2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

FAB-MAP 3D: Topological Mapping with Spatial and Visual Appearance

Rohan Paul and Paul Newman

Abstract— **This paper describes a probabilistic framework for appearance based navigation and mapping using spatial and visual appearance data. Like much recent work on appearance based navigation we adopt a bag-of-words approach in which positive or negative observations of visual words in a scene are used to discriminate between already visited and new places. In this paper we add an important extra dimension to the approach. We explicitly model the spatial distribution of visual words as a random graph in which nodes are visual words and edges are distributions over distances. Care is taken to ensure that the spatial model is able to capture the multi-modal distributions of inter-word spacing and account for sensor errors both in word detection and distances. Crucially, these inter-word distances are viewpoint invariant and collectively constitute strong place signatures and hence the impact of using both spatial and visual appearance is marked. We provide results illustrating a tremendous increase in precision-recall area compared to a state-of-the-art visual appearance only systems.**

I. Introduction

This paper concerns non-metric navigation and mapping in appearance space - a by-product of which is loop closure detection. Simply put, we want to have a robot create a topological representation of its trajectory represented by a graph in which each node is a distinct place and edges represent transitions between places. There has been a substantial corpus of work on this topic in recent years (Section II provides an overview) most of which has used a single sensing modality - usually vision. In this work, we provide and test a formulation which uses not only the visual appearance of scenes but also aspects of its geometry. Our approach, called FAB-MAP 3D, has its roots in the FAB-MAP algorithm [7], [8], [9] which has recently been shown to operate in realtime over trajectories of 1000km with high precision [10]. The essence of FAB-MAP is that it learns a probabilistic model of scene appearance online using a generative model of visual word observations and a sensor model which explains missed observations of visual words. We take the same approach in this work but have the added complication that the observation of spatial ranges between words is coupled to the observation of pairs of visual words. We capture this interaction via a random graph which models a distribution over word occurrences as well as their pairwise distances. We describe how through non-parametric Kernel Density Estimation we can learn interesting and suitable distributions over inter-word distances and also accelerate inference by executing a Delaunay tessellation of the observed graph. We shall demonstrate our system and show improved performance over vision only sensing in an outdoor setting.

Our motivation for incorporating range information is two fold. Firstly, prior to this the work, the FAB-MAP framework only modeled the presence or absence of a word at a location and did not incorporate the spatial arrangement of visual words. Hence, the system assigned equal probability to two places if exactly the same visual words were seen in two places, even if the spatial arrangement was different (Figure 2). Secondly, FAB-MAP currently discards the number of times a word appears in a scene - there is information being neglected here. This is addressed in FAB-MAP 3D because by using the range between occurrences of visual words we are implicitly counting word occurrence. Note also that we are in the business of robotics where range information is ubiquitous be it from lidar, stereo or structure from motion - we should use it if we can. Finally, there is also an important prima facia advantage of using distances because they are invariant under rigid transformation and that is precisely what we require of a place descriptor in topological navigation. We must stress though that throughout this work we only need intra-scene distances which can be derived in a local frame, nowhere do we require a single metric picture of the world.

Authors are with the Mobile Robotics Group, University of Oxford, UK.
{rohanp,pnewman}@robots.ox.ac.uk

II. Related work

Use of shape and spatial information for object recognition and categorization has been explored in computer vision. Burl et al. [4] introduced the *constellation model* describing objects as a set of characteristic parts arranged in variable spatial configurations. This was later utilised for category-level object recognition by [13]. Ranganathan et al. [21] extended the model for recognizing indoor workspaces for mobile robots. They present a 3D generative model for indoor places using objects modeled by their shape and appearance with feature positions obtained using depth from a stereo camera. The idea of attribute and random graphs was introduced by [27] who applied them to structural pattern recognition. [23] generalized this framework to *function-described graphs* and applied them to object recognition from 2D views for indoor robotic applications.

There is related research in appearance-based mapping and loop-closure detection. Konolige et al. [16] present a topological mapping scheme, *view based maps,* using geometric feature matching in stereo views and maintaining a vocabulary tree [20] to check loop-closure candidates. Milford and Wyeth [19] describe a large scale appearance-based navigation experiment using biologically inspired techniques.

Tapus et al. [26] model places through *fingerprints* (multi-modal feature based representation like colour bins from cameras and corners from laser scanner) and use POMDP (Partially Observable Markov Decision Process) for mapping and global localization. In [1], Angeli et al. present an incremental loop-closure detection scheme using a bag-of-words approach coupled with epipolar geometric checks.

III. Random Graph Location Model

The world is modeled as a set of independent and disjoint locations. A mobile robot collects image and range observations of the environment and computes the probability that the observation comes from a known location in the topological map or from a new place. We adopt a bag-of-words representation for visual data [24], where images are represented as a set of words or attributes from a vocabulary of size, $|v|$. Additionally, we assume that the vehicle is equipped with a range measuring sensor, e.g., a laser range finder or a stereo camera that gives 3D positions of visual features detected in the scene relative to the vehicle[1].

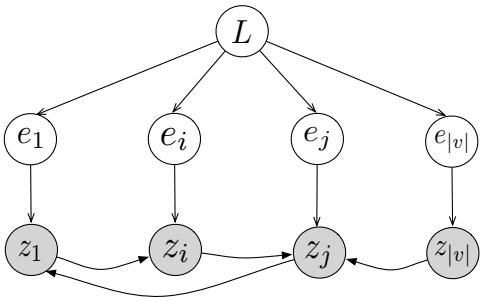
Each location is modeled as a random graph, $L_k = \{E_k, H_k\}$, such that, E_k, represents a random vertex set, $\{e_i | 1 \leq i \leq |v|\}$, where binary variable e_i is the event that the i^{th} word exists at the location[2]. The visual appearance of a place is characterized by the set $\{p(e_1 = 1|L_i), \ldots, p(e_{|v|} = 1|L_i)\}$, an estimate of the probability that each word exists at the location.

