# FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance

Mark Cummins and Paul Newman

Oxford University Mobile Robotics Research Group. {mjc,pnewman}@robots.ox.ac.uk

April 10, 2008

**Abstract**

This paper describes a probabilistic approach to the problem of recognising places based on their appearance. The system we present is not limited to localization, but can determine that a new observation comes from a previously unseen place, and so augment its map. Effectively this is a SLAM system in the space of appearance. Our probabilistic approach allows us to explicitly account for perceptual aliasing in the environment – identical but indistinctive observations receive a low probability of having come from the same place. We achieve this by learning a generative model of place appearance. By partitioning the learning problem into two parts, new place models can be learned online from only a single observation of a place. The algorithm complexity is linear in the number of places in the map, and is particularly suitable for online loop closure detection in mobile robotics.

## 1 Introduction

This paper considers the problem of recognising locations based on their appearance. This problem has recently received attention in the context of large-scale global localization [Schindler et al., 2007] and loop-closure detection in mobile robotics [Ho and Newman, 2007]. This paper advances the state of the art by developing a principled probabilistic framework for appearance-based place recognition. Unlike many existing systems, our approach is not limited to localization – we are able to decide if new observations originate from places already in the map, or rather from new, previously unseen places. This is possible even in environments where many places have similar sensory appearance (a problem known as perceptual aliasing). Previous solutions to this problem [Ho and Newman, 2007] had complexity cubic in the number of observations, whereas the algorithm presented here has linear complexity, extending its applicability to much larger environments. We refer to the new algorithm as FAB-MAP, for Fast Appearance-Based Mapping.

Our basic approach is inspired by the bag-of-words image retrieval systems developed in the computer vision community [Sivic and Zisserman, 2003, Nister and Stewenius, 2006]. However, we extend the approach by learning a generative model for the bag-of-words data. This generative model captures the fact that certain combinations of appearance words tend to co-occur, because they are generated by common objects in the environment. Our system effectively learns a model of these common objects

(without supervision, see Figure 5), which allows us to improve inference by reasoning about which sets of features are likely to appear or disappear together (for example, see Figure 12). We present results which demonstrate that the system is capable of recognising places even when observations have few features in common (as low as 8%), while rejecting false matches due to perceptual aliasing even when such observations share many features (as much as 45%). The approach also has reasonable computational cost – fast enough for online loop closure detection in realistic mobile robotics problems where the map contains several thousand places. We demonstrate our system by detecting loop closures over a 2km path length in an initially-unknown outdoor environment, where the system detects a large fraction of the loop closures without false positives.

## 2 Related Work

Due to the difficulty of detecting loop closure in metric SLAM algorithms, several appearance-based approaches to this task have recently been described [Newman et al., 2006, Levin and Szeliski, 2004, Ranganathan and Dellaert, 2006, Se et al., 2005]. These methods attempt to determine when the robot is revisiting a previously mapped area on the basis of sensory similarity, which can be calculated independently of the robot's estimated metric position. Thus similarity-based approaches may be robust even in situations where the robot's metric position estimate is in gross error.

Many issues relevant to an appearance-based approach to SLAM have previously been examined in the context of appearance-based localization systems. In particular, there has been extensive examination of how best to represent and match sensory appearance information. Several authors have described systems that apply dimensionality reduction techniques to process incoming sensory data. Early examples include representing appearance as a set of image histograms [Ulrich and Nourbakhsh, 2000], and as ordered sequences of edge and colour features [Lamon et al., 2001]. More recently Torralba et al. [Torralba et al., 2003] represent places by a set of texture features, and describe a system where localization is cast as estimating the state of a hidden Markov model. Kröse et al. [Kröse et al., 2001] use Principal Component Analysis to reduce the dimensionality of the incoming images. Places are then represented as a Gaussian density in the low dimensional space, which enables principled probabilistic localization. Ramos et al. [Ramos et al., 2005] employ a dimensionality reduction technique combined with variational Bayes learning to find a generative model for each place. (A drawback of both of these approaches is that they require significant supervised training phases to learn their generative place models — an issue which we will address in this paper.) Bowling et al. [Bowling et al., 2005] describe an unsupervised approach that uses a sophisticated dimensionality reduction technique called Action Respecting Embedding; however the method suffers from high computational cost [Bowling et al., 2006] and yields localization results in a "subjective" representation which is not straight forward to interpret.

The methods described so far represent appearance using global features, where a single descriptor is computed for an entire image. Such global features are not very robust to effects such as variable lighting, perspective change and dynamic objects that cause portions of a scene to change from visit to visit. Work in the computer vision community has led to the development of local features which are robust to transformations such as scale, rotation and some lighting change, and in aggregate allow

object recognition even in the face of partial occlusion of the scene. Local feature schemes consist of a region of interest detector combined with a descriptor of the local region, SIFT [Lowe, 1999] being a popular example. Many recent appearance-based techniques represent sensory data by sets of local features. An early example of the approach was described in [Sim and Dudek, 1998]. More recently, [Wolf et al., 2005] used an image retrieval system based on invariant features as the basis of a Monte Carlo localization scheme. The issue of selecting the most salient local features for localization was considered in [Li and Kosecka, 2006]. Wang et al. employ the idea of a visual vocabulary [Wang et al., 2005] built upon invariant features, taking inspiration from work in the computer vision community [Sivic and Zisserman, 2003, Squire et al., 2000]. The visual vocabulary model treats an image as a "bag of words" much like a text document, where a "word" corresponds to a region in the space of invariant descriptors. While the bag-of-words model discards geometric information about the image, it enables fast visual search through the application of methods developed for text retrieval. Several authors have proposed extensions to this basic approach. Wang et al. use the geometric information in a post-verification step to confirm putative matches. Filliant described a system where the visual vocabulary is learned online Filliat [2007]. Recent work by Ferreira et al. employs a Bernoulli mixture model to capture the conditional dependencies between words in the vocabulary [Ferreira et al., 2006]. Most recently Schindler et al. have described how to tailor visual vocabulary generation so as to yield more discriminative visual words [Schindler et al., 2007], and discuss the application of the technique to city-scale localization with a database of 10 million images.

The work discussed so far has been mainly concerned with the localization problem, where the map is known a-priori and the current sensory view is guaranteed to come from somewhere within the map. To use a similar appearance-based approach to detect loop closure in SLAM, the system must be able to deal with the possibility that the current view comes from a previously unvisited place and has no match within the map. As pointed out in [Gutmann and Konolige, November 1999], this makes the problem considerably more difficult. Particularly challenging is the *perceptual aliasing* problem — the fact that different parts of the workspace may appear the same to the robot's sensors. As noted in [Silpa-Anan and Hartley, 2004], this can be a problem even when using rich sensors such as cameras due to repetitive features in the environment, for example mass-produced objects or repeating architectural elements. Gutmann and Konolige [Gutmann and Konolige, November 1999] tackle the problem by computing a similarity score between a patch of the map around the current pose and older areas of the map to identify possible loop-closures. They then employ a set of heuristics to decide if the putative match is a genuine loop closure or not. Chen and Wang [Chen and Wang, 2006] tackle the problem in a topological framework, using a similarity measure similar to that developed by Kröse [Kröse et al., 2001]. To detect loop closure they integrate information from a sequence of observations to reduce the effects of perceptual aliasing, and employ a heuristic to determine if a putative match is a loop closure. While these methods achieved some success, they did not provide a satisfactory solution to the perceptual aliasing problem. Goedemé described an approach to this issue using Dempster-Shafer theory, using sequences of observations to confirm or reject putative loop closures [Goedemé, 2006]. The approach involved separate mapping and localization phases, so was not suitable for unsupervised mobile robotics applications.

Methods based on *similarity matrices* have recently become a popular way to extend appearance-

based matching beyond pure localization tasks. These methods define a similarity score between observations, then compute the pairwise similarity between all observations to yield a square similarity matrix. If the robot observes frequently as it moves through the environment and observations are ordered by time, then loop closures appear as off-diagonal stripes in this matrix. Levin and Szeliski describe such a method [Levin and Szeliski, 2004] that uses a cascade of similarity functions based on global colour histograms, image moments and epipolar geometry. Silpa-Anan and Hartley describe a similar system [Silpa-Anan and Hartley, 2004, 2005] which employs SIFT features. Zivkovic et al. [Zivkovic et al., 2005] consider the case where the order in which the images were collected is unknown, and use graph cuts to identify clusters of related images. A series of papers by Ho and Newman [Newman et al., 2006, Newman and Ho, 2005, Ho and Newman, 2005a,b] advance similarity matrix based techniques by considering the problem of perceptual aliasing. Their approach is based on a singular value decomposition of the similarity matrix that eliminates the effect of repetitive structure. They also address the issue of deciding if a putative match is indeed a loop closure based on examining an extreme value distribution [Gumbel, 1958] related to the similarity matrix.

