Training-Free Probability Models for Whole-Image Based Place Recognition

Stephanie Lowry, Gordon Wyeth and Michael Milford
School of Electrical Engineering and Computer Science
Queensland University of Technology
stephanie.lowry@student.qut.edu.au

Abstract

Whole-image descriptors such as GIST have been used successfully for persistent place recognition when combined with temporal filtering or sequential filtering techniques. However, whole-image descriptor localization systems often apply a heuristic rather than a probabilistic approach to place recognition, requiring substantial environmental-specific tuning prior to deployment. In this paper we present a novel online solution that uses statistical approaches to calculate place recognition likelihoods for whole-image descriptors, without requiring either environmental tuning or pre-training. Using a real world benchmark dataset, we show that this method creates distributions appropriate to a specific environment in an online manner. Our method performs comparably to FAB-MAP in raw place recognition performance, and integrates into a state of the art probabilistic mapping system to provide superior performance to whole-image methods that are not based on true probability distributions. The method provides a principled means for combining the powerful change-invariant properties of whole-image descriptors with probabilistic back-end mapping systems without the need for prior training or system tuning.

1 Introduction

Current appearance-based visual localization systems frequently use probabilistic, Bayesian approaches to determining robot location [Badino, *et al.*, 2012; Cummins and Newman, 2008; Murillo, *et al.*, 2013], and have been shown to perform successfully on many different datasets. These systems require probability models to be generated before localization can be performed. A frequent method of determining an appropriate probability model is to perform a training phase [Badino, *et al.*, 2012; Cummins and Newman, 2008], typically on a similar environment to the test dataset. Once the training data is collected and processed, the computed probability is used for calculation during the testing phase.

In this paper, we present a method that creates a non-parametric probability model online (see Figure 1). The method enables a mobile robotic system to create its own probability model, even while localizing in an unknown environment. We evaluate the place recognition performance of this method on a large real-world dataset, and also present results from a full mapping scenario by combining the method with CAT-SLAM, a state of the art particle filter-based localization system [Maddern, *et al.*, 2012]. We demonstrate that such a system performs comparably to FAB-MAP on a real-world dataset, and out-performs non-probabilistic techniques, particularly in its ability to be integrated with CAT-SLAM.

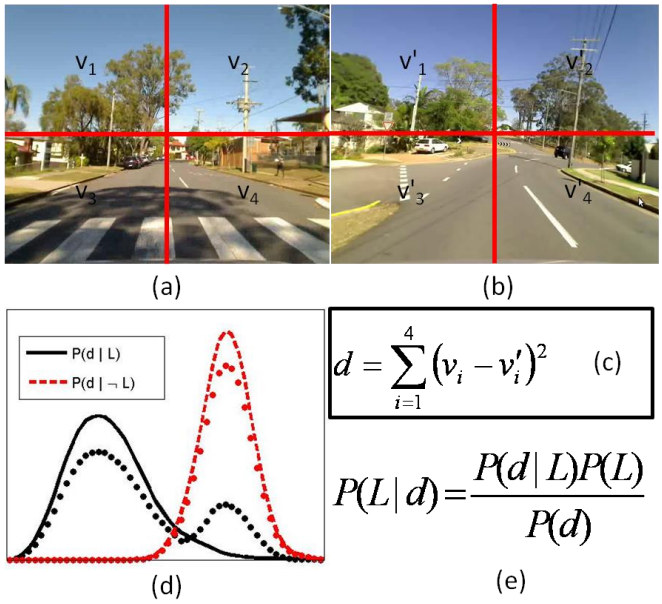

Figure 1. Global image descriptors simplify an image into a vector [v_1 v_2 v_3 v_4] (a),(b). Two image vectors can be compared using a difference function to create a difference value *d* (c). The distribution of image difference values for images taken at the same location (d) (black solid line) and those that come from different locations (red dashed line)can be used to calculate the localization probability (e). We propose to approximate the distributions online (d) (black and red dotted lines) and thus perform localization without prior knowledge of the probability distribution over the environment.

This work was in part funded by the Australian Research Council through Discovery project DP1113006. S. Lowry is supported by an Australian Postgraduate Award and a QUT Vice-Chancellor's Scholarship.

We use global image description methods (referred to as *whole-image descriptors*) as used in [Badino, *et al.*, 2012; Murillo, *et al.*, 2013; Sunderhauf and Protzel, 2011] to perform online probability calculations, rather than the local image description methods such as SIFT [Lowe, 1999] or SURF [Bay, *et al.*, 2006] used in [Cummins and Newman, 2008; Se, *et al.*, 2002].

The paper proceeds as follows. In Section 2, we review approaches to appearance-based visual localization – both probabilistic and non-probabilistic – and summarise the use of whole-image descriptors for visual localization. Section 3 describes the Bayesian probability filter underlying appearance-based visual localization, and presents details of our novel online calculation method. In Section 4 we describe the experimental setup. Section 5 presents results showing successful probabilistic localization using online probability calculations and whole-image descriptors, and provides comparative results to other analytical methods. The paper concludes in Section 6 with discussion and future work.

2 Background

This section surveys the use of whole-image descriptors in visual localization, and the roles of probability and prior knowledge in appearance-based visual localization.