H_k, is the random arc set, $\{h_{ij} | 1 \leq i, j \leq |v|\}$, where h_{ij} is the discrete probability distribution (histogram) over euclidean distances in 3D space between the i^{th} and the j^{th} word. These distributions capture the spatial appearance of a location by maintaining the belief over distances between all pairwise words, including words of the same type (details in Section IV).

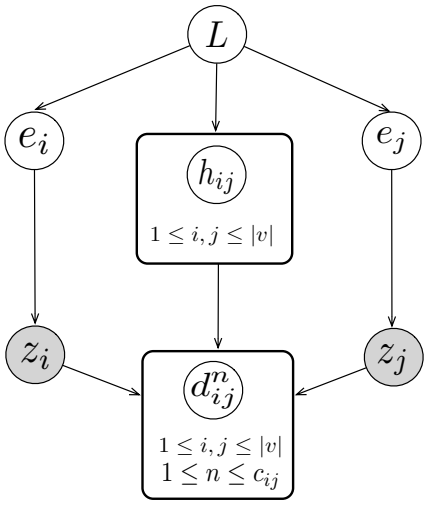
An observation of a local scene is represented by a graph, $G_k = \{Z_k, D_k\}$, where Z_k is the vector $\{z_1, \ldots, z_{|v|}\}$, in which each z_i is a binary variable indicating the presence (or absence) of the i^{th} word of the vocabulary in the scene.

D_k, is the set of spatial distances observed between word pairs, including distances perceived between words of the same type. Let c_{ij}, be the count of all pairwise distances observed between the i^{th} and the j^{th} word. Note that c_{ij} exceeds one when either i^{th} and/or the j^{th} word occurs multiple times and c_{ij} is zero, when either word is not observed. Formally, the set of observed spatial distances, D_k can be represented as $\{d^n_{ij} | 1 \leq i, j \leq |v|, 1 \leq n \leq c_{ij}\}$.

Figure 1 provides a representation of the generative model for locations. A location, L independently generates objects features, e_i which produce observations, z_i detected by the visual sensor. A detector model, Det_{visual} for the appearance sensor connects hidden variables for feature existence to the observed variables of feature detection.



(a) Visual component of the generative model.



(b) Spatial component of the generative model.

Fig. 1: Generative Model: Locations independently generate object features, e_i which produce observations, z_i detected by the visual sensor (top). First-order correlations exist for word observations. Additionally, locations possess distributions over word pair distances, h_{ij} which give rise to observed distances conditioned on the observations z_i and z_j of each word pair (bottom). The model includes distance observations from multiple occurrences of a word.

$$Det_{visual} \begin{cases} p(z_i = 1|e_i = 0), & \text{false positive rate.} \\ p(z_i = 0|e_i = 1), & \text{false negative rate.} \end{cases} \quad (1)$$

In addition, each location possesses distributions over word pairs, h_{ij} which give rise to measured distances, d_{ij}. The uncertainty in the range measuring process is modeled as a gaussian conditional density, Det_{range}, centered on the discrete ranges of h_{ij} and parameterized by variance, σ_{range}.

$$Det_{range} = \quad p(d_{ij}|h_{ij} = x) \sim \quad N(x, \sigma_{range}) \quad (2)$$

IV. Learning Distributions over Word Distances

The spatial appearance of a place is characterized by probability distributions over word distances (Figure 3). Visual words observed by an outdoor mobile robot can appear at highly varied distances, e.g., features detected on foliage are commonly seen at short distances and features on repeated structures like brick walls or railings can display large variations (multi-modal behaviour). To represent

[1]Please note that we do not require large scale 3D reconstruction but simply a way to calculate the range between visual words.

[2]We ask for the reader's forbearance for the counter-intuitive naming of the vertex set as E. This notation is used for consistency with previous FAB-MAP papers.

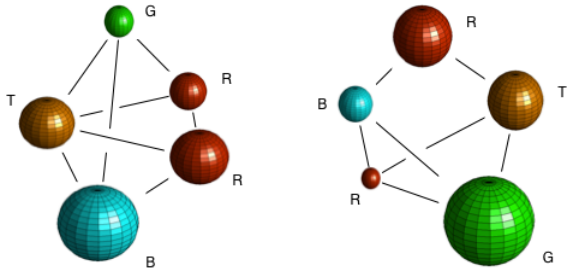


Fig. 2: An illustrative example showing the significance of spatial information. The two scenes have the same visual words {**R**ed, **G**reen, **B**lue, **T**an} but different configurations (pairwise distances). The FAB-MAP framework considers both places to be the same. However, FAB-MAP 3D captures the spatial information through the random graph model and infers the places to be different. Note that sphere sizes differ due to perspective view.

