Appearance-based SLAM in a Network Space

Padraig Corcoran[1], Ted J. Steiner [2], Michela Bertolotto[1] and John J. Leonard[2]

***Abstract*— The task of Simultaneous Localization and Mapping (SLAM) is regularly performed in network spaces consisting of a set of corridors connecting locations in the space. Empirical research has demonstrated that such spaces generally exhibit common structural properties relating to aspects such as corridor length. Consequently there exists potential to improve performance through the placement of priors over these properties. In this work we propose an appearance-based SLAM method which explicitly models the space as a network and in turn uses this model as a platform to place priors over its structure. Relative to existing works, which implicitly assume a network space and place priors over its structure, this approach allows a more formal placement of priors. In order to achieve robustness, the proposed method is implemented within a multi-hypothesis tracking framework. Results achieved on two publicly available datasets demonstrate the proposed method outperforms a current state-of-the-art appearance-based SLAM method.**

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is a problem in the field of robotics which concerns modelling, or mapping, the geometry of the space within which a robot is located, while simultaneously localizing within this model [1]. SLAM methods can broadly be considered as belonging to two categories commonly referred to as appearance-based and metric SLAM methods which are distinguished by the types of properties they model. Appearance-based SLAM methods model the set of discrete locations in a space and the existence of paths between these locations [2]. On the other hand, metric SLAM methods (specifically a pose-graph formulation [3]) model the set of discrete locations in a space, the existence of paths between these locations and the metric transformations between all locations. As alluded to, both categories of methods are not distinct and in fact metric SLAM methods commonly use appearance-based methods as a *front-end* which proposes loop-closures.

SLAM is regularly performed in network spaces consisting of a set of corridors connecting locations in space. Such spaces include street networks and building interiors. A number of empirical studies have demonstrated that such networks generally exhibit common structural properties [4], [5]. For example, corridors are generally not of small length. Also, it is uncommon to have more than four corridors meeting at a single point. Consequently, when it is known a priori that SLAM is being performed in a network space, priors may be placed on the properties of the space. In this work we propose an appearance-based SLAM method which explicitly models the space as a network which in turn allows a formal placement of priors. Specifically, the space is modelled as a graph $G = (V, E)$ where V is a set of vertices modelling points where corridors meet and E is a set of edges modelling corridors connecting adjacent vertices. Each edge is in turn modelled as a sequence of interior points, or poses, where the number interior points represents the length of the edge. We place a prior over the space of graphs such that those graphs of lesser complexity are assigned a higher relative probability. The complexity of a graph is quantified in terms of the number of vertices and edges of small length it contains. This prior represents an implementation of Occam's razor [6]. Inference with respect to this model is performed using recursive Bayesian estimation. In order to achieve robustness, this is implemented within a multi-hypothesis tracking framework.

This approach represents a formalization of existing appearance-based SLAM methods which implicitly assume a network space and place priors over its structure [7], [8]. These methods assume that valid loop-closures occur in sequences which is in fact a consequence of one's trajectory being constrained to visit the same sequence of locations when it retraverses a network corridor. The advantage of the proposed formalization is that it allows the placement of specific priors in a formal way.

The layout of this paper is as follows. Section II reviews related work. Section III describes in detail the proposed network model of space. Section IV describes the recursive Bayesian estimation formulation used. Section V describes the multi-hypothesis tracking framework implemented. Finally in sections VI and VII we present results and draw conclusions respectively.

II. RELATED WORK

Given the vast literature in the SLAM domain, in this section we only consider those SLAM methods which are appearance-based, place priors over the structure of the space or perform multi-hypothesis tracking. A number appearance-based SLAM methods determine loop closures using solely appearance information [2], [8]. It has been demonstrated that augmenting appearance information with metric information can improve performance. In [9], [7] the authors augment appearance information with local geometric information relating to the spatial configuration of features. In

*This work was supported in part by a European Marie Curie International Outgoing Fellowship

[1]Padraig Corcoran and Michela Bertolotto are with the School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland. padraig.corcoran@ucd.ie; michela.bertolotto@ucd.ie

[2]Ted J. Steiner and John J. Leonard are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. tsteiner@mit.edu; jleonard@mit.edu

[10], [11] the authors augment appearance information with metric robot pose information.

The placement of priors over the structure of a space when performing SLAM has recently been considered by a number of authors. Salas-Moreno et al. [12] proposed a method for performing SLAM in man-made spaces by placing a prior over the fact that such spaces contain many planes. In this work we propose a SLAM method for network spaces and place priors over the structure of this space. A number of existing SLAM methods implicitly assume a network space and place priors over its structure. These methods assume that valid loop-closures occur in sequences or groups which is in fact a consequence of one's trajectory being constrained to visit the same sequence of locations when it retraverses a network corridor. We now consider these works. Latif et al. [13] proposed a metric SLAM method called Realizing, Reversing, Recovering (RRR) where sequences of loop-closers are proposed and subsequently accepted or rejected depending on whether they are consistent or inconsistent with odometry measurements respectively. Galvez-Lopez and Tardos [7] proposed an appearance-based SLAM method which contains a verification step that rejects loop-closures if they do not occur in a sequence. Milford [8] proposed an appearance-based SLAM method which uses dynamic time warping to find sequences of loop closures. Finally it is worth noting that the appearance-based SLAM method of FabMap by Cummins and Newman [2] does explicitly assume that valid loop-closures occur in sequences. However it does incorporates a weak motion prior such that the likelihood of matching to adjacent locations is slightly higher. These methods are constrained by the fact that they do not allow the formal placement of specific priors.