2.1 Whole-Image Descriptors and Visual Localization

Whole-image descriptors of images have long played a key role in topological visual localization systems: early global image processing techniques included colour histograms [Ulrich and Nourbakhsh, 2000] and principal component analysis [Krose, *et al.*, 2001]. More recently, global image descriptors such as GIST [Oliva and Torralba, 2001; Oliva and Torralba, 2006] have been adopted as tools for appearance-based visual localization. GIST has been used successfully for localization on a number of occasions [Murillo and Kosecka, 2009; Siagian and Itti, 2009].

One driver of the use of whole-image descriptors in visual localization has been biological inspiration [Milford and Wyeth, 2010; Siagian and Itti, 2007]. In [Milford and Wyeth, 2010], low-resolution images and simple pixel difference measures are used to compare images; filtering over sequences is then performed. Using such techniques, it has been shown that highly reliable localization can be performed despite poor quality sensors and challenging environmental conditions [Milford, 2013].

Recent research has demonstrated that whole-image description methods can prove very robust to environmental change [Milford and Wyeth, 2012; Milford, 2013; Siagian and Itti, 2009; Sünderhauf, *et al.*, 2013] when used for visual localization. Whilst the addition of a geometric verification step improves the performance of local descriptor-based methods in challenging environments [Cadena and Neira, 2011; Cummins and Newman, 2011; Valgren and Lilienthal, 2010], performance still falls short of whole-image based methods.

Whole-image descriptors do suffer from a disadvantage in comparison to local descriptor methods: in general, they are more susceptible to pose change, which can cause difficulties when using omnidirectional or panoramic images. However, this problem can be somewhat ameliorated by the use of circular shifts as in [Milford and Wyeth, 2008; Murillo, *et al.*, 2013].

2.2 Appearance-Based Probabilistic Localization

A probabilistic interpretation of localization has been a foundation principle underlying SLAM since its inception [Bailey and Durrant-Whyte, 2006; Durrant-Whyte and Bailey, 2006; Thrun and Leonard, 2008]. Appearance-based visual localization gained a probabilistic foundation with the development of FAB-MAP [Cummins and Newman, 2008] which is generally considered to represent the state-of-the art in visual appearance-based localization capability.

Probabilistic localization with whole-image descriptors has been demonstrated in [Badino, *et al.*, 2012; Murillo, *et al.*, 2013]. In [Badino, *et al.*, 2012] sample visual data is used to determine the appropriate likelihood function which is then modelled parametrically and combined with laser range data, whilst [Murillo, *et al.*, 2013] uses an analytic function to approximate the likelihood function.

2.3 Localization and Environmental Knowledge

In order to perform probabilistic localization a probability model of the environment is required. Generally this model is generated via a training phase, whether local features [Cummins and Newman, 2008] or whole-image descriptors [Badino, *et al.*, 2012] are used. This allows a probability model to be generated, but requires prior knowledge of the environment so that a suitable model can be computed.

Heuristic methods of localization do not require a probabilistic model to be generated. However in general such methods still require prior knowledge of the environment, as environment-specific parameters need to be tuned.

In this paper, we investigate a different approach: whether a localization method can be used that assumes no prior knowledge, but instead generates a probability model on the fly for use in the current environment. In the following section, we introduce an algorithm that calculates the probability model for a global image descriptor without prior training.

3 Whole-Image Probabilistic Localization

In this section we present an algorithm for calculating a non-parametric approximation to the true whole-image probability curve without manual intervention. This process can be performed online whilst a robot is operational without requiring a training phase. The method can be used with any underlying whole-image descriptor. We discuss the probabilistic calculations underpinning appearance-based localization in Section 3.1, describe the application of probabilistic calculations to global image descriptors in Section 3.2, and introduce the online generation algorithm in Section 3.3.

3.1 Theory of Appearance-Based Localization

Appearance-based localization is often characterized as a recursive Bayesian filter [Cummins and Newman, 2008; Murillo, *et al.*, 2013]. At each time step *k*, given the current observation I_k and the series of all previous observations I^k, the probability that the current location L_k is the same as a previous location L_i is

$$P(L_k = L_i \mid I^k) = \frac{P(I_k \mid L_k = L_i)P(L_k = L_i \mid I^{k-1})}{P(I_k \mid I^{k-1})} \quad (1)$$

The prior $P(L_k = L_i \mid I^{k-1})$ represents the system's current state of knowledge about the robot's location and is often updated using transition or motion information.

The denominator of Equation (1) does not depend on L_i and so is considered in some cases as a normalizing constant for a given value of *k* [Murillo, *et al.*, 2013]. However, it has significance when comparing probabilities across different values of *k*, as a common feature of appearance-based localization is the inclusion of a "new place" probability [Cummins and Newman, 2008; Murillo, *et al.*, 2013].

The likelihood function $P(I_k \mid L_k = L_i)$ represents the probability that such an image is likely to be observed, if the robot was at location L_i. This function may be represented by an analytic function [Murillo, *et al.*, 2013] or be calculated through prior training [Cummins and Newman, 2008].

3.2 Whole-Image Localization

The Bayesian filter from Equation (1) has been used for whole-image comparison [Murillo, *et al.*, 2013] as well as for feature-based systems [Cummins and Newman, 2008]. However, in this work we redefine the Bayesian probability calculation to make it more appropriate for the available whole-image data. Instead of calculating $P(I_k \mid L_k = L_i)$ we calculate $P(d_{ki} \mid L_k = L_i)$, where d_{ki} is shorthand for the difference function $d(I_k, I_i)$ being used to compare the images.