This paper will describe a new appearance-based technique that improves on these existing results, particularly by addressing perceptual aliasing in a probabilistic framework. The core of our approach we have previously described in [Cummins and Newman, 2007]; here we expand on that presentation and present a detailed evaluation of the system's performance.

# 3 Approximating High Dimensional Discrete Distributions

This section introduces some background material that forms a core part of our system. Readers familiar with Chow Liu trees may wish to skip to Section 4.

Consider a distribution $P(Z)$ on $n$ discrete variables, $Z = \{z_1, z_2, \ldots, z_n\}$. We wish to learn the parameters of this distribution from a set of samples. If $P(Z)$ is a general distribution without special structure, the space needed to represent the distribution is exponential in $n$ — as $n$ increases dealing with such distributions quickly becomes intractable. A solution to this problem is to approximate $P(Z)$ by another distribution $Q(Z)$ possessing some special structure that makes it tractable to work with, and that is in some sense similar to the original distribution. The natural similarity metric in this context is the Kullback-Leibler divergence, defined as:

$$D_{KL}(P, Q) = \sum_{x \in X} P(x) log \frac{P(x)}{Q(x)} \tag{1}$$

where the summation is carried out over all possible states in the distribution. The KL divergence is zero when the distributions are identical, and strictly larger otherwise.

The Naive Bayes approximation is an example of a structural constraint that allows for tractable learning of $Q(Z)$, under which $Q(Z)$ is restricted such that each variable must be independent of all others. This extreme structural constraint limits the degree to which $Q(Z)$ can represent $P(Z)$. A less severe constraint is to require each variable in $Q(Z)$ to be conditioned on at most one other variable — this restricts the graphical model of $Q(Z)$ to be a tree. Unlike the Naive Bayes case, where there is only one possible graphical model, in this case there are $n^{n-2}$ possible tree-structured distributions

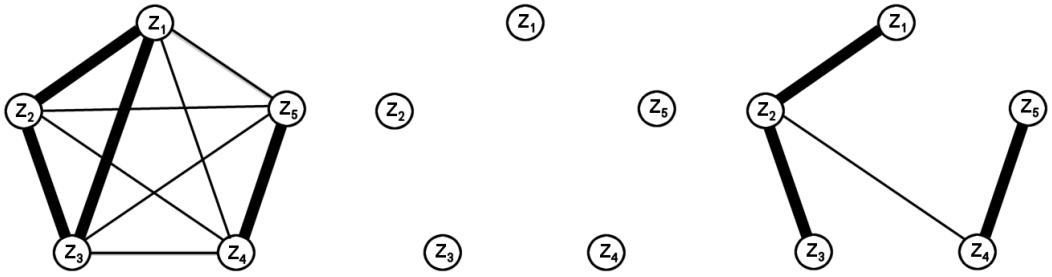



Figure 1: **(a)** Graphical model of the underlying distribution $P(Z)$. Mutual information between variables is shown by the thickness of the edge. **(b)** Naive Bayes approximation. **(c)** Chow Liu tree.

that satisfy the structural constraint. Chow and Liu [Chow and Liu, 1968] describe a polynomial time algorithm to select the best such distribution, which we elaborate on below.

More generally we could constrain $Q(Z)$ such that each variable is conditioned on at most $n$ other variables. This structure is known as a *thin junction tree*. It was shown in [Srebro, 2001] that the problem of determining the optimal thin junction tree is NP-hard, though there are several methods to find a good approximate solution (for example [Bach and Jordan, 2002]).

### 3.1 Chow Liu Trees

The Chow Liu algorithm approximates a discrete distribution $P(Z)$ by the closest tree-structured Bayesian network $Q(Z)_{opt}$, in the sense of minimizing the Kullback-Leibler divergence. The structure of $Q(Z)_{opt}$ is determined by considering a mutual information graph $\mathcal{G}$. For a distribution over $n$ variables, $\mathcal{G}$ is the complete graph with $n$ nodes and $\frac{n(n-1)}{2}$ edges, where each edge $(z_i, z_j)$ has weight equal to the mutual information $I(z_i, z_j)$ between variable $i$ and $j$:

$$I(z_i, z_j) = \sum_{z_i \in \Omega, z_j \in \Omega} p(z_i, z_j) log \frac{p(z_i, z_j)}{p(z_i)p(z_j)} \tag{2}$$

and the summation is carried out over all possible states of $z$. Mutual information measures the degree to which knowledge of the value of one variable predicts the value of another. It is zero if two variables are independent, and strictly larger otherwise.

Chow and Liu prove that the maximum-weight spanning tree [Cormen et al., 2001] of the mutual information graph will have the same structure as $Q(Z)_{opt}$ (see Figure 1). Intuitively, the dependencies between variables that have been approximated to independent have as little mutual information as possible, and so are the best ones to approximate as independent. Chow and Liu further prove that the conditional probabilities $p(z_i|z_j)$ for each edge in $Q(Z)_{opt}$ are the same as the conditional probabilities in $P(Z)$ – thus maximum likelihood estimates can be obtained directly from co-occurrence frequency in training data. To mitigate potential problems due to limited training data, we use the pseudo-Bayesian $p^*$ estimator described in [Bishop et al., 1977], rather than the maximum likelihood estimator. This prevents any probabilities from having unrealistic values of 0 or 1.

In the following section we will use Chow Liu trees to approximate large discrete distributions.

Because we are interested in learning distributions over a large number of variables (>10k) the mutual information graph to be computed is very large, typically too large to be stored in RAM. To deal with this, we use a semi-external spanning tree algorithm [Dementiev et al., 2004]. The mutual information graph is required only temporarily at learning time – it is discarded once the maximal spanning tree has been computed. Meilă [Meilă, 1999] has described a more efficient method of learning Chow Liu trees that avoids the explicit computation of the full mutual information graph, and provides significant speed-up for "sparse data" such as ours. However, as we need to compute Chow Liu trees only infrequently and offline, we have not yet implemented this approach.

We have chosen to use Chow Liu trees because they are tractable to compute even for very high-dimensional distributions, are guaranteed to be the optimal approximation within their model class, and require only first order conditional probabilities, which can be reliably estimated from available training data. However, our approach could easily substitute a more complex model such as a mixture of trees [Meilă-Predoviciu, 1999] or a thin junction tree of wider tree-width [Bach and Jordan, 2002], should this prove beneficial.

# 4 Probabilistic Navigation using Appearance

We now describe the construction of a probabilistic framework for appearance-based navigation. In overview, the world is modeled as a set of discrete locations, each location being described by a distribution over appearance words. Incoming sensory data is converted into a bag-of-words representation; for each location, we can then ask how likely it is that the observation came from that location's distribution. We also find an expression for the probability that the observation came from a place not in the map. This yields a PDF over location, which we can use to make a data association decision and update our belief about the appearance of each place. Essentially this is a SLAM algorithm in a discrete world. The method is outlined in detail in the following sections.

## 4.1 Representing Appearance

We adopt a "bag-of-words" representation of raw sensor data [Sivic and Zisserman, 2003], where scenes are represented as a collection of attributes (words) chosen from a set (vocabulary) of size $|v|$ . An observation of local scene appearance, visual or otherwise, captured at time $k$ is denoted $Z_k = \{z_1, \ldots, z_{|v|}\}$, where $z_i$ is a binary variable indicating the presence (or absence) of the $i^{th}$ word of the vocabulary. Furthermore $\mathcal{Z}^k$ is used to denote the set of all observations up to time $k$.

The results in this paper employ binary features derived from imagery, based on quantized SURF descriptors (see Section 5); however binary features from any sensor or combination of sensors could be used. For example, laser or sonar data could be represented using quantized versions of the features proposed in [Frome et al., 2004, Mozos et al., 2005, Johnson, 1997].