Multi-hypothesis tracking has previously been used by SLAM methods in order to achieve robustness [14] [15]. However the authors believe it has yet to be considered in the context of appearance-based SLAM.

III. Network and Location Models

Existing appearance-based SLAM methods model the space as a graph where the set of vertices model points in space and the set of edges model the existence of paths between adjacent vertices. Within this model one's location is modeled as corresponding to an individual vertex. Toward illustrating this model, consider the environment represented in Figure 1(a) which contains an upper, a middle and a lower corridor where a and b indicate where the individual corridors meet. Now consider the case where one begins at a, traverses the middle path to b, then traverses the lower path to a, then traverses the middle path to b and finally traverses the upper path to a. If we assume the ability to detect the events of encountering previously visited and unvisited vertices, the estimated set of vertices and edges would correspond to that illustrated in Figure 1(b). Appearance information is stored for each vertex and this allows the events of encountering previously visited and unvisited vertices to be detected.

In this work we model the space as a graph $G = (V, E)$. The set of vertices $V = \{v^1, \ldots, v^n\}$ model the points

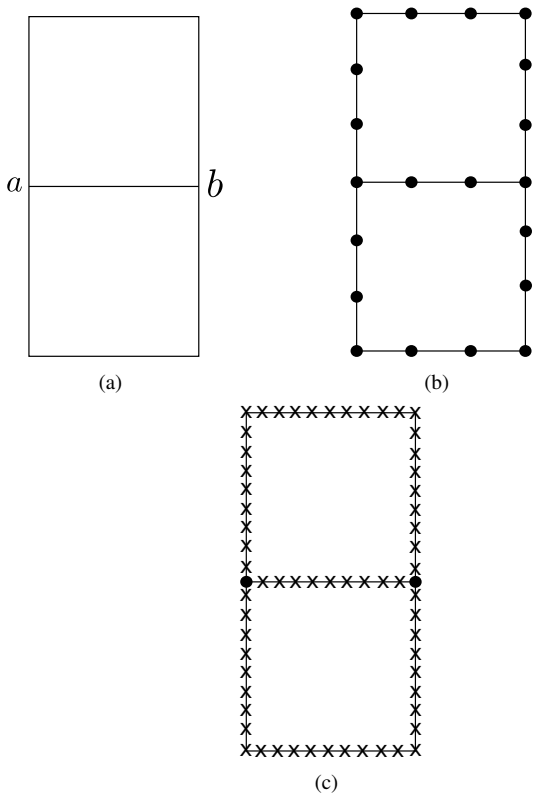


Fig. 1. An environment containing three corridors represented by lines is illustrated in (a). Corresponding traditional and proposed graph models of this environment are illustrated in (b) and (c) respectively where circles, lines and x's represent vertices, edges and edge interior points respectively.

where corridors meet. The set of edges $E = \{e^1, \ldots, e^m\}$ model corridors connecting adjacent vertices. Each $e^i \in E$ is in turn modeled as a sequence of interior points $e^i = \{e^i_1, \ldots, e^i_k\}$ where the order corresponds to that in which the points are encountered as one traverses the edge in a given direction. One's location L is modeled as corresponding to an individual edge interior point; that is $L = e^i_j$ for some i and j. Toward illustrating this model consider again the space represented in Figure 1(a) and the same trajectory through this space described above. If we assume the ability to robustly detect the events of encountering previously visited and unvisited edge interior points, the estimated graph would correspond to that illustrated in Figure 1(c). For each point e^i_j appearance information is stored and denoted $A(e^i_j)$; see section IV-C for specific details regarding how appearance is modelled. We do not perform inference with respect to the connectivity between individual corridors; upon exiting any corridor the probability of entering all corridors is equal. Consequently, we do not explicitly model this connectivity and instead model the graph as a set of edges.

Consider a hypothesis at time $t-1$ and corresponding hypothesis extension at time t denoted (G_{t-1}, L_{t-1}) and (G_t, L_t) respectively. The location L_{t-1} in G_t is denoted $T^{G_{t-1}}_{G_t}(L_{t-1})$. To illustrate this term consider the sample hypothesis (G_{t-1}, L_{t-1}) and corresponding extension (G_t, L_t) illustrated in Figure 2(a) and 2(b) respectively. The extension

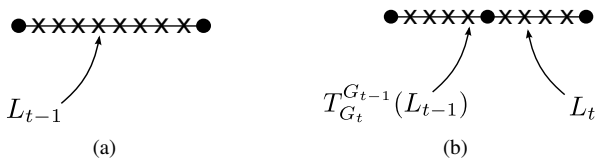


Fig. 2. A hypothesis (G_{t-1}, L_{t-1}) and corresponding extension (G_t, L_t) are illustrated in (a) and (b) respectively.

in question corresponds to the splitting of the single edge in G_{t-1} into two edges, the addition of a new vertex to G_{t-1} and the assignment of L_t. In this case $T_{G_t}^{G_{t-1}}(L_{t-1})$ corresponds to that location illustrated in Figure 2(b).