Given a large enough sample to draw data from, the distributions $P(d_{ki} \mid L_k = L_i)$ and $P(d_{ki} \mid L_k \neq L_i)$ can be determined statistically for an environment (see Figure 2 for an example from a real-world dataset). These two distributions are clearly distinct (although overlapping), and they can be used in the Bayes filter equation to perform place recognition and localization using whole-image descriptors.

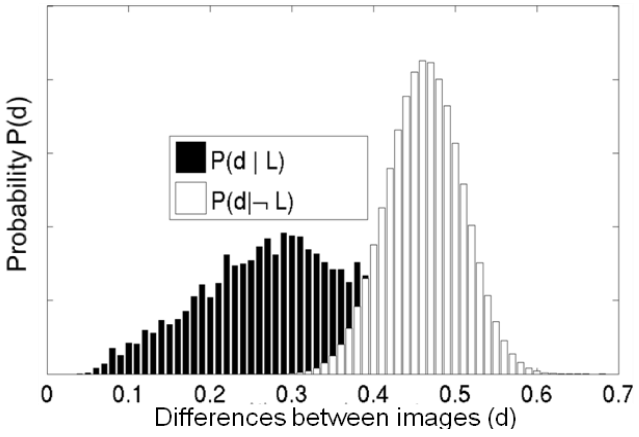

Figure 2. Probability distributions for $P(d_{ki} \mid L_k = L_i)$ (black distribution) and $P(d_{ki} \mid L_k \neq L_i)$ (white distribution) from the St Lucia dataset (see Section 4.1) using a BRISK-based descriptor (see Section 4.2).

The Bayesian filter equation, when updated to refer to the difference function between whole-image descriptors, now becomes:

$$P(L_k = L_i \mid d_{ki}, I^{k-1}) = \frac{P(d_{ki} \mid L_k = L_i)P(L_k = L_i \mid I^{k-1})}{P(d_{ki} \mid I^{k-1})} \quad (2)$$

The two key differences between Equation (1) and Equation (2) are as follows: Equation (1) depends on $P(I_k \mid L_k = L_i)$, the probability of seeing the current *observation* given the robot is at location L_i, while Equation (2) depends on $P(d_{ki} \mid L_k = L_i)$, the probability of seeing the *difference* between the current observation and the previous observation at L_i. The second difference is that the denominator now contains an explicit dependence on the value of *i*. Therefore, the denominator can no longer be considered a normalizing constant over *k*, but must be calculated independently for each L_i.

We note that the denominator of Equation (2) can be re-written using the marginal distribution, as shown in Equation (3) (using the simplified notation $P(d \mid L)$ for $P(d_{ki} \mid L_k = L_i)$, and $P(d \mid \neg L)$ for $P(d_{ki} \mid L_k \neq L_i)$).

$$P(d) = P(d \mid L)P(L) + P(d \mid \neg L)P(\neg L) \quad (3)$$

From Equations (2) and (3) is can be seen that the recursive Bayesian equation for whole-image localization depends on the calculation of $P(d_{ki} \mid L_k = L_i)$ and $P(d_{ki} \mid L_k \neq L_i)$; that is, the two distributions depicted in Figure 2. In the following section we present a method for a robot to approximate these distributions in a training-free manner, whilst simultaneously localizing in an unknown environment.

3.3 Generating an Online Probability Model

In this section we introduce a method for generating an online probability model for whole-image localization. To generate the required distributions $P(d \mid L)$ and $P(d \mid \neg L)$ online the system adds, at each time step *k*, one sample to the each distribution. In order to select a suitable sample value for each, we use an assumption, which underlies all whole-image descriptors localization systems. This assumption is that, in general, a lower value of d_{kj} implies a more probable location match. Figure 2 demonstrates that this assumption is a reasonable one in this particular environment. The success of whole-image descriptor methods in other environments [Badino, *et al.*, 2012; Murillo, *et al.*, 2013] also supports this assumption.

We therefore propose to estimate the distribution $P(d \mid L)$ by adding to it, at time *k*, the sample d_{kj} where d_{kj} is the smallest difference value; that is:

$$d_{kj} = \min\{d_{ki} \mid i = 1,2,\ldots,k-1\}. \quad (4)$$

To the distribution $P(d \mid \neg L)$ we propose to add the next-smallest difference value from $\{d_{ki} \mid i = 1,2,\ldots,k-1\}$.

The algorithm has an initialization stage; this provides a baseline result for $P(d \mid \neg L)$ and stops erroneous matching to $P(d \mid L)$ before a meaningful probability distribution has been generated. During the initialization stage, only $P(d \mid \neg L)$ is updated, not $P(d \mid L)$. As a consequence, $P(L \mid d)$ will be zero for all

d during the initialization stage, and so no localization will take place. The initialization stage can be quite brief; in our experiments we found an initialization stage of around 100-500 time steps was sufficient.

Once the initialization stage is complete, the algorithm performs two processes at each time step: a localization calculation and a probability update. The algorithm therefore allows a system to both localize *and* to generate the probability models required to do so, simultaneously. A pseudocode description of this algorithm is included in the Appendix.