## 4.2 Representing Locations

At time $k$, our map of the environment is a collection of $n_k$ discrete and disjoint locations $\mathcal{L}^k = \{L_1, \ldots, L_{n_k}\}$. Each of these locations has an associated appearance model. Rather than model each

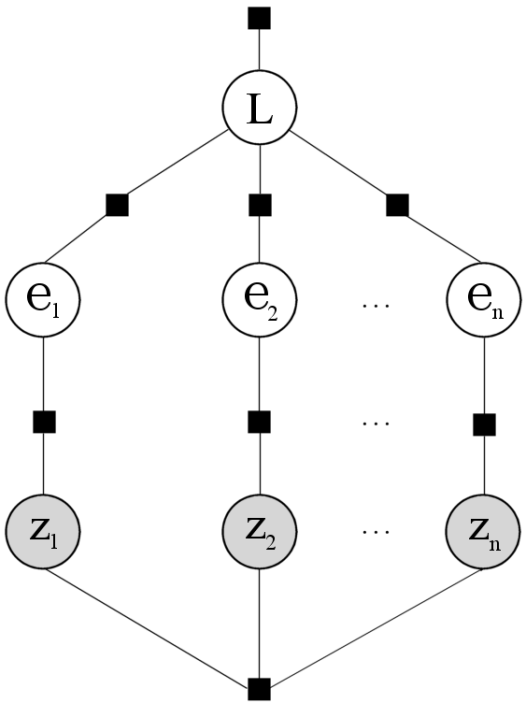



Figure 2: Factor graph of our generative model. Observed variables are shaded, latent variables unshaded. The environment generates locations $L_i$. Locations independently generate words $e_j$. Words generate observations $z_k$, which are interdependent.

of the locations directly in terms of which features are likely to be observed there (i.e. as $p(Z|L_j)$), we here introduce an extra hidden variable $e$. $e_i$ is the event that an object which generates observations of type $z_i$ exists. We model each location as a set $\{p(e_1 = 1|L_i), \ldots, p(e_{|v|} = 1|L_i)\}$, where each of the feature generating objects $e_i$ are generated independently by the location. A detector model relates feature existence $e_i$ to feature detection $z_i$. The detector is specified by

$$\mathcal{D}: \quad \begin{cases} p(z_i = 1|e_i = 0), & \text{false positive probability.} \\ p(z_i = 0|e_i = 1), & \text{false negative probability.} \end{cases} \tag{3}$$

The reason for introducing this extra hidden variable $e$ is twofold. Firstly, it provides a natural framework for incorporating data from multiple sensors with different (and possibly time-varying) error characteristics. Secondly, as outlined in the next section, it facilitates factoring the distribution $p(Z|L_j)$ into two parts — a simple model for each location composed of independent variables $e_i$, and a more complex model that captures the correlations between detections of appearance words $p(z_i|Z_k)$. Because the location model has a simple structure, it can be estimated online from the few available observations (combined with a prior). The more complex model that captures the correlations between observations is learned offline from abundant training data, and can be combined with the simple location models on the assumption that the conditional dependencies between appearance words are independent of location. We describe how this is achieved in the following section. The generative model is illustrated in Figure 2.

### 4.3 Estimating Location via Recursive Bayes

Calculating $p(L_i|\mathcal{Z}^k)$ can be formulated as a recursive Bayes estimation problem:

$$p(L_i|\mathcal{Z}^k) = \frac{p(Z_k|L_i, \mathcal{Z}^{k-1})p(L_i|\mathcal{Z}^{k-1})}{p(Z_k|\mathcal{Z}^{k-1})} \tag{4}$$

Here $p(L_i|\mathcal{Z}^{k-1})$ is our prior belief about our location, $p(Z_k|L_i, \mathcal{Z}^{k-1})$ is the observation likelihood, and $p(Z_k|\mathcal{Z}^{k-1})$ is a normalizing term. We discuss the evaluation of each of these terms below.

#### Observation Likelihood

To simplify evaluation of the observation likelihood $p(Z_k|L_i, \mathcal{Z}^{k-1})$, we first assume independence between the current and past observations, conditioned on the location. The simplified observation likelihood $p(Z_k|L_i)$ can then be expanded as:

$$p(Z_k|L_i) = p(z_n|z_1, z_2, ..., z_{n-1}, L_i)p(z_{n-1}|z_1, z_2, ..., z_{n-2}, L_i)...p(z_2|z_1, L_i)p(z_1|L_i) \tag{5}$$

This expression cannot be evaluated directly because of the intractability of learning the high-order conditional dependencies between appearance words. The simplest approximation is to make a Naive Bayes assumption, yielding:

$$p(Z_k|L_i) \approx p(z_n|L_i)...p(z_2|L_i)p(z_1|L_i) \tag{6}$$

Each factor can be further expanded as:

$$p(z_j|L_i) = \sum_{s\in\{0,1\}} p(z_j|e_j = s, L_i)p(e_j = s|L_i) \tag{7}$$

Applying the independence assumptions in our model ($p(z_j|e_j, L_i) = p(z_j|e_j)$, i.e. detector errors are independent of position in the world) this becomes:

$$p(z_j|L_i) = \sum_{s\in\{0,1\}} p(z_j|e_j = s)p(e_j = s|L_i) \tag{8}$$

which can now be evaluated directly — $p(z_j|e_j = s)$ are the components of the detector model specified in Equation 3 and $p(e_j = s|L_i)$ is a component of the place appearance model.

While a Naive Bayes approach yields reasonable results (see Section 5), substantially better results can be obtained by instead employing a Chow-Liu approximation to capture more of the dependencies between appearance words. Using a Chow Liu assumption, the appearance likelihood becomes

$$p(Z_k|L_i) \approx p(z_r|L_i) \prod_{q=2}^{|v|} p(z_q|z_{p_q}, L_i) \tag{9}$$

where $z_r$ is the root of the tree and $z_{p_q}$ is the parent of $z_q$ in the tree. The observation factors in

Equation 9 can be further expanded as:

$$p(z_q|z_{p_q}, L_i) = \sum_{s_{e_q} \in \{0,1\}} p(z_q|e_q = s_{e_q}, z_{p_q}, L_i) p(e_q = s_{e_q}|z_{p_q}, L_i) \tag{10}$$

Applying our independence assumptions and making the approximation that $p(e_j)$ is independent of $z_i$ for all $i \neq j$ , the expression becomes:

$$p(z_q|z_{p_q}, L_i) = \sum_{s_{e_q} \in \{0,1\}} p(z_q|e_q = s_{e_q}, z_{p_q}) p(e_q = s_{e_q}|L_i) \tag{11}$$

The term $p(z_q|e_q, z_{p_q})$ can be expanded (see Appendix B) as:

$$p(z_q = s_{z_q}|e_q = s_{e_q}, z_p = s_{z_p}) = (1 + \frac{\alpha}{\beta})^{-1} \tag{12}$$

where $s_{z_q}, s_{e_q}, s_{z_p} \in \{0, 1\}$ and

$$\alpha = p(z_q = s_{z_q}) p(z_q = \overline{s_{z_q}}|e_q = s_{e_q}) p(z_q = \overline{s_{z_q}}|z_p = s_{z_p}) \tag{13}$$

$$\beta = \underbrace{p(z_q = \overline{s_{z_q}})}_{prior} \underbrace{p(z_q = s_{z_q}|e_q = s_{e_q})}_{detector\ model} \underbrace{p(z_q = s_{z_q}|z_p = s_{z_p})}_{conditional\ from\ training} \tag{14}$$

where $\overline{s_z}$ denotes the opposite state to $s_z$. $\alpha$ and $\beta$ are now expressed entirely in terms of quantities which can be estimated from training data (as indicated by the under-braces). Hence $p(Z_k|L_i)$ can now be computed directly.

**New Place or Old Place?**

We now turn our attention to the denominator of Equation 4, $p(Z_k|\mathcal{Z}^{k-1})$. For pure localization we could compute a PDF over location simply by normalizing the observation likelihoods computed as described in the previous section. However, we would like to be able to deal with the possibility that a new observation comes from a previously unknown location, which requires an explicit calculation of $p(Z_k|\mathcal{Z}^{k-1})$.

If we divide the world into the set of mapped places $M$ and the unmapped places $\overline{M}$, then

$$p(Z_k|\mathcal{Z}^{k-1}) = \sum_{m \in M} p(Z_k|L_m) p(L_m|\mathcal{Z}^{k-1}) + \sum_{u \in \overline{M}} p(Z_k|L_u) p(L_u|\mathcal{Z}^{k-1}) \tag{15}$$

where we have applied our assumption that observations are conditionally independent given location. The second summation cannot be evaluated directly because it involves all possible unknown places. We have compared two different approximations to this term. The first is a mean field approximation [Jordan et al., 1999] :

$$p(Z_k|\mathcal{Z}^{k-1}) \approx \sum_{m \in M} p(Z_k|L_m) p(L_m|\mathcal{Z}^{k-1}) + p(Z_k|L_{avg}) \sum_{u \in \overline{M}} p(L_u|\mathcal{Z}^{k-1}) \tag{16}$$

where $L_{avg}$ is the "average place" where the $e_i$ values are set to their marginal probability, and

$\sum_{u \in \overline{M}} p(L_u|\mathcal{Z}^{k-1})$ is the prior probability that we are at an unknown place, which can be obtained from our previous position estimate and a motion model (discussed in the next section).