IV. Recursive Bayesian Estimation

Let (G_{i-1}, L_{i-1}) and (G_i, L_i) denote a hypothesis at time $i-1$ and a corresponding extension of this hypothesis at time i respectively. We make the Markov assumption that the probability of (G_i, L_i) given (G_{i-1}, L_{i-1}) is conditionally independent of (G_{i-j}, L_{i-j}) for $j > 1$. Let Z_i be a sensor appearance measurement at time i which will be described in section IV-C. We assume that Z_i is dependent only upon (G_i, L_i). Based on these assumptions the joint distribution over (G_i, L_i) given Z_i for i from time 0 to the current time t can be factored as Equation 1. In this work we are interested in estimating only (G_i, L_i). We therefore factor Equation 1 as the recursive estimation of the joint distribution over G_t and L_t of Equation 2. This factorization contains the following four terms: a data term $P(Z_t|G_t, L_t, G_{t-1}, L_{t-1})$, a graph term $P(G_t|G_{t-1}, L_{t-1})$, a location term $P(L_t|G_t, G_{t-1}, L_{t-1})$ and a prior term $P(G_{t-1}, L_{t-1})$. For a given hypothesis H, the product of the corresponding graph, location and data terms equals the hypothesis likelihood and is denoted $\mathcal{L}(H)$. The following three subsections describe the estimation of each of these terms.

$$\begin{aligned} &P(G_0, L_0, \ldots, G_t, L_t|Z_0, \ldots, Z_t) = \\ &P(G_0, L_0)\prod_{i=1}^{t} P(Z_i|G_i, L_i)P(G_i, L_i|G_{i-1}, L_{i-1}) = \\ &P(G_0, L_0)\prod_{i=1}^{t} P(Z_i|G_i, L_i)P(L_i|G_i, G_{i-1}, L_{i-1}) \\ &\qquad P(G_i|G_{i-1}, L_{i-1}) \end{aligned} \quad (1)$$

$$\begin{aligned} &P(G_t, L_t|Z_t, G_{t-1}, L_{t-1}) = P(Z_t|G_t, L_t) \\ &P(L_t|G_t, G_{t-1}, L_{t-1})P(G_t|G_{t-1}, L_{t-1})P(G_{t-1}, L_{t-1}) \end{aligned} \quad (2)$$

A. Graph Term

In this section we describe how the graph term $P(G_t|G_{t-1}, L_{t-1})$ is evaluated. This term models the probability that the graph G_{t-1} was transformed into G_t. It is designed such that a greater relative probability is assigned to those graphs G_t which represent a lesser increase in complexity relative to G_{t-1}. This represents an implementation of Occam's razor. The graph term is defined in Equation 3 to be the probability of the graph $G_t = (V_t, E_t)$ which is in turn factored to be a product of the probability of V_t and E_t which we now describe.

$$P(G_t|G_{t-1}, L_{t-1}) = P(G_t) = P(V_t)P(E_t) \quad (3)$$

The probability distribution $P(V_t)$ is defined in Equation 4 where p_v is a specified model parameter in the range $[0, 1)$ and α is a normalization constant corresponding to an infinite geometric series that may be computed easily [16]. This distribution assigns a higher relative probability to those graphs G with fewer vertices. Choosing a specific p_v value has the effect of placing a specific prior over the degree to which the number of vertices measures the complexity of the graph in question.

$$P(V_t) = \alpha p_v^{|V_t|} \sim p_v^{|V_t|} \quad (4)$$

The probability distribution $P(E_t)$ is defined in Equation 5 where p_e is a specified model parameter in the range $[0, 1)$ and β is a normalization constant. Computing the normalization constant in this case is intractable. For a given edge e the value $1 - p_l^{|e|}$ approaches 1 as $|e|$ approaches ∞ where the rate of convergence is a function of p_e. This distribution assigns a higher relative probability to those graphs G with fewer edges containing a small number of points and in turn corresponds to a model of lesser complexity. Choosing a specific p_e value has the effect of placing a specific prior over the degree to which the number of edges of small length measures the complexity of the graph in question.

$$P(E_t) = \beta \prod_{e \in E_t} 1 - p_e^{|e|} \sim \prod_{e \in E_t} 1 - p_e^{|e|} \quad (5)$$

In this work our goal is to compute the most probable graph G_t and not its actually probability. Therefore it is only necessary to specify the probability distribution $P(G_t)$ up to a constant term. This is achieved in Equation 6 by combining Equations 3, 4 and 5.

$$P(G_t) \sim \left(p_v^{|V_t|}\right)\left(\prod_{e \in E_t} 1 - p_e^{|e|}\right) \quad (6)$$

Evaluating the graph term $P(G_t|G_{t-1}, L_{t-1})$ up to a constant using Equation 6 has two potential issues. Firstly, as a consequence of the fact that it is an unnormalized probability value, the evaluation of Equation 6 will approach zero as the number of graph vertices or edges approaches infinity. This will in turn result in arithmetic underflow. Secondly, since the set of vertices and edges within a graph will not change significantly from one time step to the next, the evaluation of Equation 6 will result in repeated computation.