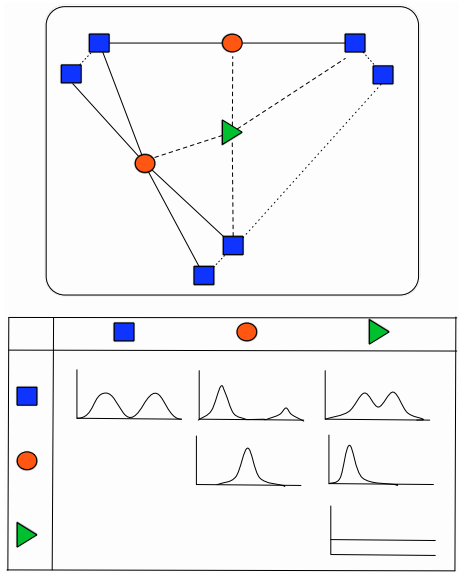


Fig. 3: A toy example illustrating the spatial model with a planar scene (top) with three words {square, triangle and disc}. The distributions over pairwise distances are illustrated below. Note that the figure is not to scale.

these complex multi-modal distributions, we adopt a *non-parametric* histogram representation. Formally, let L_{max} be the maximum distance between any two words in the scene. This is typically the maximum range of a distance measuring sensor. The continuous range is sub-divided into bins, b_k, each of length, Δ and let R be the total number of bins in each histogram ($R = \frac{L_{max}}{\Delta}$) . Let $p(h_{ij} = b_k)$ represent the cumulative density in bin, b_k, where $1 \leq k \leq R$.

$$p(h_{ij} = b_k) \quad = \quad p((k-1)\Delta \leq h_{ij} \leq k\Delta) \qquad (3)$$

Given observed inter-word distances from training data, the probability mass in each bin is estimated through Kernel Density Estimation (KDE) [14]. In this approach, density at a point x is estimated through a linear combination of kernel functions centered on training data $\{x_i\}_{i=1\ldots N}$, where samples $\{x_i\}$ are assumed i.i.d. according to an underlying distribution. The kernel $K(u)$ satisfies the conditions $K(u) \geq 0$ and $\int K(u)du = 1$. The most widely used kernel is the gaussian of zero mean and unit variance for which the KDE can be written as:

$$\begin{aligned} \hat{p}(x) &= \frac{1}{Nh}\sum_{i=1}^{N} K(\frac{x - x_i}{h}) \\ &= \frac{1}{N\sqrt{2\pi}h}\sum_{i=1}^{N} \exp -(\frac{(x-x_i)^2}{2h^2}) \end{aligned} \qquad (4)$$

The kernel function is characterized by a bandwidth (h) that determines the accuracy of the model. Kernels too narrow lead to overfitting and very wide bandwidths lead to underfitting [14]. A number of techniques have been proposed for data-driven bandwidth selection. These methods minimise the *asymptotic integrated mean integrated square error* (AMISE) between the estimate, $\hat{p}(x)$ and the actual density, $p(x)$. The most successful methods rely on estimation of *density derivative functionals* through the *solve-the-equation plug-in method* [22]. We used an implementation of an efficient $\epsilon - exact$ approximation algorithm for optimal bandwidth estimation based on the *improved fast gaussian transform* (IFGT) [28] with computational complexity linear in the number of training points. Once the optimal bandwidth is estimated, Equation 4 is used for calculating probabilities for each histogram bin.[3]

Sometimes due to limited training data, there are very few range samples for rare word pairs. Such sampling error can cause probability estimates in some histogram bins can take degenerate values of 0 or 1. To mitigate this effect, maximum likelihood probability estimates must be smoothed and then renormalized. A variety of smoothing techniques exist [11], typically of the form in Equation 5. We assume uniform prior, i.e., $p_{prior} = \frac{1}{L_{max}}$ and set $K = \sqrt{N}$, where N is the number of training samples. In case no training samples are seen, each bin is assigned a flat prior.

$$p_{smooth} \quad = \quad (\frac{N}{N+K})p_{mle} + (\frac{K}{N+K})p_{prior} \qquad (5)$$

V. Probabilistic Navigation and Mapping

A. Estimating Location

At time k, the workspace is modeled as a collection of n_k discrete and disjoint locations $\mathcal{L}^k = \{L_1, \ldots, L_{n_k}\}$. Given a random graph model for each location, we compute the probability that the observed graph was generated by each location, L_n. Calculating $p(L_n|\mathcal{G}^k)$ can be posed as a recursive Bayes estimation problem:

$$p(L_n|\mathcal{G}^k) = \frac{p(G_k|L_n, \mathcal{G}^{k-1})p(L_n|\mathcal{G}^{k-1})}{p(G_k|\mathcal{G}^{k-1})} \qquad (6)$$

where $p(L_n|\mathcal{G}^{k-1})$ is the prior estimate of the robot's location, $p(G_k|L_n, \mathcal{G}^{k-1})$ represents the observation likelihood,

[3]There is scope for further work here. Since distances are non-negative, kernels with positive support like gamma kernels can be used for density estimation, where mixture parameters can be learnt through expectation maximization [5].

and $p(G_k|\mathcal{G}^{k-1})$ is the normalization constant. Observations are assumed to be conditionally independent given location. Thus, $p(G_k|L_i, \mathcal{G}^{k-1})$ is approximated as $p(G_k|L_i)$. The likelihood that the observed graph was generated by location L_n is factored as two terms: (i) $p(Z_{k,}|L_n)$, likelihood of the visual appearance given location and (ii) $p(D_k|Z_{k,}L_n)$, likelihood of the observed spatial distances conditioned on visual observations and location.

$$\begin{aligned} p(G_k|L_n, \mathcal{G}^{k-1}) &\approx p(G_k|L_n) \\ &= p(\{Z_{k,}D_k\}|L_n) \\ &= p(D_k|Z_{k,}L_n)p(Z_{k,}|L_n) \end{aligned} \quad (7)$$