An alternative to the mean field approach is to approximate the second summation via sampling. The procedure is to sample location models $L_u$ according to the distribution by which they are generated by the environment, and evaluate $\sum_{u \in \overline{M}} p(Z_k|L_u)p(L_u|\mathcal{Z}^{k-1})$ for the sampled location models. In order to do this, some method of sampling location models $L_u$ is required. Here we make an approximation for ease of implementation — we instead sample an observation $Z$ and use the sampled observation to create a place model. The reason for doing this is that sampling an observation $Z$ is extremely easy — our sampler simply consists of selecting at random from a large collection of observations. For our robot which navigates in outdoor urban environments using imagery, large collections of street-side imagery for sampling are readily available – for example from previous runs of the robot, or from such sources as Google Street View[1]. In general this sampling procedure will not create location models according to their true distribution because models may have been created from multiple observations of the location. However, it will be a good approximation when the robot is exploring a new environment, where most location models will have only a single observation associated with them. In practice we have found that it works very well, and it has the advantage of being very easy to implement.

Having sampled a location model, we must evaluate $p(Z_k|L_u)p(L_u|\mathcal{Z}^{k-1})$ for the sample. Calculating $p(Z_k|L_u)$ has already been described, however we currently have no method of evaluating $p(L_u|\mathcal{Z}^{k-1})$, the prior probability of our sampled place model with respect to our history of observations, so we assume it to be uniform over the samples. Equation 15 thus becomes

$$p(Z_k|\mathcal{Z}^{k-1}) \approx \sum_{m \in M} p(Z_k|L_m)p(L_m|\mathcal{Z}^{k-1}) + p(L_{new}|\mathcal{Z}^{k-1}) \sum_{u=1}^{n_s} \frac{p(Z_k|L_u)}{n_s} \tag{17}$$

where $n_s$ is the number of samples used, and $p(L_{new}|\mathcal{Z}^{k-1})$ is our prior probability of being at a new place.

### Location Prior

The last term to discuss is the location prior $p(L_i|\mathcal{Z}^{k-1})$. The most straight forward way to obtain a prior is to transform the previous position estimate via a motion model, and use this as our prior for location at the next time step. While at present our system is not explicitly building a topological map, for the purpose of calculating a prior we assume that sequentially collected observations come from adjacent places, so that if the robot is at place $i$ at time $t$, it is likely to be at one of the places $\{i-1, i, i+1\}$ at time $t+1$, with equal probability. For places with unknown neighbours (e.g. the next place after the most recently collected observation, where place $i+1$ may be a new place not already in the map), part of the probability mass is assigned to a "new place" node and the remainder is spread evenly over all places in the map. The split is governed by the probability that a topological link with unknown endpoint leads to a new place, which is a user-defined parameter in our system. Clearly this prior could be improved via a better topology estimate or through the use of some odometry information — however, we have found that the effect of the prior is relatively weak, so these issues

[1] http://maps.google.com/help/maps/streetview/

are not crucial to performance.

If the assumption that sequential observations come from adjacent places is not valid, the prior can be simply left uniform. While this results in some increase in the number of false matches, performance is largely unaffected. Alternatively, if we have an estimate for the topology of the environment, but no estimate of our position within this topology, we could obtain a prior as the limit of a random walk on the topology. This can be obtained efficiently as the dominant eigenvector of the normalized link matrix of the topology [Page et al., 1998].

### Smoothing

We have found that the performance of the inference procedure outlined above is strongly dependent on an accurate estimation of the denominator term in Equation 4, $p(Z_k|\mathcal{Z}^{k-1})$. Ideally we would like to perform the Monte Carlo integration in Equation 17 over a large set of place models $L_u$ that fully capture the visual variety of the environment. In practice, we are limited by running time and available data. The consequence is that occasionally we will incorrectly assign two similar images high probability of having come from the same place, when in fact the similarity is due to perceptual aliasing.

An example of this is illustrated in Figure 13, where two images from different places, both of which contain iron railings, are incorrectly assigned high probability of having come from the same place. The reason for this mistake is that the railings generate a large number of features, and there are no examples of railings in the sampling set used to calculate $p(Z_k|\mathcal{Z}^{k-1})$, so the system cannot know these sets of features are correlated.

In practice, due to limited data and computation time, there will always be some aspects of repetitive environmental structure that we fail to capture. To ameliorate this problem, we apply a smoothing operation to our likelihood estimates:

$$p(Z_k|L_i) \longrightarrow \sigma p(Z_k|L_i) + \frac{(1-\sigma)}{n_k} \tag{18}$$

where $n_k$ is the number of places in the map and $\sigma$ is the smoothing parameter, which for our experiments was set to 0.99.

The effect of this very slight smoothing of the data likelihood values is to prevent the system asserting loop closure with high confidence on the basis of a single similar image pair. Smoothing the likelihood values effectively boosts the importance of the prior term $p(L_i|\mathcal{Z}^{k-1})$, so that high probability of loop closure can now only be achieved by accumulating evidence from a sequence of matched observations. While in principle mistakes due to our approximate calculation of $p(Z_k|\mathcal{Z}^{k-1})$ can still occur, in practice we have found that this modification successfully rejects almost all outliers, while still allowing true loop closures to achieve high probability after a sequence of only two or three corresponding images.

### Updating Place Models

We have now outlined how to compute a PDF over location given a new observation. The final issue to address is data association and the updating of place appearance models. At present we simply

make a maximum likelihood data association decision after each observation. Clearly there is room for improvement here, for example by maintaining a PDF over topologies using a particle filter, as described in [Ranganathan and Dellaert, 2006].

After taking a data association decision, we can update the relevant place appearance model, which consists of the set $\{p(e_1 = 1|L_j), \ldots, p(e_{|v|} = 1|L_j)\}$. When a new place is created, its appearance model is initialized so that all words exist with marginal probability $p(e_i = 1)$ derived from the training data. Given an observation that relates to the place, each component of the appearance model can be updated according to:

$$p(e_i = 1|L_j, \mathcal{Z}^k) = \frac{p(Z_k|e_i = 1, L_j)p(e_i = 1|L_j, \mathcal{Z}^{k-1})}{p(Z_k|L_j)} \tag{19}$$

where we have applied our assumption that observations are conditionally independent given location.

### 4.4 Summary of Approximations and Parameters

We briefly recap the approximations and user-defined terms in our probabilistic scheme. For tractability, we have made a number of independence assumptions:

1. Sets of observations are conditionally independent given position: $p(Z_k|L_i, \mathcal{Z}^{k-1}) = p(Z_k|L_i)$.
2. Detector behavior is independent of position: $p(z_j|e_j, L_i) = p(z_j|e_j)$.
3. Location models are generated independently by the environment.

In addition to these independence assumptions, we also make an approximation for tractability:

1. Observations of one feature do not inform us about the existence of other features: $p(e_j|Z_k) \approx p(e_j|z_{j_k})$. While more complex inference is possible here, in effect this approximation only means that we will be somewhat less certain about feature existence than if we made full use of the available data.

#### Input Parameters

The only user-specified inputs to the algorithm are the detector model, $p(z_i = 1|e_i = 0)$ and $p(z_i = 0|e_i = 1)$ (two scalars), the smoothing parameter $\sigma$ and a term that sets the prior probability that a topological link with an unknown endpoint leads to a new place. Of these, the algorithm is particularly sensitive only to the detector model, which can be determined from a calibration discussed in the following section.

## 5 Evaluation

We tested the described algorithm using imagery from a mobile robot. Each image that the robot collects is passed into a processing pipeline that produces a bag-of-words representation, which is then the input to the algorithm described. There are many possible ways to convert input sensory data into a bag-of-words representation. Our system uses the SURF detector/descriptor [Bay et al., 2006]

to extract regions of interest from the images, and compute 128D non-rotation-invariant descriptors for these regions. Finally, we map regions of descriptor space to visual words as suggested in [Sivic and Zisserman, 2003]. This is achieved by clustering all the descriptors from a set of training images using a simple incremental clustering procedure, then quantizing each descriptor in the test images to its approximate nearest cluster centre using a kd-tree. We have not focused on this bag-of-words generation stage of the system – more sophisticated approaches exist [Nister and Stewenius, 2006, Schindler et al., 2007] and there are probably gains to be realized here.

## 5.1 Building the Vocabulary Model

The next step is to construct a Chow Liu tree to capture the co-occurrence statistics of the visual words. To do this, we construct the mutual information graph as described in Section 3.1. Each node in the graph corresponds to a visual word, and the edge weights (mutual information) between node $i$ and $j$ are calculated as per Equation 2 - essentially this amounts to counting the number of images in which word $i$ and $j$ co-occur. The Chow Liu tree is then the maximum weight spanning tree of the graph.

If the Chow Liu tree is to be a good approximation to the true distribution, it must be computed from a large number of observations. To prevent bias, these observations should be independent samples from the observation distribution. We collected 2,800 images from 28km of urban streets using the robot's camera. The images were taken 10m apart, perpendicular to the robot's motion, so that they are non-overlapping and approximate independent samples from the observation distribution. From this dataset we compute the clustering of SURF features that form our vocabulary, then compute the Chow Liu tree for the vocabulary. The clustering procedure generated a vocabulary of approximately 11k words. The combined process takes 2 hours on a 3GHz Pentium IV. This is a one-off computation that occurs offline.