In order to overcome these issues the following solution is implemented. Firstly, at each time t we normalize the set of

hypotheses being tracked such that the sum of their posterior probabilities, $P(G_t, L_t|Z_t, G_{t-1}, L_{t-1})$, is 1. Secondly, we evaluate the graph term $P(G_t|G_{t-1}, L_{t-1})$ up to a constant using Equation 7 where ΔV and ΔE are the set of vertices and edges respectively added to G_{t-1} to obtain G_t. We now describe how the terms $P(\Delta V_t)$ and $P(\Delta E_t)$, which we refer to as the *delta vertex* and *delta edge* terms respectively, are evaluated.

$$P(G_t|G_{t-1}, L_{t-1}) \sim P(\Delta V_t)P(\Delta E_t)P(G_{t-1}) \quad (7)$$

1) Delta Vertex:
The term $P(\Delta V_t)$ is computed using Equation 8 where p_v is the parameter specified in section IV-A. The form of this equation is a consequence of the fact that within our environment model vertices may only be added but not removed at each time step.

$$P(\Delta V_t) = p_v^{|\Delta V_t|} \quad (8)$$

2) Delta Edge:
The term $P(\Delta E_t)$ is computed using Equation 9 where the term $P(e)$ specifies the probability that the edge e was added. If e is an entirely new edge then $P(e)$ is evaluated using Equation 10. If e is a new edge resulting from an existing edge being split into two distinct edges then $P(e)$ is computed using Equation 11 where s corresponds to the edge being split. The factor of 2 in the denominator is a consequence of the fact that we can make the assumption that within a given time step a single edge will only ever be split into two distinct edges.

$$P(\Delta E_t) = \prod_{e \in \Delta E_t} P(e) \quad (9)$$

$$P(e) = 1 - p_e^{|e|} \quad (10)$$

$$P(e) = \frac{1 - p_e^{|e|}}{(1 - p_e^{|s|})/2} \quad (11)$$

To illustrate the Delta Edge term consider Figure 2(a) which displays a graph containing a single edge which we entitle e_1. Now consider the situation where e_1 is split into two edges as displayed in Figure 2(b) which we entitle e_2 and e_3. In this case $\Delta E_t = \{e_2, e_3\}$ and $s = e_1$.

B. Location Term

In this section we describe how the location term $P(L_t|G_t, G_{t-1}, L_{t-1})$ is evaluated. This term models the probability that one traversed a path from $T_{G_t}^{G_{t-1}}(L_{t-1})$ to L_t in G_t. There are an infinite number of possible paths between these locations each having a different probability. In order to avoid the problem of attempting to evaluate the probability of all such paths, we use a simple proposal distribution which proposes the single path satisfying the following three assumptions. Firstly we assume that each edge in G_t is always traversed in the same direction. This assumption is justified by the fact that a monocular camera sensor is assumed and therefore a corridor traversed in opposite directions will appear visually distinct. In turn each traversal direction will be represented by a distinct edge in G_t which is always traversed in the same direction. Consequently each edge can be considered to have a head and tail. Secondly, we assume that when traversing from $T_{G_t}^{G_{t-1}}(L_{t-1})$ to L_t one always takes the shortest path where the length of a path is defined below. Finally we assume that when exiting one edge there is no restriction regarding which edges may be subsequently entered. To illustrate this proposal distribution consider the hypothesis (G_t, L_t), where the direction that each edge may be traversed is indicated by a dashed arrow, and the location $T_{G_t}^{G_{t-1}}(L_{t-1})$ of Figure 3. The graph G_t contains an upper, a middle and a lower edge. The proposed path corresponds to traversing the end of the upper edge followed by traversing the start of the middle edge.

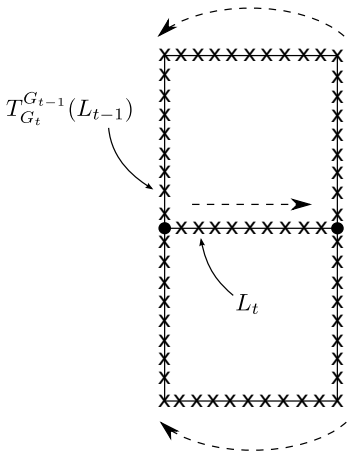


Fig. 3. The direction corresponding to each edge is represented by a dashed arrow. A path from $T_{G_t}^{G_{t-1}}(L_{t-1})$ to L_t is returned by the proposal distribution.

Given the proposed path from $T_{G_t}^{G_{t-1}}(L_{t-1})$ to L_t in G_t, we next compute the probability of this path. Toward this goal, we define a step as moving from one point in a graph G to a neighboring point, and the length of a path as the number of steps it contains. Let $T_{G_t}^{G_{t-1}}(L_{t-1}) = e_b^a$ and $L_t = e_d^c$; see section III for definitions of these terms. The length of the proposed path from $T_{G_t}^{G_{t-1}}(L_{t-1})$ to L_t in G_t, denoted $l\left(T_{G_t}^{G_{t-1}}(L_{t-1}), L_t\right)$, is evaluated using Equation 12. For example the length of the proposed path corresponding to Figure 3 is 3.

$$l\left(T_{G_t}^{G_{t-1}}(L_{t-1}), L_t\right) = |e^a| + d \quad (12)$$

The probability distribution $P(L_t|G_t, G_{t-1}, L_{t-1})$ is defined in Equation 13 where p_l is a specified model parameter and δ is a normalization constant corresponding to an infinite geometric series that may be computed easily. In this work our goal is to compute the most probable location L_t and not its actual probability. Therefore we only evaluate the probability distribution up to a constant term. The form of

this distribution is motivated by the fact that we assume the camera sensor is moving at close to uniform velocity and therefore the closer the length of a path is to 1 the higher the relative probability it will be assigned.