The visual appearance likelihood term is expanded using the Chow-Liu approximation [6], Equation 8. This expansion approximates the discrete joint distribution $p(z_1, z_2, \ldots, z_{|v|})$ by the closest tree-structured Bayesian network according to the Kullback-Leiber (KL) divergence criteria. Here, z_r is the root of the tree and z_{p_q} is the parent of z_q in the Chow-Liu tree. These factors can further be expressed in terms of prior probabilities, the range detector model and conditionals from training data (details in [7], [9]).

$$p(Z_k|L_n) \approx p(z_r|L_n) \prod_{q=2}^{|v|} p(z_q|z_{p_q}, L_n) \quad (8)$$

Conditioned on the visual observation and location, the spatial likelihood term is estimated as follows:

$$p(D_k|Z_{k,}L_n) = \prod_{i,j=1}^{|v|} \prod_{n=1}^{C_{ij}} p(d_{ij}^n|z_i, z_j, L_n) \quad (9)$$

Pairwise distance edges in the graph are considered independent of other edges given observations of their end points. The likelihood of observing a pairwise distance $p(d_{ij}^n|z_i, z_j, L_n)$ is factored in terms of the prior belief over the distance $p(h_{ij} = b_r|L_n)$ from histogram, h_{ij} and the probability of observing the distance given belief, $p(d_{ij}^n|h_{ij} = b_r)$ via the range detector model. The likelihood is obtained by marginalizing over the discrete range estimates, b_k of the histogram, h_{ij}. The range detector model is assumed independent of location.

$$p(D_k|Z_{k,}L_n) = \prod_{i,j=1}^{|v|} \prod_{n=1}^{|C_{ij}|} \sum_{r=1}^{R} \underbrace{p(d_{ij}^n|h_{ij} = b_r)}_{Det_{range}} \underbrace{p(h_{ij} = b_r|L_n)}_{histogram} \quad (10)$$

B. Evaluating the Normalization Term

The normalization term $p(G_k|\mathcal{G}^{k-1})$ is the total likelihood of the observation G_k. An observation can come from the set of locations currently in the robot's map (M) as well as the set of all previously unknown locations ($\overline{M}$). Hence, $p(G_k|\mathcal{G}^{k-1})$ can be expressed as a sum:

$$\begin{aligned} p(G_k|\mathcal{G}^{k-1}) &= \sum_{m \in M} p(G_k|L_m)p(L_m|\mathcal{G}^{k-1}) \\ &+ \sum_{u \in \overline{M}} p(G_k|L_u)p(L_u|\mathcal{G}^{k-1}) \end{aligned} \quad (11)$$

The second term involves summation over all unmapped places and cannot be directly computed. The summation is approximated through mean field approximation[4] [15] by constructing an *average place* model, $L_{avg} = (E_{avg,}H_{avg})$.

$$\begin{aligned} p(G_k|\mathcal{G}^{k-1}) &\approx \sum_{m \in M} p(G_k|L_m)p(L_m|\mathcal{G}^{k-1}) \\ &+ p(G_k|L_{avg}) \sum_{u \in \overline{M}} p(L_u|\mathcal{G}^{k-1}) \end{aligned} \quad (12)$$

The visual appearance component of the average place, E_{avg} is constructed by assigning e_i variables their marginal probabilities from training data. In similar vein, the spatial appearance of the average place, H_{avg} is the set of marginal histograms for each word pair, where each histogram is a density estimate learnt from all pairwise distance samples observed in training data.

This formulation also addresses the perceptual aliasing problem: the fact that different parts of the environment appear the same to robot's sensors. e.g., similar looking foliage and brick walls appear commonly while navigating outdoors. The visual appearance model for the average place learns which features are common in the environment, like words appearing on foliage have high marginal probabilities. Additionally, the spatial appearance model for the average place learns what distances words commonly appear at. Hence, it can learn that features detected on brick walls commonly appear at repeated distances. Overall, the system matches an observation to a location only when *both* the visual and spatial appearance is distinctive.

C. Updating Location Model

A new location in the topological map is initialized with the average place model where (i) word generators exist with marginal probability, $p(e_i = 1|L_{new}) = p(e_i = 1)$ and (ii) word pair histograms are initialized to marginal histograms, $p(h_{ij} = b_r|L_{new}) = p(h_{ij} = b_r)$. When an observed graph relates to a location in the map, the random graph model for the location is updated according to the current belief and the sensor models. For the visual component, the probability of feature existence, $p(e_i = 1|L_n)$ is updated as:

$$p(e_i = 1|L_n, \mathcal{G}^k) = \frac{p(z_i|e_i = 1)p(e_i = 1|L_n, \mathcal{G}^{k-1})}{p(z_i|L_n)} \quad (13)$$

Additionally, the observed word-pair distances are used to update their corresponding density historgrams. Hence, probability associated with each bin, $p(h_{ij} = b_r|L_n)$ is updated as follows:

$$p(h_{ij} = b_r|L_n, \mathcal{G}^k) = \frac{p(d_{ij}^n|b_r)p(h_{ij} = b_r|L_n, \mathcal{G}^{k-1})}{p(d_{ij}^n|L_n)} \quad (14)$$

[4]As described in [9], a superior alternative to the *mean-field* approximation is the *sampling* based approximation. The current implementation uses the *mean-field* approach due to lack of a large representative dataset with both vision and range data required for constructing the sampling set. Collecting a larger dataset is planned as part of future work.