4 Experiment Setup

In the following experiments, we compare the online probability algorithm described in Section 3 to non-probabilistic whole-image descriptor methods, and also to FAB-MAP. We first perform a global localization experiment, and then use each method as input to CAT-SLAM [Maddern, *et al.*, 2012], a particle filter based visual SLAM system.

4.1 Testing Environment

The experimental environment was a large outdoor dataset first presented in [Glover, *et al.*, 2010]. This dataset provided visual and GPS data from a car driven around suburban streets (see Figure 3). The distance of the path traversed was approximately 15 kilometres. The GPS data was logged at 1 Hz and used for ground truth.

Figure 3. Ground truth of St Lucia dataset. (Imagery ©2012 Cnes/Spot Image, DigitalGlobe,GeoEye, Sinclair Knight Merz & Fugro).

4.2 Image Processing

The images were captured by a forward-facing commercial web camera at a resolution of 640×480 pixels and at an average of 4 frames per second. Figure 4 displays a few sample images of the environment.

Figure 4. Sample images from St Lucia dataset.

Two whole-image comparison techniques were used: GIST [Oliva and Torralba, 2001], and a GIST-inspired image descriptor similar to BRIEF-Gist [Sunderhauf and Protzel, 2011] but using the feature descriptor BRISK [Leutenegger, *et al.*, 2011] instead of BRIEF [Calonder, *et al.*, 2012].

GIST

From each image, a GIST vector of length 512 was computed: the image was partitioned into 4 × 4 tiles, each of which was represented by the average of filters at 8 orientations and 4 scales, using the implementation from [Oliva and Torralba, 2001] (downloaded from http://people.csail.mit.edu/torralba/code/spatialenvelope/). The difference between two images was defined as the sum of squared differences between the images' respective GIST descriptors g_i and g_j:

$$d_{ij} = \sum_{n=1}^{512} \left(g_i(n) - g_j(n)\right)^2 \quad (5)$$

BRISK

For the BRISK descriptor each image was partitioned into 5 × 5 tiles, each of size 48 × 48 pixels, and a 512-bit BRISK feature descriptor was calculated around the centre of each tile using the OpenCV [Bradski, 2000] implementation. Each image was thus represented by a 12800 binary vector.

Image comparison was performed by measuring the Hamming distance between the (whole-image) BRISK image vectors b_i and b_j as in [Sunderhauf and Protzel, 2011]:

$$d_{ij} = \sum_{n=1}^{12800} b_i(n) \oplus b_j(n) \quad (6)$$

Note that in the remainder of this paper, when we refer to BRISK we are referring to the whole-image version described here.

4.3 Experiment 1

The first experiment tested the effectiveness of the online probability algorithm at performing global localization. The experiment was performed on the St Lucia dataset using both GIST and whole-image BRISK as described in Section 4.2. The online probability algorithm described in Section 3.3 was used to calculate $P(L \mid d)$.

The performance of the online probability model was compared to place recognition performance using an analytic function to approximate the probability. The function chosen was $\exp(-d_{ij})$; this equation is based on the likelihood used in [Murillo, *et al.*, 2013]. Note that both the probabilistic model and the analytic model use the same input; that is, the difference value d_{ij}. The only difference is the way in which the output is calculated (either using the analytic function or using the generated probability models).

As both versions access identical visual input (the difference value d_{ij}), we would expect similar precision and recall values for both systems. However, to use a localization system in a real-world situation, it is often necessary to select a cut-off threshold. In non-probabilistic localization systems, a cut-off threshold parameter is often selected via tuning prior to deployment; conversely, the stated goal of the current system is to require as little environment-specific tuning as possible. With this in mind,

we compare the performance of the probabilistic and the analytic functions across various cut-off thresholds.

The experiment compares F-scores for each system against threshold value. An F-score [van Rijsbergen, 1979] is a weighted average of precision P and recall R:

$$F_\beta = (1+\beta^2)\frac{P \times R}{\beta^2 P + R} \quad (7)$$

The parameter β determines the relative importance placed on precision P and recall R - common values of β are 0.5 (for greater emphasis on precision), 1 (for equal important) and 2 (which weights recall more than precision).

The online probability model was also compared to FAB-MAP, using the implementation from [Maddern, *et al.*, 2012]. A codebook of 10000 words based on SURF [Bay, *et al.*, 2006] features was created. To ensure the system was trained under optimal conditions, the codebook was generated from image data captured from the test environment.

4.4 Experiment 2

The second experiment tested the performance of whole-image probability models when integrated with an appearance-based visual localization and mapping system, CAT-SLAM [Maddern, *et al.*, 2012]. The original version of CAT-SLAM uses a FAB-MAP-derived visual probability as input to a particle filter; and as for FAB-MAP, a training phase is required during which SURF descriptors are extracted and a bag-of-words model of the environment is generated.

For this experiment, the standard FAB-MAP-based version of CAT-SLAM was compared to a version of CAT-SLAM that accepted a whole-image probability input. The original code for CAT-SLAM was converted to accept a whole-image probability input. As for Experiment 1, both BRISK and GIST descriptors were tested. For the FAB-MAP version, the codebook from Experiment 1 was also used for Experiment 2.

Furthermore, the whole-image version of CAT-SLAM was also tested with an analytic function approximation of probability in place of the probabilistic distribution. As for Experiment 1, the analytic function used was $\exp(-d_{ij})$.