$$P(L_t|G_t, G_{t-1}, L_{t-1}) = \delta p_l^{l\left(T_{G_t}^{G_{t-1}}(L_{t-1}), L_t\right)-1} \sim p_l^{l\left(T_{G_t}^{G_{t-1}}(L_{t-1}), L_t\right)-1} \quad (13)$$

C. Data Term

In this section we describe the evaluation of the data term $P(Z_t|G_t, L_t)$ where Z_t is the appearance at time t. Let e_j^i be the value of L_t and $A(e_j^i)$ the corresponding appearance. This term models the probability that Z_t is similar to $A(e_j^i)$. It is evaluated using two distinct approaches depending on whether L_t is a previously visited or unvisited location. We now describe how each of these cases is evaluated.

1) Previously visited location:
We quantify Z_t and $A(e_j^i)$ using a 64-dimensional SURF feature based Bag-of-words representation [7]. We weigh words using *tf-idf* and measure the similarity between two bag of words using the L_1-score. Let s be the L_1-score between Z_t and $A(e_j^i)$. The data term is evaluated using Equation 14 which maps the score to a probability using a sigmoid function. The variables α and β are both parameters. This sigmoid function is commonly used by the machine learning community to map scores to probabilities [17].

$$P(Z_t|G_t, L_t) = \frac{1}{1 + \exp(\alpha s + \beta)} \quad (14)$$

2) Previously unvisited location:
Let H_t be the set of hypotheses at t as described in section III. This set can be partitioned into two sets H_t^v and H_t^u where the corresponding L_t is a previously visited or unvisited location respectively. We evaluate the data term for a previously unvisited location using Equation 15. The motivation for this expression is the fact that in the case of a previously unvisited location the corresponding set of hypotheses in H_t^v will have a low likelihood.

$$P(Z_t|G_t, L_t) = \max_{H \in H_t^v} (1 - \mathcal{L}(H)) \quad (15)$$

V. Multi-Hypothesis Tracking

In order to achieve robustness to noise a multi-hypothesis tracking framework was implemented. Toward illustrating how robustness is achieved consider again the environment in Figure 1(a) and the case where one begins at a, traverses the middle path to b, then traverses the lower path to a followed by the middle path to b. Assuming correct inference, the hypothesis with largest posterior probability at this time will correspond to that hypothesis illustrated in Figure 4(a). Now consider the case where one begins to traverse the upper path to a in Figure 1(a). The correct hypothesis in this case corresponds to that illustrated in Figure 4(b). However due to the fact that this hypothesis contains an additional vertex and a short edge relative to the hypothesis of Figure 4(a) it will have a corresponding low likelihood and in turn low posterior probability. Only when the corridor in question has been traversed further will an extension of this hypothesis, such as that illustrated in Figure 4(c), have a high corresponding posterior probability. This has the effect of achieving robust inference because the posterior probability of particular hypothesis will not become large unless it is supported by multiple observations. At each time step the set of hypotheses being tracked is pruned keeping only those w hypotheses with the greatest posterior probability. The ith hypothesis at time t is denoted (G_t^i, L_t^i).

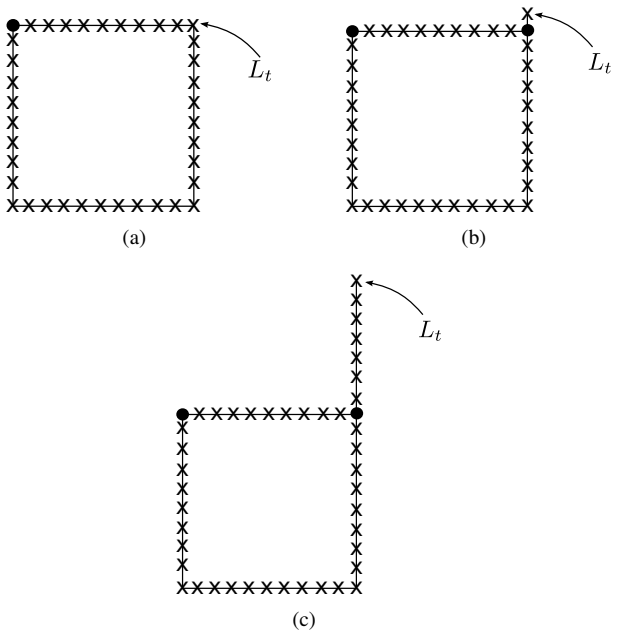


Fig. 4. The hypothesis in (a) is extended in (b) and subsequently extended further in (c).

Consider a hypothesis (G_{t-1}^i, L_{t-1}^i) and a sensor appearance measurement Z_t. Determining the hypothesis (G_t^j, L_t^j) given (G_{t-1}^i, L_{t-1}^i) and Z_t with greatest posterior probability, as defined by Equation 2, represents a difficult combinatorial optimization problem; the search space is that of all possible graphs and locations within these graphs. In order to overcome this challenge a proposal distribution is used which, using simple heuristics, proposes a small subset of all potential hypothesis. Although there is no guarantee that this set contains a hypothesis with maximum or close to maximum posterior probability, empirical results suggest that it generally does. The proposal distribution contains two steps which we describe in the following subsections.