This step assumes that observations of a word or an observed distance between a word-pair does not convey information about either existence or pairwise distances of other words. The data association decision for observations and locations is based on maximum likelihood criterion. Loop closures are accepted only when the loop closure probability exceeds a user defined threshold, $p_{accept} = 0.999$.

VI. Accelerating Graph Likelihood Computation

In a given scene, let N_f be the total number of visual words detected. While computing the spatial likelihood term $p(D_k|Z_k, L_n)$, we compute individual distance likelihoods for $\frac{N_f(N_f-1)}{2} \approx O(N_f^2)$ pairwise distances. Essentially, all distance edges of the observed 3D graph are considered and $O(N_f^2)$ histograms are updated according to Equation 13. However, features detected in a workspace originate from objects that generally possess high local spatial correlation, like features detected on a window. Similarly, features separated by large distances are spatially less correlated. Using this intuition, we would like to consider distances only to *neighbouring* points, where by *neighbouring* we imply a pair of points whose cells in the Voronoi tessellation share an edge. Formally, we compute the Delaunay tessellation of the 3D graph that results in a division of the graph into tetrahedrons (simplices) such that no data point is contained in any circumsphere of the simplices[5] [12]. The condition on circumspheres prevents the tessellation to return skewed tetrahedrons, in effect connecting points to local neighbours. We restrict the graph likelihood computation only to edges of the tessellated graph which scales $O(N_f)$ compared to $O(N_f^2)$ for the complete graph. We use an implementation based on *Qhull*[6], a standard computational geometry package, to compute the tessellation [2]. The 3D Delaunay tessellation algorithm has $O(N_f log N_f)$ complexity [2]. Under certain cases, a valid tessellation does not exist due to numerical issues or coincident points. In such cases we can use nearest-neighbour criterion to pick relevant edges. An experimental evaluation of several implementations of 3D Delaunay tessellations appears in [18].

VII. Evaluation

A. Platform and dataset

The topological mapping algorithm was tested on image and laser data from a mobile robot shown in Figure 4. Imagery was captured at 3Hz from a Point Grey Ladybug 2 camera and laser data was obtained from a SICK LMS291 laser, scanning 90° at 75Hz with 0.5° resolution. The laser is mounted so as to scan in a vertical plane normal to the vehicle's forward motion. The camera and the laser are experimentally cross-calibrated. The dataset was gathered within New College, Oxford in an environment of medieval buildings enclosing an oval lawn and cambered tarmac space [25]. The site possesses repetitive architectural features causing perceptual aliasing and is also traversed by people, thereby testing the system's robustness to scene change.

[5]Delaunay tessellation is the dual of Voronoi tessellation.

[6]Available at http://www.qhull.org/

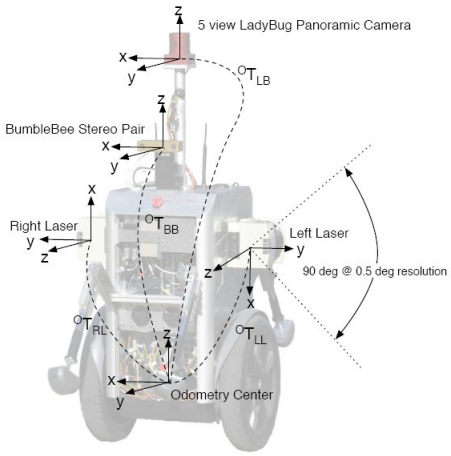


Fig. 4: Robotic platform used for experimentation with sensors and coordinate frame centres shown.

B. Processing pipeline

Every image is converted to a bag-of-words representation by first extracting SURF features [3] and then quantizing them against a fixed vocabulary to obtain visual words for the image, yielding Z_k. The vocabulary is generated by clustering SURF features obtained from training images where cluster centres correspond to vocabulary visual words. A vocabulary size of 10K words is employed. The SURF descriptor also determines the scale at which the feature was detected, hence approximating feature size in image space.

The next task is to determine inter-word distances for the scene. Laser scans obtained in a 16sec window around image capture time are back projected into the view of the camera. For each visual word detected in the image, close laser points are determined that lie within a radius equal to feature size (from the SURF descriptor). The visual word is assigned a 3D coordinate by taking an inverse-radially weighted-average of the 3D coordinates of all nearby laser points within the search radius. Hence, we now know where visual words lie in 3D space and can determine pairwise distances forming the set D_k. The resulting observation graph, $G_k = \{Z_k, D_k\}$ is then passed on to the inference engine.

The next step is building the vocabulary model by constructing the Chow-Liu tree by a procedure outlined in [9]. It consists of constructing the mutual information graph using word co-occurrence data from the training set and then computing the maximum weight spanning tree. The marginal pairwise distance histograms were determined from a training set of 400 images. Although the set of all pairwise distance histograms is very large $(10K \times 10K)$, only a relatively smaller number, 557491 word pairs were observed to co-occur. For the other pairs a uniform prior over ranges was assumed. Since the word co-occurrence matrix is very sparse, the number of histograms required for inference is tractable. For space efficiency, only a single global copy of the marginal word-pair histograms is maintained. While initializing a new place model, only the modified distance histograms are maintained locally.