In order to ensure a fair comparison, all CAT-SLAM parameters were kept the same across the different systems. Parameters were chosen to match those used on this dataset in [Maddern, *et al.*, 2012], where the standard FAB-MAP-based version of CAT-SLAM was presented. It was expected that the parameters would therefore be optimized for the FAB-MAP version of CAT-SLAM rather than any whole-image-based version.

5 Results

In this section we present results for Experiments 1 and 2. Precision and recall results are provided for both experiments. Experiment 1 also compares the F-scores of the probabilistic model with the F-scores of the analytic functions, and loop closure maps are used to display the distribution of image matches across the environment for Experiment 2.

5.1 Experiment 1

The precision recall curves of the probabilistic model and analytic model of localization are displayed in Figure 5. As expected, the probabilistic and analytic versions perform similarly. From this we can conclude that the probabilistic model does not lose localization power relative to an analytic version. Furthermore, both whole-image descriptors out-perform FAB-MAP (using local feature SURF) at 100% precision and 99% precision for this dataset.

The results of F-score comparison against threshold value are displayed in Figure 6. Notably, the probabilistic F-score (denoted as a solid black line) is significantly more consistent across a range of threshold values than the analytic version F-score (denoted as a dashed red line) for both the $F_{0.5}$-score and F_1-score. This consistency holds for both the GIST and the BRISK descriptors. Furthermore, for almost all threshold values between 0.4 and 1 the probabilistic model has a consistently higher F-score than the analytic model, except in a very narrow threshold range.

The results are not so clear-cut for the F_2-score. For the GIST descriptor, the analytic version performs slightly better. For the BRISK descriptor, the analytic version performs slightly better for lower threshold values and worse for higher values. However, the probabilistic F_2-score still maintains relatively good consistency across a wide threshold range.

These results illustrate the difficulty of selecting threshold values for non-probabilistic functions and shows how dependent on threshold selection the performance of such systems are. As can be seen from Figure 6, the analytic functions perform as well or better than the probabilistic versions only when a threshold value is carefully tuned using the test data (around 0.99 for analytic GIST and around 0.75 for analytic BRISK appear to be optimal values). Away from the correct threshold values the F_1-score and $F_{0.5}$-score values drop off significantly, meaning performance is brittle. Furthermore, such optimal threshold values cannot necessarily be predicted ahead of time, as they may depend not only on descriptor type, but also on the particular environment being traversed as well. In contrast, the probabilistic model shows less sensitivity to an appropriate choice of threshold, which is an advantage for an autonomous mobile system operating long-term.

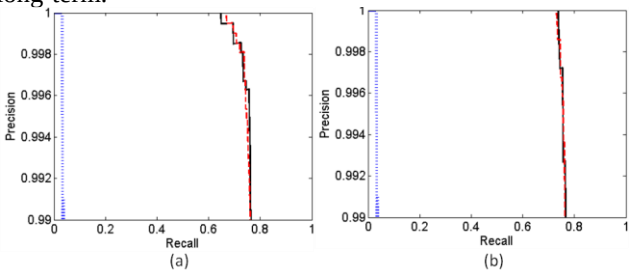

Figure 5. Precision-recall curve for St Lucia using a probabilistic model (solid black line) and an analytic model (dashed red line) and FAB-MAP (dotted blue line) for GIST descriptors (a) and whole-image BRISK descriptors (b). The precision and recall curves are very similar for the probabilistic and the analytic versions, for both descriptor types.