A. Location Proposal

First a set of locations, denoted $P_L = \{L_t^0, \ldots L_t^m\}$, is proposed which correspond to the most probable locations in G_{t-1}^i at time t given (G_{t-1}^i, L_{t-1}^i) and Z_t. This set contains those locations in G_{t-1}^i where the corresponding appearance information $A(.)$, as described in section III, is most similar to Z_t. To allow fast querying an inverse index is used [7]. The set P_L also contains those locations adjacent to L_{t-1}^i and

a location corresponding to a previously unvisited location.

B. Graph Proposal

Next for each element L_t^j in P_L a corresponding set of graphs, denoted $P_G = \{G_t^0, \dots G_t^r\}$, is proposed. This set will contain the following four subsets if the graphs in question can be realized. We refer to L_{t-1}^i and L_t^j as previously unvisited locations if they were previously unvisited locations at times $t-1$ and t respectively.

1) If L_{t-1}^i and L_t^j are both previously visited locations:
   - The graph G_{t-1}^i.
   - G_{t-1}^i following the introduction of a single new vertex immediately after L_{t-1}^i. This has the effect of adding a new vertex and replacing an existing edge with two shorter edges. This case is illustrated in Figure 5.
   - G_{t-1}^i following the introduction of a vertex immediately before L_t^j. This has the effect of adding a new vertex and replacing an existing edge with two shorter ones.
   - G_{t-1}^i following the introduction of a vertex immediately after L_{t-1}^i and a vertex immediately before L_t^j. This has the effect of adding two new vertices and replacing each of the edges in question with two shorter ones.
2) If L_{t-1}^i and L_t^j are previously visited and unvisited locations respectively:
   - G_{t-1}^i following the introduction of a new vertex immediately after L_{t-1}^i where this vertex is at the tail of new edge containing L_t^j. This has the effect of adding a new vertex, replacing an existing edge with two shorter ones and adding a new edge of length one. This case is illustrated in Figure 6.
   - G_{t-1}^i following the introduction of new edge containing L_t^j. This has the effect of adding a new edge of length one.
3) If L_{t-1}^i and L_t^j are previously unvisited and visited locations respectively:
   - G_{t-1}^i following the introduction of a new vertex immediately before L_t^j. This has the effect of adding an additional vertex to the graph and replacing an existing edge with two shorter ones.
   - G_{t-1}^i following the introduction of a new vertex immediately after L_{t-1}^i. This has the effect of adding an additional vertex to the graph.
4) If L_{t-1}^i and L_t^j are both previously unvisited locations:
   - G_{t-1}^i following the insertion of L_t^j immediately after L_{t-1}^i. This has the effect of replacing an existing edge with a longer one.

After completion of the above location and graph proposal steps the set of proposed hypotheses is created by pairing each element G_t^l in each of the sets P_G with $T_{G_t^l}^{G_{t-1}^i}(L_t^j)$.

VI. Evaluation

In this section we present an evaluation of the proposed appearance-based SLAM method. This evaluation is performed with respect to the following dimensions. Section VI-A describes the datasets used within the evaluation. Section VI-B presents a quantitative performance evaluation of the proposed method relative to an existing state-of-the-art method. Finally, section VI-C presents an analysis of the proposed method in terms of execution time and number of hypothesis simultaneously tracked.

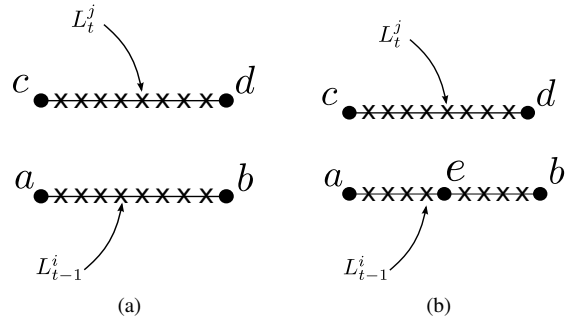


Fig. 5. The graph in (a) contains two edges and four vertices which are labelled a, b, c and d. In (b) the vertex e has been added immediately after L_{t-1}^i and splits the edge (a, b) into two shorter edges.

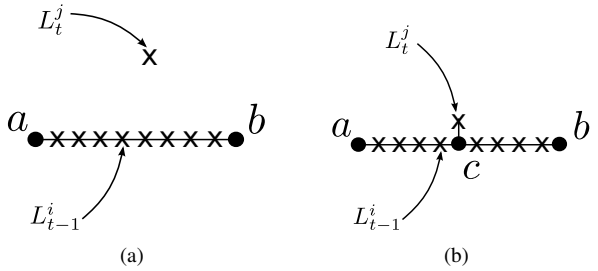


Fig. 6. The graph in (a) contains one edge and two vertices which are labelled a and b. In (b) the vertex c has been added immediately after L_{t-1}^i; this vertex is also at the tail of new edge containing L_t^j.

A. Datasets

In order to perform an evaluation of the proposed appearance-based SLAM method the New College dataset [18] and City Centre Dataset [2] were used. Each of these datasets was captured in a network space and therefore fulfils the assumption of such a space made by the proposed method. We now describe each of these datasets in turn.

The New College dataset contains a sequence of stereo images sampled at 20 Hz while the robot in question traverses a path of 2.2km. In this work we only consider the right stereo images following their conversion to grey-scale. This sequence was down-sampled by keeping only every 20th image to give an effective sampling rate of 1 Hz. A subset of this sequence of images, necessary for construction of a dictionary for use in a visual bag-of-words representation, was selected as follows. We down-sampled the original sequence keeping every 20th image using an offset of 10 frames relative to the down-sampling described above. Finally, we selected 25% of these images at random.