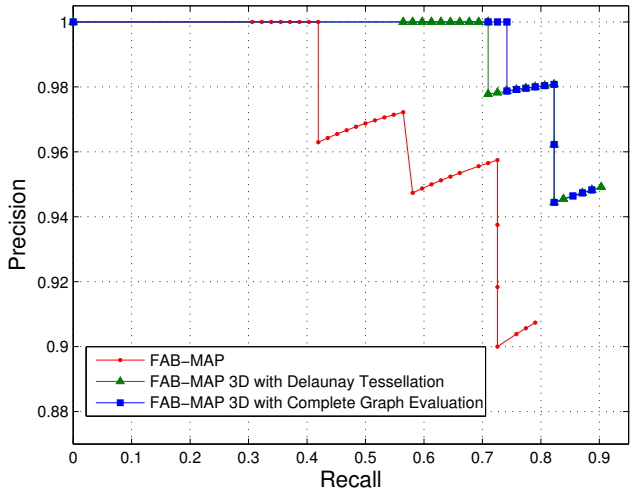


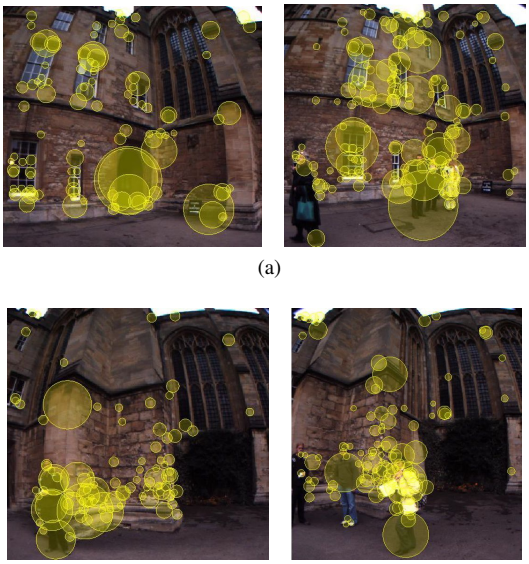
Fig. 5: Precision-recall curves comparing FAB-MAP, FAB-MAP 3D with complete graph evaluation and FAB-MAP 3D with accelerated graph inference on the New College data set. Note the scale. FAB-MAP 3D has a higher recall of 74% at 100% than FAB-MAP that has 42% recall at 100% precision. The graph for the accelerated approach partially overlaps with FAB-MAP 3D with complete evaluation. The accelerated approach has marginally lower recall of 71% but still performs better than FAB-MAP.

The final ingredient is the detector model. For the visual detector, $p(z_i = 1|e_i = 0) = 0$ and $p(z_i = 0|e_i = 1) = 0.39$. The variance for the range detector, σ_{range} was set to 1.5m. Although, as noted by [17], the range uncertainty for LMS laser scanners (for close range) is $\approx$ 3cm, the range detector model also incorporates (i) uncertainty arising from vehicle odometry errors that affect projection of laser scans into image space taken a few seconds before or after image time and (ii) slight errors in cross-calibrating the laser ranger and the camera.

C. Results

The test set consisted of 117 images. Precision-recall curves are shown in Figure 5. The curves are obtained by varying the probability threshold at which loop closures are accepted. Ground truth was obtained from GPS data and determined by hand in sections where reception was intermittent. For comparing only the core inference aspect of the system, the prior probability of being at a location was kept uniform (no motion model) for both implementations.

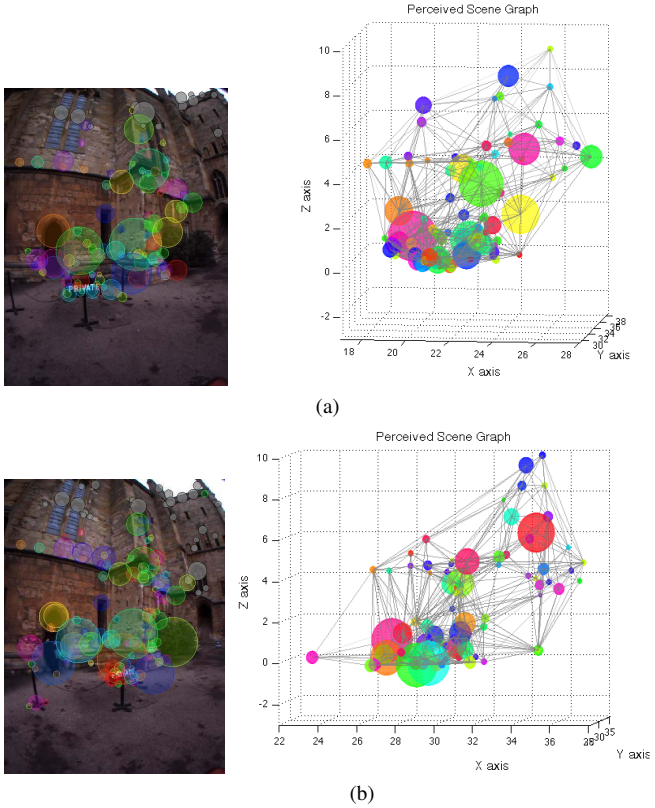
FAB-MAP 3D with complete graph evaluation achieved 100% precision at 74% recall whereas the original FAB-MAP algorithm had a lower recall of 42% at 100% precision. In our setting, recall refers to the fraction of total loop closures that exceed the probability threshold and hence declared by the system [9]. FAB-MAP 3D with accelerated graph inference based on Delaunay tessellation achieved 100% precision at 71% recall. Hence, the graph approximation has a marginally lower recall than the full graph computation but still significantly outperforms FAB-MAP.



