos. IEEE International Conference on Computer Vision, Nice, France.

Smith, M., Baldwin, I., Churchill, W., Paul, R. and Newman, P. (2009). "The new college vision and laser data set." The International Journal of Robotics Research 28(5): 595.

Teynor, A. and Burkhardt, H. (2007). "Fast Codebook Generation by Sequential Data Analysis for Object Classification." Advances in Visual Computing: 610-620.

Thrun, S. and Leonard, J. (2008). Simultaneous Localization and Mapping. Springer Handbook of Robotics. B. Siciliano, Springer Berlin Heidelberg.

Thrun, S. and Montemerlo, M. (2006). "The graph SLAM algorithm with applications to large-scale mapping of urban structures." The International Journal of Robotics Research 25(5-6): 403.

Van der Merwe, R., Doucet, A., De Freitas, N. and Wan, E. (2001). "The unscented particle filter." Advances in neural information processing systems: 584-590.