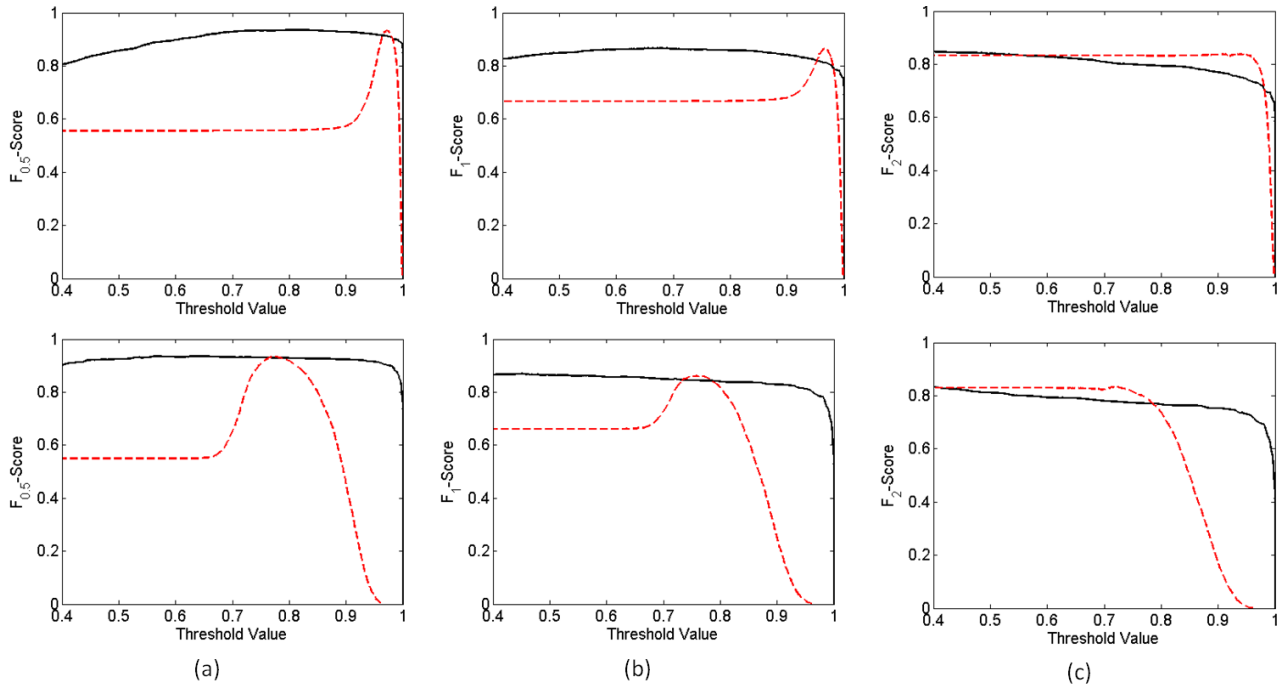

Figure 6. F-Scores for localization against threshold value using a probabilistic model (solid black line) and an analytic model (dashed red line) for GIST descriptors (top line) and whole-image BRISK descriptors (bottom line). From left to right: $F_{0.5}$-score (a), F_1-score (b) and F_2-score (c).

5.2 Experiment 2

In this section we present results CAT-SLAM using whole-image descriptors (both probabilistically and with functions) and from CAT-SLAM using FAB-MAP. Figure 7 displays the precision recall curves for CAT-SLAM using FAB-MAP, probabilistic GIST, and probabilistic whole-image BRISK. The probabilistic whole-image functions both perform comparably with FAB-MAP.

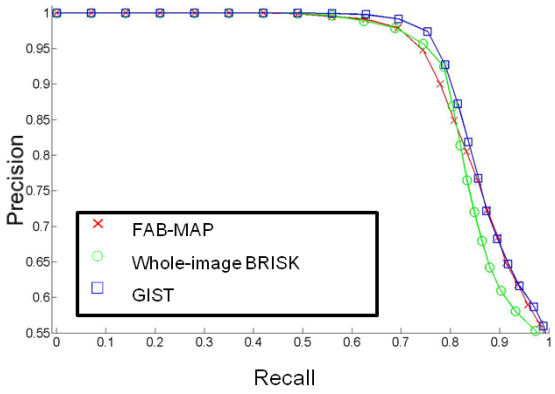

Figure 7. Precision-Recall curve for CAT-SLAM using FAB-MAP (red crosses), whole-image BRISK (green circles), and GIST (blue squares) for probabilistic visual input calculations

A summary of the performance of each input style is included in Table I; BRISK performs equally well to FAB-MAP, and GIST performs even better, achieving over 10% better recall at 100% precision.

TABLE I
PRECISION-RECALL USING CAT-SLAM

Visual input type	Recall at 100% Precision	Recall at 99% Precision
FAB-MAP	47.22%	67.64%
BRISK	48.61%	64.85%
GIST	59.21%	74.28%

We note that neither BRISK or GIST performs as well in combination with CAT-SLAM than in the standalone place recognition test in Experiment 1, whereas CAT-SLAM increases the performance of FAB-MAP substantially (see Section 5.1 and [Maddern, *et al.*, 2012]). We are unsure why this is the case, although we note that FAB-MAP's output is often very sharply defined between highly probable places and all other places, whereas the whole-image probability curves represent a smoother, more gradual model. However, further investigation is needed to determine the true cause behind this difference.

The precision-recall curves for the analytic versions of GIST and BRISK are shown in Figure 8. Neither analytic function performs successfully with CAT-SLAM at all: despite all experimental parameters being kept the same across each test, the analytic GIST function achieves 3% precision at best, and the analytic BRISK function performs even more poorly, not even reaching 1% precision. This is in contrast to the performance of the analytic functions in Experiment 1, where it was shown that each could achieve high precision and high recall (see Figure 5) *if* a judicious choice of threshold was made (see Figure 6). However, when combined with a probabilistic particle filter such as CAT-SLAM the benefits of a probabilistic model over a non-probabilistic approximation can be seen very clearly.

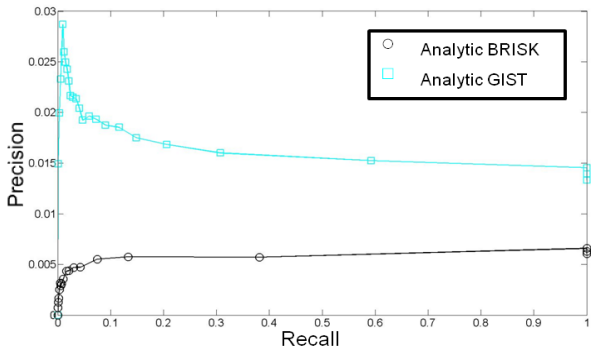

Figure 8. Precision-Recall curve for CAT-SLAM using analytic BRISK (black circles) and analytic GIST (blue squares) for visual input calculations.

Figure 9 shows a map of loop closures generated by CAT-SLAM using probabilistic GIST. The red lines denote loop closures and the thick green lines denote the parts of the environment that are only visited once (and so in which no loop closures are possible). No incorrect loop closures were generated. Probabilistic GIST generates a wide distribution of loop closures over almost all the portions of the environment where a loop closure is possible.