Illustrative results are shown in Figures 3, 4 and 5, which show sample visual words learned by the system and a visualization of a portion of the Chow Liu tree. These words (which correspond to parts of windows) were determined to have high pairwise mutual information, and so are neighbours in the tree. The joint probability of observing the five words shown in the tree is 4,778 times higher under the Chow Liu model than under a Naive Bayes assumption. Effectively the system is learning about the presence of objects that generate the visual words. This yields a significant improvement in inference performance (see Figure 12).

## 5.2 Testing Conditions

We used this vocabulary to navigate using imagery collected by a mobile robot. We modeled our detector by $p(z_i = 0|e_i = 1) = 0.39$ and $p(z_i = 1|e_i = 0) = 0$ – i.e. a per-word false negative rate of 39%, and no false positives[2]. In principle different detector terms could be learned for each word – for example words generated by cars might be less reliably reobservable than words generated by buildings.

[2]To set these parameters, we first assumed a false-positive rate of zero. The false negative rate could then be determined by collecting multiple images at a test location – $p(z_i = 0|e_i = 1)$ then comes directly from the ratio of the number of words detected in any one image to the number of unique words detected in union of all the images from that location.

Figure 3: A sample word in the vocabulary, showing typical image patches and an example of the interest points in context. Interest points quantized to this word typically correspond to the top-left corner of windows. The most correlated word in the vocabulary is shown in Figure 4.

Figure 4: A sample word in the vocabulary, showing typical image patches and an example of the interest points in context. Interest points quantized to this word typically correspond to the cross-piece of windows. The most correlated word in the vocabulary is shown in Figure 3.

Figure 5: Visualization of a section the Chow Liu tree computed for our urban vocabulary. Each word in the tree is represented by a typical example. Clockwise from top, the words correspond to the cross-pieces of window panes, right corners of window stills, top-right corners of window panes, bottom-right corners of window panes and top-left corners of window panes. Under the Chow Liu model the joint probability of observing these words together is 4,778 times higher than under the a Naive Bayes assumption.

At present we assume all detector terms are the same. The number of samples for determining the denominator term $p(Z_k|\mathcal{Z}^{k-1})$ was fixed at 2,800 – the set of images used to learn the vocabulary. The smoothing parameter $\sigma$ was 0.99, and the prior that a new topological link leads to a new place was 0.9. To evaluate the effect of the various alternative approximations discussed in Section 4, several sets of results were generated, comparing Naive Bayes vs. Chow Liu approximations to $p(Z|L_i)$ and Mean Field vs. Monte Carlo approximations to $p(Z_k|\mathcal{Z}^{k-1})$.

### 5.3 Results

We tested the system on two outdoor urban datasets collected by our mobile robot. Both datasets are included in Extension 3. As the robot travels through the environment, it collects images to the left and right of its trajectory approximately every 1.5m. Each collected image is processed by our algorithm and is used either to initialize a new place, or, if loop closure is detected, to update an existing place model.

The area of our test datasets did not overlap with the region where the training data was collected. The first dataset – New College – was chosen to test the system's robustness to perceptual aliasing. It features several large areas of strong visual repetition, including a medieval cloister with identical repeating archways and a garden area with a long stretch of uniform stone wall and bushes. The second dataset – labeled City Centre – was chosen to test matching ability in the presence of scene change. It was collected along public roads near the city centre, and features many dynamic objects such as traffic and pedestrians. Additionally it was collected on a windy day with bright sunshine, which makes the abundant foliage and shadow features unstable.

Figures 6 and 7 show navigation results overlaid on an aerial photo. These results were generated using the Chow Liu and Monte Carlo approximations, which we found to give the best performance. The system correctly identifies a large proportion of possible loop closures with high confidence. There are no false positives that meet the probability threshold. See also Extension 1 and Extension 2 for videos of these results.

Precision recall curves are shown in Figure 8. The curves were generated by varying the probability at which a loop closure was accepted. Ground truth was labeled by hand. We are primarily interested in the recall rate at 100% precision – if the system were to be used to complement a metric SLAM algorithm, an incorrect loop closure could cause unrecoverable mapping failure. At 100% precision, the system achieves 48% recall on the New College dataset, and 37% on the more challenging City Centre dataset. "Recall" here is the proportion of possible image-to-image matches that exceed the probability threshold. As a typical loop closure is composed of multiple images, even a recall rate of 37% is sufficient to detect almost all loop closures.

Some examples of typical image matching results are presented in Figures 9 and 10. Figure 9 highlights robustness to perceptual aliasing. Here very similar images that originate from different locations are correctly assigned low probability of having come from the same place. We emphasize that these results are not outliers; they show typical system performance. Figure 10 shows matching performance in the presence of scene change. Many of these image pairs have far fewer visual words in common than the examples of perceptual aliasing, yet are assigned high probability of having come from the same place. The system can reliably reject perceptual aliasing, even when images have as

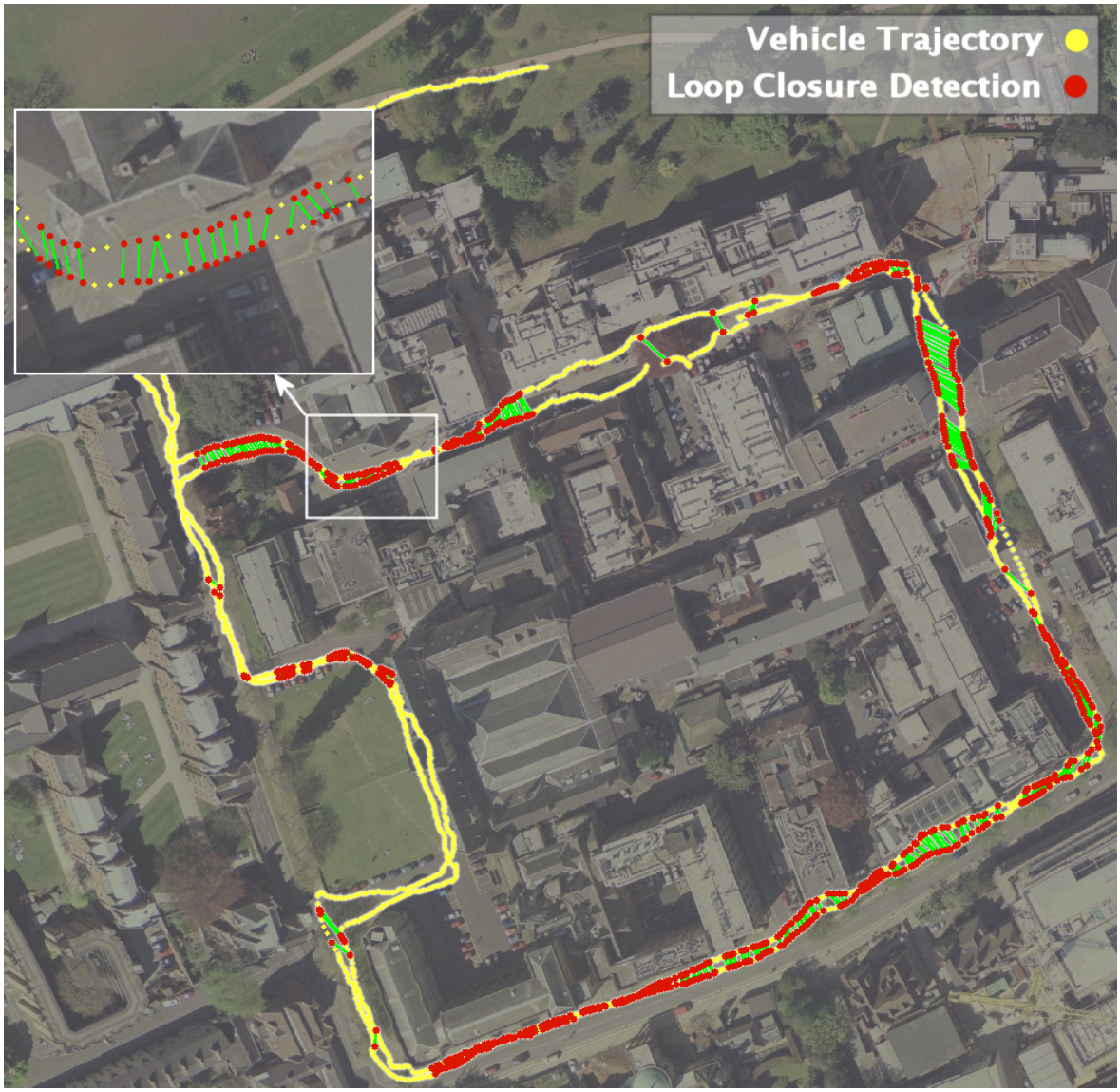



Figure 6: Appearance-based matching results for the City Centre dataset overlaid on an aerial photograph. The robot travels twice around a loop with total path length 2km, collecting 2,474 images. Positions (from hand-corrected GPS) at which the robot collected an image are marked with a yellow dot. Two images that were assigned a probability $p \geq 0.99$ of having come from the same location (on the basis of appearance alone) are marked in red and joined with a line. There are no incorrect matches that meet this probability threshold. This result is best viewed as a video (Extension 2).