(a)

(b)

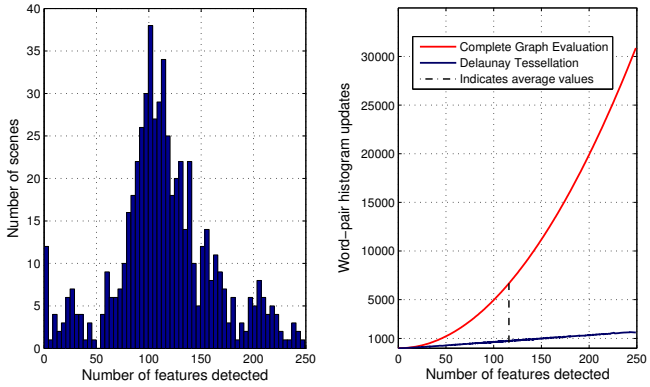
Fig. 6: Examples of true loop closures detected by FAB-MAP 3D using spatial similarity and not detected by FAB-MAP.



(a)

(b)

Fig. 7: An example of a true loop closure detected by FAB-MAP 3D with high confidence whereas FAB-MAP assigned close to zero loop closure probability. The observed 3D graphs for both scenes appear on the right. Spheres represent visual words and colours indicate word type. The broad similarity in the graphs enables FAB-MAP 3D to infer loop closure. Image features shown in gray did not have associated range information due to limited coverage of the laser scanner, hence not included in the graph computation.



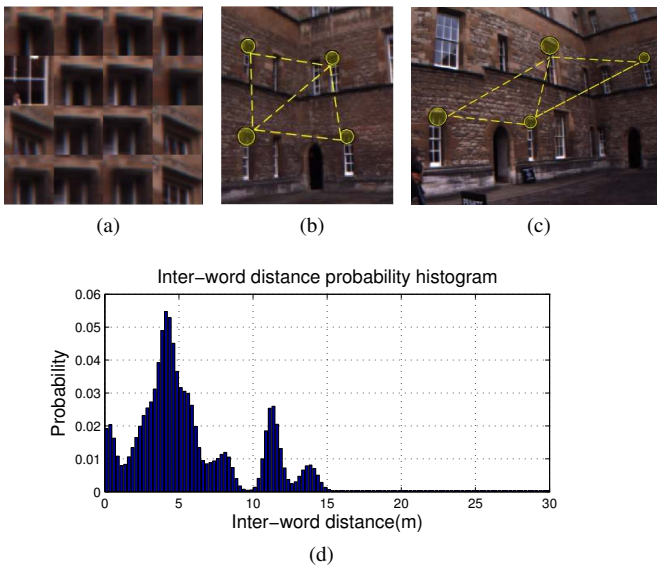
(a) (b)

Fig. 8: (a) Number of scenes vs. number of features detected, N_f for the combined set of training and test sets (517 locations). Average $N_f = 116$ (std. dev. 48, median 112). (b) Number of word-pair distance probability histogram updates vs. N_f. The histogram updates required, scales quadratically in N_f for the complete graph evaluation and linearly for computation with Delaunay tessellation, illustrating the speed-up of the appoximate scheme.

Figure 6 illustrates examples of true loop closures declared by FAB-MAP 3D but not detected by FAB-MAP. The perceived 3D graphs for one such loop closure pair appear in Figure 7. The spheres represent visual words and colours indicate word type. The graphs are similar but possess minor differences caused by scene change (dynamic objects) as well as sensor uncertainty. The FAB-MAP framework only considers words presence and does not find the two places to be distinctively similar. However, the probabilistic random graph framework in FAB-MAP 3D models both word presence and spatial characteristics. Utilizing the extra spatial information, the system infers high likelihood of the perceived graphs originating from the same location (random graph), thereby declaring loop closure with high confidence.

Figure 9 presents an example of learning word-pair distance probability histograms. Figure 9a shows a visual word that typically appears on the upper half of windows, occurring repetitively in the environment (Figures 9b and 9c). Figure 9d plots the probability distribution histogram modeling distances between multiple occurrences of this word. The variable inter-word distances due to repetitive structure are captured by the multi-modal distribution learned through kernel density estimation with optimal bandwidth estimation. Note that smoothing prevents probability estimates for (unlikely) large distances from becoming zero. This provides support for a possible later observation in this range to be incorporated via recursive Bayesian updates for each bin during data association.

Figure 8a plots the number of scenes vs. number of features detected in each scene (N_f), representing scene complexity, for the combined set of training and test sets (517 locations). The average N_f was found to be 116 (standard



(a) (b) (c) (d)

Fig. 9: Learning word pair histograms. (a) A visual word that typically appears on the upper half of windows, (b) and (c) are the typical scenes where the word occurs repeatedly at regular distances. (d) The learned multi-modal probability distribution (histogram) modeling the distances between multiple occurrences of this word.

(a) (b) (c)

Fig. 10: Practical limitations. Laser scan points projected into the camera frame are shown in blue. (a) Laser scanner range is insufficient to cover the entire image causing absence of range estimates for visual words detected in the upper portion of the image. (b) The rectangular patch seen on the background wall forms the shadow for the signpost for the laser scanner. Hence, no laser points project in this region. (c) Dynamic objects like moving people can cause incorrect point clouds to appear in the scene.

deviation 48, median 112). The main additional computational cost of FAB-MAP 3D over FAB-MAP stems from the number of inter-word distance probability histograms updated per scene. Figure 8b plots the number of histogram updates vs. N_f for the data set. The number of histogram updates required, scales quadratically with scene complexity for the complete graph evaluation and linearly for computation with Delaunay tessellation, illustrating the advantage of the approximate method with a marginal decrease in performance (Figure 5). Computing the tessellation scales log-linearly with N_f, however the *Qhull* implementation

is very fast adding little overhead cost (average 12ms per scene).