The City Centre dataset contains a sequence of images captured using a camera mounted on a pan-tilt which collects images to the left and right of the robot. An image is captured every 1.5m travelled, which is determined using odometry, for a distance of 2.0km. In this work we only consider the left

images following their conversion to a grey-scale. A subset of this sequence of images, necessary for construction of a dictionary, was selected by choosing one third of the images at random.

The performance of an appearance-based SLAM method is generally quantified in terms of precision and recall with respect to loop-closure detection. In order to compute these metrics a corresponding ground truth dataset was constructed for each dataset as follows. For a given sequence of images, for every fifth image in that sequence we determined a set containing all intervals in the sequence which corresponded to valid loop-closures.

B. Comparison to DBoW

In this section we present an evaluation of the proposed appearance-based SLAM method relative to the state-of-the-art method of DBoW [7]. Within this method individual loop closures are detected using a visual bag-of-word followed by a geometric verification. As discussed in section II, this method implicitly assumes a network space by rejecting loop-closures if they do not occur in a sequence.

DBoW has the following parameters: α which is a threshold on visual similarly used to determine whether or not a loop closure has occurred and k which represents the length of a sequence of loop closures which must occur in order for the final loop closure in the sequence to be considered valid. In our evaluation we assigned k a value 3 as recommended by the original authors. The parameter α was varied in order to generate precision-recall curves.

The proposed appearance-based SLAM method has the following parameters: p_v (defined in Equation 4), p_e (defined in Equation 5), p_l (defined in Equation 13), α and β (defined in Equation 14) and w (the number of hypothesis tracked as defined in section V). These parameters were optimized using cross validation and the parameter β was subsequently varied in order to generate precision-recall curves.

Both the proposed appearance-based SLAM method and DBoW require the provision of a dictionary for use in a visual bag-of-words representation. For each dataset a corresponding dictionary containing $100,000$ words was constructed using the corresponding sets of images described in section VI-A. The same dictionary was used for both the proposed method and DBoW. This represents an important point because, as a consequence, relative performance is a function of solely the inference methods in question.

Figures 7 and 8 display the precision-recall curves for the New College and City Centre datasets respectively. The maximum recall with 100% precision achieved by the proposed method and DBoW on the New College dataset was 86% and 67% respectively. Toward illustrating this high recall achieved by the proposed method, consider Figures 9(a) and 9(b) which display the corresponding visual odometry and visual odometry overlaid with detected loop closures respectively. It is evident from these figures that the proposed method detected loops closure with high recall. The maximum recall with 100% precision achieved by the proposed method and DBoW on the City Centre dataset was 93% and 56% respectively. In summary, the proposed appearance-based SLAM method significantly outperforms the method of DBoW on both datasets considered.

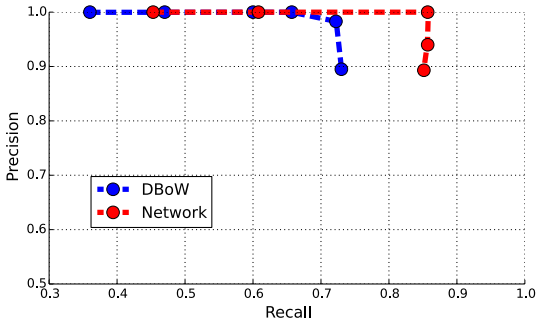


Fig. 7. Precision/Recall curves for the New College dataset.

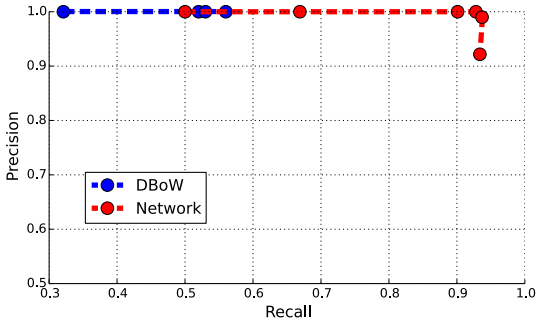


Fig. 8. Precision/Recall curves for the City Centre dataset.

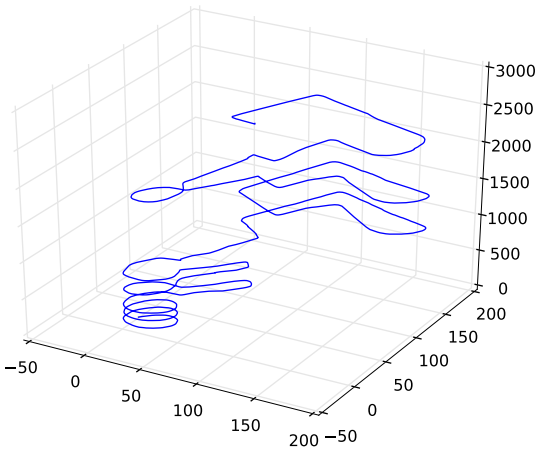
C. Execution Time and Number of Hypothesis

In this section we examine the relationship existing between w (the number of hypothesis tracked), the performance in terms of precision and recall, and execution time. The proposed method was implemented in C++ and runs on a single CPU core. All experiments were performed using a laptop containing an Intel Core i7 2.70GHz processor. For different values of w, Table I displays the corresponding execution time and maximum recall achieved for 100% precision on the New College dataset. A maximum recall is achieved when w is assigned a value of 20; no increase in recall is achieved through the assignment of a larger value. A significantly lower recall is achieved when w is assigned a value of 1; this result justifies the use of a multi-hypothesis tracking framework. In Table I execution time in expressed in terms of mean frequency per second, or Hz, over the entire dataset. It is evident that w and Hz are inversely related.