Figure 7: Appearance-based matching results for the New College dataset overlaid on an aerial photograph. The robot traverses a complex trajectory of 1.9km with multiple loop closures. 2,146 images were collected. Positions (from hand-corrected GPS) at which the robot collected an image are marked with a yellow dot. Two images that were assigned a probability $p \geq 0.99$ of having come from the same location (on the basis of appearance alone) are marked in red and joined with a line. There are no incorrect matches that meet this probability threshold. This result is best viewed as a video (Extension 1).

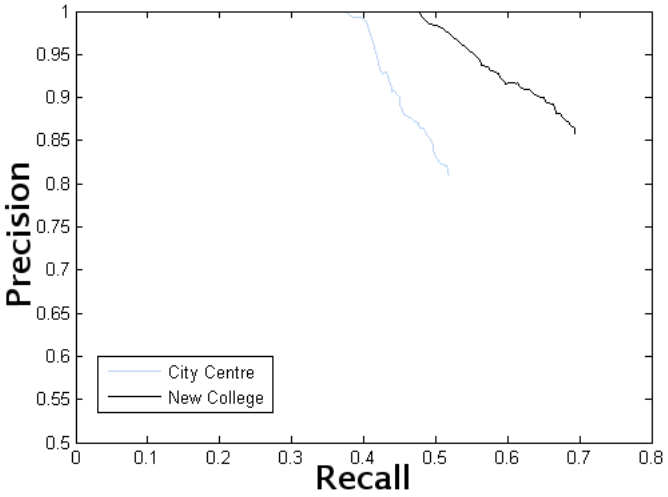



Figure 8: Precision-Recall curves for the City Centre and New College datasets. Notice the scale.

(a) p=4.6x10$^{-10}$ (b) p=3x10$^{-9}$ (c) p=0.74

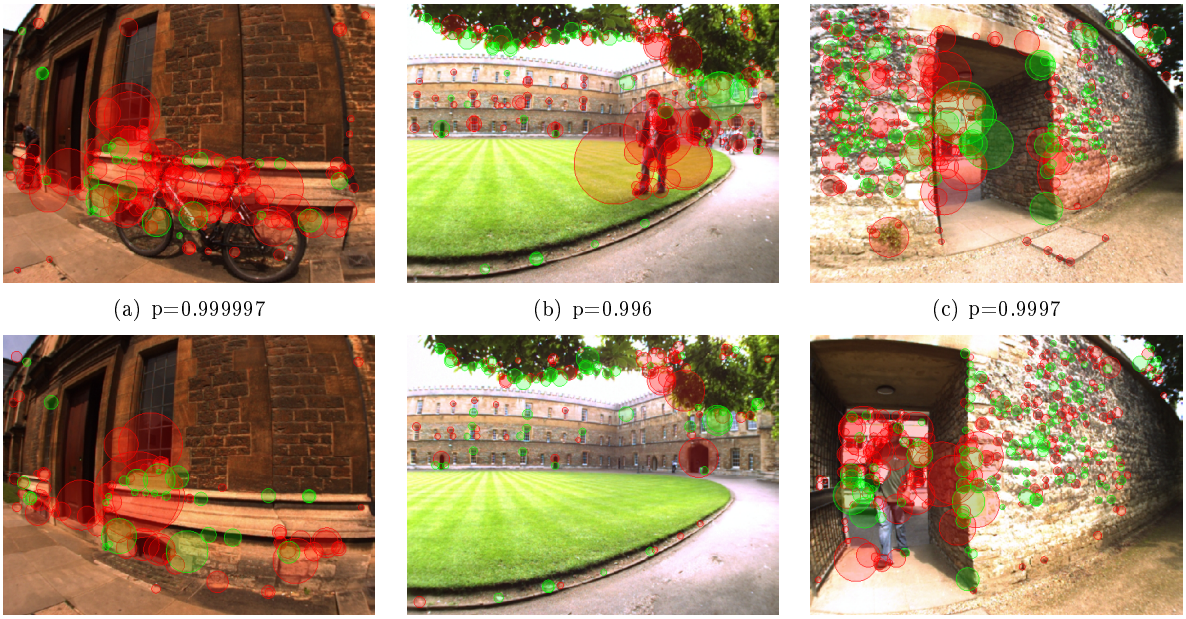
Figure 9: Some examples of remarkably similar-looking images from different parts of the workspace that were correctly assigned low probability of having come from the same place. The result is possible because most of the probability mass is captured by locations in the sampling set – effectively the system has learned that images like these are common. Of course, had these examples been genuine loop closures they might also have received low probability. We would argue that this is correct behaviour, modulo the fact that the probabilities in (a) and (b) are too low. The very low probabilities in (a) and (b) are due to the fact that good matches for the query images are found in the sampling set, capturing almost all the probability mass. This is less likely in the case of a true but ambiguous loop closure. Words common to both images are shown in green, others in red. (Common words are shown in blue in (b) for better contrast). The probability that the two images come from the same place is indicated between the pairs.

much as 46% of visual words in common (e.g. Figure 9(b)), while detecting loop closures where images have as few as 8% of words in common. Poor probability estimates do occasionally occur – some examples of images incorrectly assigned high probability of a place match are shown in Figure 13. Note however that the typical true loop closure receives a much higher probability of a match. Neither of the examples in Figure 13 met the data association threshold of 0.99.

### Comparing Approximations

This section presents a comparison of the alternative approximations discussed in Section 4 — Naive Bayes vs. Chow Liu approximations to $p(Z|L_i)$ and Mean Field vs. Monte Carlo approximations to $p(Z_k|\mathcal{Z}^{k-1})$. Figure 11 shows precision-recall curves from the New College dataset for the four possible combinations. Timing and accuracy results are summarized in Table 1. The Chow Liu and Monte Carlo approximations both considerably improve performance, particularly at high levels of precision, which is the region that concerns us most. This is not surprising as our variables display a high level of correlation. Some examples of typical loop closures detected by the Chow Liu approximation but not by the Naive Bayes are shown in Figure 12.

The extra performance comes at the cost of some increase in computation time; however even the



(a) p=0.999997 (b) p=0.996 (c) p=0.9997

Figure 10: Some examples of images that were assigned high probability of having come from the same place, despite scene change. Words common to both images are shown in green, others in red. The probability that the two images come from the same place is indicated between the pairs.

| Algorithm | Recall - New College | Recall - City Centre | Run Time |
|---|---|---|---|
| Mean Field, Naive Bayes | 34% | 16% | 0.6ms/place |
| Mean Field, Chow Liu | 35% | 31% | 1.1ms/place |
| Monte Carlo, Naive Bayes | 40% | 31% | 0.6ms/place + 1.71 secs sampling |
| Monte Carlo, Chow Liu | 48% | 37% | 1.1ms/place + 3.15 secs sampling |

Table 1: Comparison of the four different approximations. The recall rates quoted are at 100% precision. The time to process a new observation is given as a function of the number of places already in the map, plus a fixed cost to perform the sampling. Timing results are for a 3GHZ Pentium IV.

slowest version using Chow Liu and Monte Carlo approximations is still relatively fast. Running times are summarized in Table 1. The maximum time taken to process a new observation over all datasets was 5.9 seconds. As the robot collects images approximately every two seconds, this is not too far from real time performance. The dominant computational cost is the calculation of $p(Z|L_i)$ for each place model in the map and each sampled place. Each of these calculations is independent, so the algorithm is highly parallelizable and will perform well on multi-core processors.