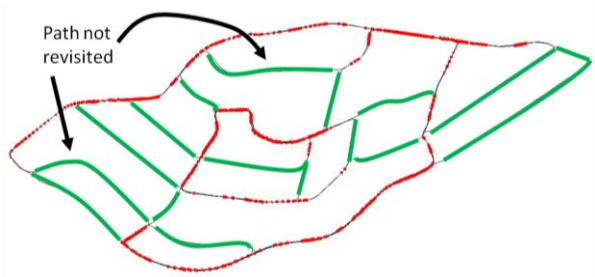

Figure 9. Ground truth loop closure plot for CAT-SLAM using probabilistic GIST. Red lines denote loop closures and green lines denote parts of the environment that are only visited once.

Figure 10 shows the loop closures generated by CAT-SLAM using FAB-MAP. CAT-SLAM using FAB-MAP also has a good distribution of loop closures, although not as dense as GIST. As for whole-image CAT-SLAM, no incorrect loop closures were generated. In some areas the coverage is better than that observed in Figure 9. These areas are noted using orange circles. However, there are some regions where the coverage is outperformed by GIST. These areas are shown by the blue circles.

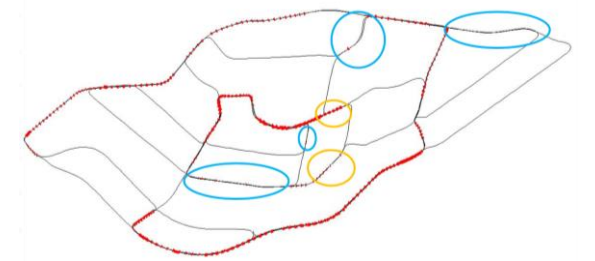

Figure 10. Ground truth loop closure plot for CAT-SLAM using FAB-MAP. Red lines denote loop closures. Blue circles denote where CAT-SLAM using GIST outperforms CAT-SLAM using FAB-MAP. Orange circles denote where CAT-SLAM with FAB-MAP is superior.

6 Discussion and Future Work

In this paper we have presented a novel method for computing probability curves for whole-image descriptors of an environment. This method does not require prior knowledge about the environment and allows a robot to perform visual localization without the necessity of a training step. This simultaneous localization and training technique has been shown to perform well using two different whole-image descriptors. It is robust to the choice of threshold values, and can be used to replace FAB-MAP in CAT-SLAM, a probabilistic localization system.

A future extension for this work is to generate a "quality control" measure for the created probability distributions to ensure they retain discriminative power; that is, to ensure that there is a measurable difference between the distributions $P(d \mid L)$ and $P(d \mid \neg L)$. This would allow the system to be applied to persistent localization problems, particularly within environments that experience environmental change, by providing an internal check that the localization model being used is still relevant.

Furthermore, if an environment changes significantly, it may be necessary to have more localised distributions; for example, if a robot has been traversing an environment during the day and the night, a separate distribution might be appropriate for each; otherwise the general similarities (i.e. "daytime" and "night-time") might saturate the distributions and remove the more subtle local similarities.

Appendix

The online probability generation and localization algorithm described in Section 3.3 is presented below. The following notation is assumed:

$\mathbf{b_m}$	one of the $\boldsymbol{s}$ bins representing $P(d_{ki} \mid L_k = L_i)$. Bin $\boldsymbol{b_m}$ is the bin which holds value $\boldsymbol{m}$.
$\mathbf{c_m}$	one of the $\boldsymbol{s}$ bins representing $P(d_{ki} \mid L_k \neq L_i)$.
$\mathbf{d_k}$	a vector containing the difference values between the current observations $\boldsymbol{I_k}$ and all previously seen observations $\boldsymbol{I_1,\ldots,I_{k-1}}$

The following pseudocode describes the algorithm behaviour at step k. The algorithm takes d_k as input. If k is less than the user-defined *startFrame* value, the **initialize** function is called, which updates $\mathbf{c_m}$, the representation of $P(d_{ki} \mid L_k \neq L_i)$. If k is greater than *startFrame*, the place recognition calculation is performed by the function **calcProb**, and then both the probability models $P(d_{ki} \mid L_k = L_i)$ and $P(d_{ki} \mid L_k \neq L_i)$ are updated via the function **updateProb**.

```
begin
    function mainFunction(d_k) :
        (m,z):=min(d_k);
        if k < startFrame :
            initialize(m);
        else :
            calcProb(m);
            updateProb(m,z,d_k);
        end
    end

    function initialize(m) :
        c_m := c_m + 1;
    end

    function calcProb(m) :
        return b_m / (b_m + c_m);
    end

    function updateProb(m,z,d_k) :
        d_k(z):= ∞;
        (m',z'):=min(d_k);
        b_m := b_m + 1;
        c_m' := c_m' + 1;
    end
end
```

Acknowledgements

We thank Arren Glover for the use of the St Lucia dataset. This dataset is available for download at https://wiki.qut.edu.au/display/cyphy/St+Lucia+Multiple+Times+of+Day. We also thank Will Maddern for access to the CAT-SLAM / CAT-Graph code base.