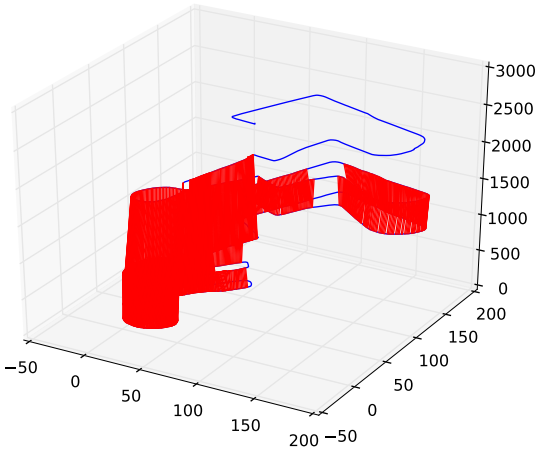
TABLE I

NUMBER OF HYP. w VS. MAX RECALL FOR 100 % PRECISION AND HZ.

w	1	5	10	15	20	25
Recall	0.24	0.43	0.54	0.71	0.86	0.86
Hz	8.0	3.1	1.3	0.7	0.4	0.3



(a)



(b)

Fig. 9. Visual odometry and visual odometry with detected loop closures represented by red lines are displayed in (a) and (b) respectively. The high density of red lines indicates a high recall.

VII. Conclusions and Future Work

SLAM is regularly performed in network spaces where priors may potentially be placed over the structure of the space. In this work we propose an appearance-based SLAM method which explicitly models the space as a network and uses this model as a platform for the placement of such priors. Specifically, we place a prior over the space of networks such that those networks of lesser complexity are assigned a higher probability. Relative to some existing SLAM methods, which implicitly assume a network space and place priors over its structure, this approach allows a more formal placement of priors. This method is implemented within a multi-hypothesis tracking framework. Results achieved on two publicly available datasets demonstrate that the proposed method achieves high precision and recall with respect to loop closure detection and in turn outperforms a current state-of-the-art appearance-based SLAM method.

Despite these achievements there exists much potential for future expansion and improvement of the current method. One of the major disadvantages of using a multi-hypothesis tracking framework is that it increases execution time. However since distinct hypothesis are independent, this issue could be addressed through the use of a parallel computing paradigm as opposed to the serial computing paradigm which is currently used.

In this work we place a prior over the complexity of the network structure. There exists potential for the placement of priors over other aspects. For example, in most network spaces it is uncommon for more than four edges to meet at a single point. This fact could be exploited through the placement of an appropriate prior.

Acknowledgements

This work was supported in part by a European Marie Curie International Outgoing Fellowship.

References

[1] D. Rosen, M. Kaess, and J. Leonard, "Rise: An incremental trust-region method for robust online sparse least-squares estimation," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1091–1108, Oct 2014.

[2] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[3] M. Kaess, A. Ranganathan, and F. Dellaert, "isam: Incremental smoothing and mapping," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.

[4] S. H. Chan, R. V. Donner, and S. Lämmer, "Urban road networks-spatial networks with universal geometric features?" *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 84, no. 4, pp. 563–577, 2011.

[5] M. Barthélemy, "Spatial networks," *Physics Reports*, vol. 499, no. 1, pp. 1–101, 2011.

[6] P. D. Grünwald, *The minimum description length principle*. MIT press, 2007.

[7] D. Galvez-Lopez and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[8] M. Milford, "Vision-based place recognition: how low can you go?" *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 766–789, 2013.

[9] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.

[10] A. Ranganathan, E. Menegatti, and F. Dellaert, "Bayesian inference in the space of topological maps," *Robotics, IEEE Transactions on*, vol. 22, no. 1, pp. 92–107, 2006.

[11] W. Maddern, M. Milford, and G. Wyeth, "Cat-slam: probabilistic localisation and mapping using a continuous appearance-based trajectory," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 429–451, 2012.

[12] R. Salas-Moreno, B. Glocker, P. Kelly, and A. Davison, "Dense Planar SLAM," *IEEE International Symposium on Mixed and Augmented Reality*, 2014.

[13] Y. Latif, C. Cadena, and J. Neira, "Robust loop closing over time for pose graph slam," *The International Journal of Robotics Research*, p. 0278364913498910, 2013.

[14] S. Tully, G. Kantor, and H. Choset, "A unified bayesian framework for global localization and slam in hybrid metric/topological maps," *The International Journal of Robotics Research*, p. 0278364911433617, 2012.

[15] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conference on Artificial Intelligence*. Edmonton, Canada: AAAI, 2002.

[16] Gilbert Strang, *Calculus*. Wellesley-Cambridge, 1991.

[17] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in large margin classifiers*, 1999.

[18] 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*, vol. 28, no. 5, pp. 595–599, 2009.