### Generalization Performance

Because FAB-MAP relies on a learned appearance model to reject perceptual aliasing and improve inference, a natural question is to what extent system performance will degrade in environments very dissimilar to the training set. To investigate this question we have recently performed some preliminary indoor navigation experiments with data from a hand-held video camera, using the same training data and algorithm parameters as used outdoors. The quality of the imagery in this indoor dataset is

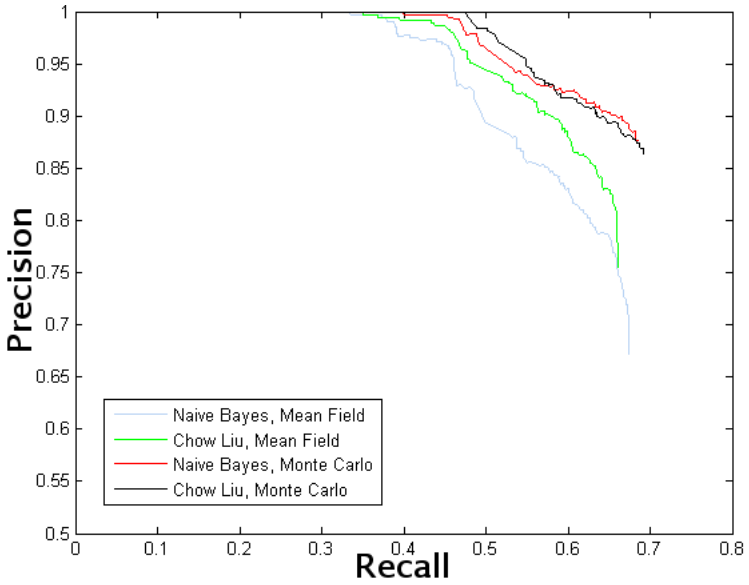
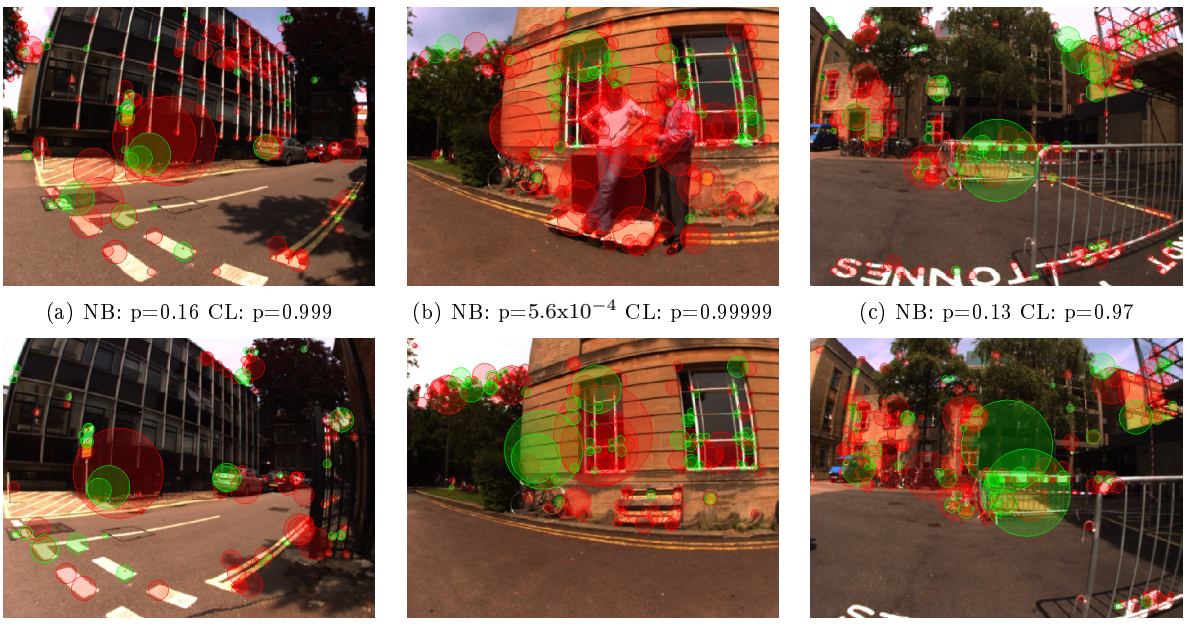



Figure 11: Precision-Recall curves for the four variant algorithms on the New College datasets. Notice the scale. Relative performance on the City Centre dataset is comparable.

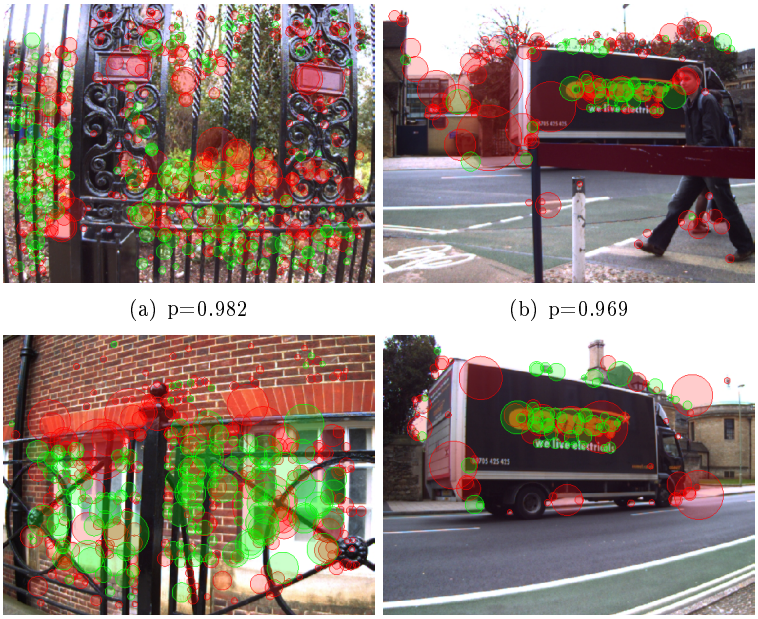


(a) NB: p=0.16 CL: p=0.999

(b) NB: p=5.6x10$^{-4}$ CL: p=0.99999

(c) NB: p=0.13 CL: p=0.97

Figure 12: Some examples of situations where the Chow Liu approximation outperforms Naive Bayes. In (a), a change in lighting means that the feature detector does not fire on the windows of the building. In (b), the people are no longer present. In (c), the foreground text and the scaffolding in the top right are not present in the second image. In each of these cases, the missing features are known to be correlated by the Chow Liu approximation, hence the more accurate probability. Words common to both images are shown in green, others in red. The probability that the two images come from the same place (according to both the Chow Liu and Naive Bayes models) is indicated between the pairs.



(a) p=0.982 (b) p=0.969

Figure 13: Some images from different locations incorrectly assigned high probability of having come from the same place. In (a), the training set contains no examples of railings, so the matched features are not known to be correlated. In (b), we encounter the same truck again in a different part of the workspace. Errors of this type are particularly challenging. Notice that while both images are assigned high probability of a match, a typical true loop closure is assigned much higher probability. Neither of these image pairs met our $p = 0.99$ data association threshold.

considerably poorer than that from the robot, due to adverse lighting conditions and greater motion blur. Interestingly, however, system performance is largely similar to our outdoor results (46% recall at 100% precision). The fact that the system works indoors despite using outdoor training data is quite surprising. Most interestingly, the Chow Liu approximation continues to noticeably outperform the Naive Bayes on the indoor dataset (46% recall as opposed to 39%). This suggests that some of the correlations being learned by the Chow Liu tree model generic low-level aspects of imagery. These results, while preliminary, indicate that the system will degrade gracefully in environments dissimilar to the training set. See (Extension 4 and 5) for video results.

# 6 Conclusions

This paper has introduced the FAB-MAP algorithm, a probabilistic framework for navigation and mapping which relies on appearance information only. The system is robust even in visually repetitive environments and is fast enough for online loop closure detection in realistic mobile robotics applications. In our evaluation, FAB-MAP was successful in detecting large portions of loop closures in challenging outdoor environments, without false positives.

Two aspects of our results are particularly noteworthy. Firstly, learning a generative model of our bag-of-words observations yielded a marked improvement in performance. We think that this technique will have applications beyond the specific problem addressed in this paper. Secondly, we have observed that the system appears to perform well even in environments quite dissimilar to the training data.

This suggests that the approach is practical for deployment in unconstrained navigation tasks, as a natural compliment to more typical metric SLAM algorithms.

## Acknowledgments

The work reported in this paper was funded by the Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre established by the UK Ministry of Defence and by the Engineering and Physical Sciences Research Council.

## References

Francis R. Bach and Michael I. Jordan. Thin junction trees. In *Advances in Neural Information Processing Systems*, 2002.

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded Up Robust Features. In *Proc 9th European Conf on Computer Vision*, volume 13, pages 404–417, Graz, Austria, May 7 2006.

Yvonne M. Bishop, Stephen E. Fienberg, and Paul W. Holland. *Discrete Multivariate Analysis: Theory and Practice*. The MIT Press, June 1977. ISBN 0262520400.

Michael Bowling, Dana Wilkinson, Ali Ghodsi, and Adam Milstein. Subjective localization with action respecting embedding. In *Proceedings of the International Symposium of Robotics Research*, 2005.

Michael Bowling, Dana Wilkinson, and Ali Ghodsi. Subjective mapping. In *Twenty-First National Conference on Artificial Intelligence*, Boston, Massachusetts, USA, July 2006. AAAI Press.

Cheng Chen and Han Wang. Appearance-based topological Bayesian inference for loop-closing detection in a cross-country environment. *The International Journal of Robotics Research*, 25(10): 953–983, 2006.

C.K. Chow and C.N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, IT-14(3), May 1968.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Second Edition*. The MIT Press, September 2001. ISBN 0262531968. URL `http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20&path=ASIN/0262531968`.