Figure 10 illustrates practical limitations of the laser ranging based approach for estimating 3D positions for visual words. Different positioning of the camera and the lasers can cause a disparity in the viewing or sensing regions of the two sensors. Laser scans projected into the camera frame may not cover the entire image (Figure 10a), causing absence of range estimates for some visual features. Figure 10b shows a rectangular patch on the background wall with no projected laser points, since for the scanning laser this region appears in the shadow of the foreground signboard. Additionally, dynamic objects like people cause scene change between the image obtained by the camera and the laser scans taken later in time, causing incorrect point clouds to appear in the scene (Figure 10c). Also, surfaces axis parallel to the laser scanner yield very few reflections and hence sparse range estimates for a surface which could possess many visual features. Computing the inverse-radially weighted-average of the 3D coordinates of the projected laser points within feature radius provides some robustness to such cases.

VIII. Conclusions

This paper introduced a probabilistic framework for appearance based topological mapping. In this formulation, locations are represented as random graphs and a generative model is learnt over word occurrences as well as their spatial distributions. This approach provides substantial and compelling improvement in precision-recall performance over the existing FAB-MAP system. By capturing spatial information, the algorithm reduces the number of false positives and shows a dramatic decrease in false negative rate, particularly in scenes possessing a large number of common words where a loop closure decision hinges on spatial information. The framework shows robustness to perceptual aliasing as well as scene change. The system scales linearly with the number of places in the map. We also presented a method for accelerating graph inference based on Delaunay tessellation of the observed graph, that scales log-linearly with scene complexity.

IX. Acknowledgements

Special thanks to Mark Cummins for constant support and guidance. We are grateful to the anonymous reviewers for their useful suggestions. This research was supported by the Rhodes Trust, University of Oxford, UK.

References

[1] A. Angeli, D. Filliat, S. Doncieux, and J.A. Meyer. A Fast and Incremental Method for Loop-Closure Detection Using Bags of Visual Words. *IEEE Transactions On Robotics, Special Issue on Visual SLAM*, 24(5):1027–1037, 2008.

[2] C.B. Barber, D.P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.

[3] H. Bay, T. Tuytelaars, and L. 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.

[4] M.C. Burl, M. Weber, and P. Perona. A probabilistic approach to object recognition using local photometry and global geometry. *Lecture notes in computer science*, 1407:628–641, 1998.

[5] SC Choi and R. Wette. Maximum likelihood estimation of the parameters of the gamma distribution and their bias. *Technometrics*, pages 683–690, 1969.

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

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

[8] M. Cummins and P. Newman. Accelerated appearance-only SLAM. In *Proc. IEEE International Conference on Robotics and Automation (ICRA'08)*, Pasadena, California, April 2008.

[9] M. Cummins and P. Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[10] M. Cummins and P. Newman. Highly scalable appearance-only SLAM - FAB-MAP 2.0. In *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.

[11] J. Cussens. Bayes and Pseudo-Bayes Estimates of Conditional Probabilities and Their Reliability. *Lecture Notes in Computer Science*, pages 136–136, 1993.

[12] B. Delaunay. Sur la sphere vide [On the empty area]. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, 7:793–800, 1934.

[13] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings*, 2, 2003.

[14] MC Jones, JS Marron, and SJ Sheather. A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association*, 91(433), 1996.

[15] M.I. Jordan, Z. Ghahramani, T.S. Jaakkola, and L.K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, 1999.

[16] K. Konolige, J. Bowman, J.D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua. View-based maps. In *Proceedings of Robotics: Science and Systems (RSS)*, 2009.

[17] K.H. Lee and R. Ehsani. Comparison of two 2D laser scanners for sensing object distances, shapes, and surface patterns. *Computers and Electronics in Agriculture*, 60(2):250–262, 2008.

[18] Y. Liu and J. Snoeyink. A comparison of five implementations of 3D Delaunay tessellation. *Combinatorial and Computational Geometry*, 52:435–453, 2005.

[19] M.J. Milford and G. Wyeth. Mapping a suburb with a single camera using a biologically inspired slam system. *IEEE Transactions on Robotics*, 24(5):1038–1053, 2008.

[20] D. Nistér and H. Stewenius. Scalable recognition with a vocabulary tree. In *Conf. Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, 2006.

[21] A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. *Robotics: Science and Systems*, 2007.

[22] V.C. Raykar and R. Duraiswami. Fast optimal bandwidth selection for kernel density estimation. *Proceedings of the sixth SIAM International Conference on Data Mining*, pages 524–528, 2006.

[23] A. Sanfeliu. The use of graph techniques for identifying objects and scenes in indoor building environments for mobile robots. *Lecture notes in computer science*, pages 30–41, 2004.

[24] 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.

[25] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The New College Vision and Laser Data Set. *The International Journal of Robotics Research*, 28(5):595, 2009.

[26] A. Tapus and R. Siegwart. A cognitive modeling of space using fingerprints of places for mobile robot navigation. *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2006)*, pages 1188–1193, May 2006.

[27] AKC Wong and M. You. Entropy and distance of random graphs with application to structural pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(5):599–609, 1985.

[28] C. Yang, R. Duraiswami, NA Gumerov, and L. Davis. Improved fast Gauss transform and efficient kernel density estimation. *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, pages 664–671, 2003.