References

[Badino, *et al.*, 2012] H. Badino, D. Huber and T. Kanade. Real-time topometric localization. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1635-1642, 2012

[Bailey and Durrant-Whyte, 2006] T. Bailey and H. Durrant-Whyte. Simultaneous Localization and Mapping (SLAM): Part II. *Robotics & Automation Magazine, IEEE*, 13(3):108-117, 2006

[Bay, *et al.*, 2006] H. Bay, T. Tuytelaars and L. Van Gool. SURF: Speeded up robust features. *Computer Vision - ECCV*, pages 404-417, 2006

[Bradski, 2000] G. Bradski. Programmer's tool chest: The OpenCV library. *Dr Dobb's Journal of Software Tools*, 25(11):120-126, 2000

[Cadena and Neira, 2011] C. Cadena and J. Neira. A Learning Algorithm for Place Recognition. In *ICRA 2011 Workshop on Long-term Autonomy*, Shanghai, China, 2011

[Calonder, *et al.*, 2012] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha and P. Fua. BRIEF: Computing a Local Binary Descriptor Very Fast. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 778-792, 2012

[Cummins and Newman, 2008] M. Cummins and P. Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *International Journal of Robotics Research*, 27(6):647-665, 2008

[Cummins and Newman, 2011] M. Cummins and P. Newman. Appearance-only SLAM at Large Scale with FAB-MAP 2.0. *International Journal of Robotics Research*, 30(9):1100-1123, 2011

[Durrant-Whyte and Bailey, 2006] H. Durrant-Whyte and T. Bailey. Simultaneous Localization and Mapping: Part I. *Robotics & Automation Magazine, IEEE*, 13(2):99-110, 2006

[Glover, *et al.*, 2010] A. Glover, W. Maddern, M. Milford and G. Wyeth. FAB-MAP + RatSLAM: Appearance-based SLAM for multiple times of day. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3507-3512, 2010

[Krose, *et al.*, 2001] B. Krose, N. Vlassis, R. Bunschoten and Y. Motomura. A probabilistic model for appearance-based robot localization. *Image and Vision Computing*, 19(6):381-391, 2001

[Leutenegger, *et al.*, 2011] S. Leutenegger, M. Chli and R. Siegwart. BRISK: Binary robust invariant scalable keypoints. *IEEE International Conference on Computer Vision (ICCV)*, pages 2548-2555, 2011

[Lowe, 1999] D. Lowe. Object recognition from local scale-invariant features. *IEEE International Conference on Computer Vision (ICCV)*, pages 1150-1157, 1999

[Maddern, *et al.*, 2012] W. Maddern, M. Milford and G. Wyeth. CAT-SLAM: Probabilistic Localisation and Mapping using a Continuous Appearance-based Trajectory. *International Journal of Robotics Research*, 31(4):429-451, 2012

[Milford and Wyeth, 2010] M. Milford and G. Wyeth. Persistent Navigation and Mapping using a Biologically Inspired SLAM System. *The International Journal of Robotics Research*, 29(9):1131-1153, 2010

[Milford and Wyeth, 2012] M. Milford and G. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1643-1649, 2012

[Milford, 2013] M. Milford. Vision-based place recognition: how low can you go? *The International Journal of Robotics Research*, 32(7):766-789, 2013

[Milford and Wyeth, 2008] M. Milford and G. Wyeth. Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System. *Robotics, IEEE Transactions on*, 24(5):1038-1053, 2008

[Murillo and Kosecka, 2009] A. Murillo and J. Kosecka. Experiments in place recognition using gist panoramas. *IEEE International Conference on Computer Vision (ICCV)*, pages 2196-2203, 2009

[Murillo, *et al.*, 2013] A. Murillo, G. Singh, J. Kosecka and J. Guerrero. Localization in Urban Environments Using a Panoramic Gist Descriptor. *Robotics, IEEE Transactions on*, 29(1):146-160, 2013

[Oliva and Torralba, 2001] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145-175, 2001

[Oliva and Torralba, 2006] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155(23-36, 2006

[Se, *et al.*, 2002] S. Se, D. Lowe and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735-758, 2002

[Siagian and Itti, 2007] C. Siagian and L. Itti. Rapid biologically-inspired scene classification using features shared with visual attention. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(2):300-312, 2007

[Siagian and Itti, 2009] C. Siagian and L. Itti. Biologically inspired mobile robot vision localization. *Robotics, IEEE Transactions on*, 25(4):861-873, 2009

[Sunderhauf and Protzel, 2011] N. Sunderhauf and P. Protzel. BRIEF-Gist - closing the loop by simple means. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1234-1241, 2011

[Sünderhauf, *et al.*, 2013] N. Sünderhauf, P. Neubert and P. Protzel. Are We There Yet? Challenging SeqSLAM on a 3000 km Journey Across All Four Seasons. *Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*, pages 2013

[Thrun and Leonard, 2008] S. Thrun and J. Leonard. *Simultaneous Localisation and Mapping*, pp. 871-889, in *Springer Handbook of Robotics*, 2008

[Ulrich and Nourbakhsh, 2000] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1023-1029, 2000

[Valgren and Lilienthal, 2010] C. Valgren and A. Lilienthal. SIFT, SURF & seasons: Appearance-based long-term localization in outdoor environments. *Robotics and Autonomous Systems*, 58(2):157-165, 2010

[van Rijsbergen, 1979] C. van Rijsbergen. *Information Retrieval*. Butterworths, 1979