Mark Cummins and Paul Newman. Probabilistic appearance based navigation and loop closing. In *Proc. IEEE International Conference on Robotics and Automation (ICRA'07)*, Rome, April 2007.

Roman Dementiev, Peter Sanders, Dominik Schultes, and Jop Sibeyn. Engineering an external memory minimum spanning tree algorithm. In *Proceedings 3rd International Conference on Theoretical Computer Science*, 2004.

F. Ferreira, V. Santos, and J. Dias. Integration of Multiple Sensors using Binary Features in a Bernoulli Mixture Model. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 104–109, 2006.

David Filliat. A visual bag of words method for interactive qualitative localization and mapping. In *IEEE International Conference on Robotics and Automation*, pages 3921–3926, 2007.

Andrea Frome, Daniel Huber, Ravi Kolluri, Thomas Bulow, and Jitendra Malik. Recognizing objects in range data using regional point descriptors. In *Proc. European Conf. on Computer Vision*. Springer, 2004.

Toon Goedemé. *Visual navigation*. PhD thesis, Katholieke Universiteit Leuven, December 2006.

E. J. Gumbel. *Statistics of Extremes*. Columbia University Press, New York, NY, 1958.

J. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pages 318–325, Monterey, California, November 1999. URL `http://citeseer.ist.psu.edu/gutmann00incremental.html`.

K. Ho and P. Newman. Multiple map intersection detection using visual appearance. In *International Conference on Computational Intelligence, Robotics and Autonomous Systems*, Singapore, December 2005a.

K. Ho and P. Newman. Combining visual and spatial appearance for loop closure detection. *Proceedings of European Conference on Mobile Robotics*, September 2005b.

Kin Leong Ho and Paul Newman. Detecting loop closure with scene sequences. *International Journal of Computer Vision*, 74(3):261–286, September 2007.

Andrew Johnson. *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 1997.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, 1999. ISSN 0885-6125. doi: 10.1023/A:1007665907178. URL `http://portal.acm.org/citation.cfm?id=339248.339252`.

Ben J. A. Kröse, Nikos A. Vlassis, Roland Bunschoten, and Yoichi Motomura. A probabilistic model for appearance-based robot localization. *Image and Vision Computing*, 19(6):381–391, 2001.

Pierre Lamon, Illah Nourbakhsh, Björn Jensen, and Roland Siegwart. Deriving and matching image fingerprint sequences for mobile robot localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 21-26 2001.

Anat Levin and Richard Szeliski. Visual odometry and map correlation. *IEEE Conference on Computer Vision and Pattern Recognition*, 01:611–618, 2004. ISSN 1063-6919. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2004.266.

Fayin Li and Jana Kosecka. Probabilistic location recognition using reduced feature set. In *IEEE International Conference on Robotics and Automation*, Orlando, Florida, 2006.

David G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the 7th International Conference on Computer Vision*, pages 1150–1157, Kerkyra, 1999.

Marina Meilă. An accelerated Chow and Liu algorithm: Fitting tree distributions to high-dimensional sparse data. In *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, pages 249–257, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1-55860-612-2.

Marina Meilă-Predoviciu. *Learning with Mixtures of Trees.* PhD thesis, Massachusetts Institute of Technology, January 1999.

OM Mozos, C. Stachniss, and W. Burgard. Supervised Learning of Places from Range Data using AdaBoost. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1730–1735, 2005.

P. Newman and K. Ho. SLAM - Loop Closing with Visually Salient Features. *IEEE International Conference on Robotics and Automation*, 18-22 April 2005.

P. M. Newman, D. M. Cole, and K. Ho. Outdoor SLAM using visual appearance and laser ranging. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Orlando Florida USA, May 2006.

David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Conf. Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, 2006.

Larry Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies, 1998.

Fabio T. Ramos, Ben Upcroft, Suresh Kumar, and Hugh F. Durrant-Whyte. A Bayesian approach for place recognition. In *IJCAI Workshop on reasoning with Uncertainty in Robotics*, July 2005.

Ananth Ranganathan and Frank Dellaert. A Rao-Blackwellized particle filter for topological mapping. In *Proceedings of International Conference on Robotics and Automation*, Orlando, Florida, USA, May 2006. URL `http://www.cc.gatech.edu/ dellaert/pub/Ranganathan06icra.pdf`.

Grant Schindler, Matthew Brown, and Richard Szeliski. City-Scale Location Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007.

Stephen Se, David G. Lowe, and James Little. Vision based global localisation and mapping for mobile robots. *IEEE Transactions on Robotics*, 21(3):364–375, June 2005.

Chanop Silpa-Anan and Richard Hartley. Localisation using an image-map. In *Australian Conference on Robotics and Automation*, 2004.

Chanop Silpa-Anan and Richard Hartley. Visual localization and loop-back detection with a high resolution omnidirectional camera. In *Workshop on Omnidirectional Vision*, 2005.

Robert Sim and Gregory Dudek. Mobile robot localization from learned landmarks. *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2, 1998.

J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, Nice, France, October 2003.

D.M.G. Squire, W. Müller, H. Müller, and T. Pun. Content-based query of image databases: inspirations from text retrieval. *Pattern Recognition Letters*, 21(13-14):1193–1198, 2000.

Nathan Srebro. Maximum likelihood bounded tree-width markov networks. In *Proceedings of the 17th Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)*, pages 504–51, San Francisco, CA, 2001. Morgan Kaufmann.

Antonio Torralba, Kevin P. Murphy, William T. Freeman, and Mark A. Rubin. Context-based vision system for place and object recognition. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 273, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1950-4.

Iwan Ulrich and Illah Nourbakhsh. Appearance-based place recognition for topological localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1023 – 1029, April 2000.

Junqiu Wang, Hongbin Zha, and Roberto Cipolla. Vision-based global localization using a visual vocabulary. In *IEEE International Conference on Robotics and Automation*, 2005.

Jürgen Wolf, Wolfram Burgard, and Hans Burkhardt. Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization. *IEEE Transactions on Robotics*, 21(2): 208–216, 2005.

Zoran Zivkovic, Bram Bakker, and Ben Kröse. Hierarchical map building using visual landmarks and geometric constraints. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 7–12, 2005.

# A Appendix of Multimedia Extensions

The multimedia extensions to this article can be found online by following the hyperlinks from www.ijrr.org.

| Extension | Media Type | Description |
|---|---|---|
| 1 | Video | Results for the New College Dataset. |
| 2 | Video | Results for the City Centre Dataset. |
| 3 | Data | Images, GPS coordinates and ground truth labels. |
| 4 | Video | Generalization Performance - Indoor Dataset A |
| 5 | Video | Generalization Performance - Indoor Dataset B |

Table 2: Index of multimedia extensions

# B Derivation

This appendix presents the derivation of Equation 12 from Section 4.3. For compactness of notation, in this appendix $z_q = s_{z_q}$ will be denoted $z_q$ and $z_q = \overline{s_{z_q}}$ by $\overline{z_q}$, etc.

We seek to express the term $p(z_q|e_q, z_p)$ as a function of the known quantities $p(z_q)$, $p(z_q|e_q)$, $p(z_q|z_p)$.

Applying Bayes Rule:

$$p(z_q|e_q, z_p) = \frac{p(e_q|z_q, z_p)p(z_q|z_p)}{p(e_q|z_p)}$$

now expanding $p(e_q|z_p)$ as

$$p(e_q|z_p) = p(e_q|z_q, z_p)p(z_q|z_p) + p(e_q|\overline{z_q}, z_p)p(\overline{z_q}|z_p)$$

and making the approximation $p(e_q|z_q, z_p) \approx p(e_q|z_q)$, the expression becomes

$$\begin{aligned} p(z_q|e_q, z_p) &\approx \frac{p(e_q|z_q)p(z_q|z_p)}{p(e_q|z_q)p(z_q|z_p) + p(e_q|\overline{z_q})p(\overline{z_q}|z_p)} \\ &= \left(1 + \frac{p(e_q|\overline{z_q})p(\overline{z_q}|z_p)}{p(e_q|z_q)p(z_q|z_p)}\right)^{-1} \end{aligned}$$

Now

$$p(e_q|z_q) = \frac{p(z_q|e_q)p(e_q)}{p(z_q)}$$

and similarly for $p(e_q|\overline{z_q})$, so

$$\frac{p(e_q|\overline{z_q})}{p(e_q|z_q)} = \frac{p(\overline{z_q}|e_q)p(z_q)}{p(z_q|e_q)p(\overline{z_q})}$$

substituting back yields

$$p(z_q|e_q, z_p) \approx (1 + \frac{\alpha}{\beta})^{-1}$$

where

$$\alpha = p(z_q)p(\overline{z_q}|e_q)p(\overline{z_q}|z_p)$$

$$\beta = p(\overline{z_q})p(z_q|e_q)p(z_q|z_p)$$