A Topological Path Layout for Autonomous Navigation of Multi-sensor Robots

F. Ferreira and V. Santos  
Department of Mechanical Engineering  
University of Aveiro, 3810-193 Portugal  
Email: cfferreira,vsantos@mec.ua.pt

J. Dias  
Institute of Systems and Robotics  
University of Coimbra, 3030-290 Portugal  
Email: jorge@isr.uc.pt

Abstract—This article describes the requisites for the specification of the topological layout of a path within an environment. Such a specification finds application in tasks requiring navigation using partial information and can also be applied to tasks requiring coordination and communication between robots or between users and robots. The topological specification of the path is achieved by representing a sequence of sampled views as the observation states of a Hidden Markov Model. A left-to-right Hidden Markov Model is adopted for its appropriate simplicity, albeit with modifications to take into account changes that might occur in the environment. Robust View matching is achieved through the use of Bernoulli Mixtures to model the features extracted from all the distinct sensors of a robot platform. Local image features (SIFT) and multiple types of features from a 2D laser range scan are all converted into binary form and integrated into a single, binary, Feature Incidence Matrix (FIM).

Index Terms - Bernoulli Mixture model, Binary data, Dimensionality reduction, Robot Localization, Topological Representation, Hidden Markov Model.

1. Introduction

Robot localization and Navigation algorithms have come of age with the demonstrable success of platforms such as the museum tour guides[1], driver-less cars, etc. Despite the large amount of successful work that is done in creating maps and matching sensor data to individual locations in maps (that are constructed or under construction), very few methods make use of the sequence of sensory data that is expected to guide the robot [2], [3].

Many of the works that use topological maps attempt to manage the problem of a large number of features using 'automatic' feature extractors. These methods either work offline (after sampling the environment at fixed intervals of time or distance) or on line (by selecting those landmarks that correlate poorly with past landmarks). Examples of the former include [4] while an introduction to the latter approach can be seen in [5].

In the case of 'on line' methods like [5], where landmarks include the entire robot perception at a point, landmark selection means choosing which data to keep in the topological map and which data to lose without compromising on the topological map's usability.

In this article, we have tried to answer the question What is a compact representation of the Topological Path through an environment?

1.1. Defining a Topological Path.

The short answer to this question is: A Bayesian Network, in our case an HMM, represents a compact representation of the Topological Path.

Any probabilistic representation of a robot environment must represent the Joint probability either implicitly or explicitly, [6]. HMMs represents the joint and increasingly large and complex observation and transition distributions in terms of local, conditional, distributions. Patterns in the joint distribution are captured in terms of short-term correlations between sequential Views.

In the case of sensor data collected along the path that the robot takes, with the increase in the size of the Path to be represented and with the addition of more sensors and of more sensor features, this joint probability (of being in a place and observing something) tends to become unmanageably large.

Representing the environment as a Hidden Markov model (indeed, as a first-order Dynamic Bayesian Network) reduces the problem of maintaining the joint distributions to combinations of two, separate, marginal distributions: the transition probabilities and the emission probabilities. This results in a much more compact representation as compared to probabilistic, grid-based, Geometric Maps where the joint probability of the robot being within each grid must be maintained. The complexity of grid-based maps increases exponentially with the size of the map and various methods must be utilized to reduce the complexity of the localization process.

Also, in the specific problem of localization in environments, we do not expect places in the environment to simply switch appearances among themselves. This particular property is one of the key reasons why the utilization of a Markovian assumption is most appropriate for most localization applications.

Therefore, what distinguishes a topological map is the ability to separate the conditional probabilities that, when multiplied provide the joint probability of the map(with the robot included). The nodes of the topological map represent convenient places where this conditional probabilities may be expressed. It matters less, how and where these nodes are located. Some mapping methods prefer to situate the nodes for ease of detection[7], others prefer to specify nodes more consistent with an approach that explores and segments the environment[8], [9] or any other criteria.

Thus, it is not relevant to debate whether the topological map is a high-level representation [8] or not. The topological map offers us the chance to break up the joint probabilities
that express a map into two or more sets of conditional probabilities and express it as a graph.

Having delved a little into the definition of a Topological Map, we define the Topological Path as being a simple graph that is a subset of the complete graph that is the Topological Map as seen in Fig. 1. This Topological Path then becomes a sequence of instructions that the robot must execute in order to do something useful.

Fig. 1: Leading the robot along a Topological Path in the larger Topological Map. The sequence of sensory information that is gathered while travelling along the Topological Path can, in many situations, be sufficient to successfully localize and navigate along that Path.

1.2. Localization along a Topological Path

Guiding a robot by providing only a sequence of expected sensor data [as against maps] seems, to us, to open a whole new set of possible applications which include more intuitive interaction with humans and navigation in environments of very large scale. The concept opens up new problems namely in the question of how the sequence is to be represented and how is data association to take place.

Navigation through the use of a sequence of sensor data requires the specification of that sequence. We have called the sequence of 'Views' that the robot attempts to repeat as the 'Reference Sequence'. In a trivial way, this Reference Sequence can be obtained by sampling the environment as the robot moves.

The sequence of observations we are currently seeing, and wish to match against the Reference Sequence is called the Query sequence or the Working Sequence. These terms have been used interchangeably in this chapter and in the rest of this document.

The Reference Sequence, as a structure, can yield information that is more than the sum of the information contained in each of its individual Views (Fig.2a); information that is gleaned from the 'order' in which the Views occur. We call this the 'contextual' information of the Reference Sequence, shown in Fig. 2b.

The additional information that the contextual information brings affects the place recognition in two ways:

Fig. 2: Place recognition can be performed without and within the context of the Reference Sequence. The latter case utilises a recent history of the perception to improve the Place recognition.

1) Firstly it has the capability of completely rejecting inconsistent sequences of place recognition.
2) It also provides an arguably consistent framework to represent the prior probability for the place recognition itself.

The work that we report in this article seeks to contribute to this discussion on the amount of information required for effectively performing localization.

It is our opinion that knowledge of the sequence of observations that are expected to be found along a particular path in the environment is usually sufficient for localization for some applications and environment. In order to get from a point A to another point B, the robot only needs to be told of the sequence of observations it will sense as it proceeds from A to B.

If we define this sequence of observations found on the path from A to B as the Reference Sequence, then, we ask; is a Reference Sequence of observations along a topological 'path' enough to allow the robot to retrace this path? How should the Reference Sequence be represented so as to extract the most information from the 'context' of one of more matches?

Three distinct questions must be answered if a Reference Sequence is to substitute a map. These are:

1) How to we represent a single View in the Reference Sequence and match any pair of Views? Our particular answer to this question has been to integrate binary features using a Bernoulli mixture model in order to perform place recognition.
2) How to we use the context of the Reference Sequence for localization? While place recognition can be performed
using data at any one instance or position, it might not be enough to disambiguate between similar places lying on the sequence. We have used Hidden Markov Models (HMMs) to 'align' the observations during localization with sections of the Reference Sequence.

3) How do we get the most compact Reference Sequence? A trivial Reference Sequence involves sampling regularly the environment along a path. This may not result in the most compact Reference Sequence for that path. Some procedure could be adopted to 'optimize' the Reference Sequence created from samples of observations. Such a procedure would allow the creation of compact Reference Sequences that still allow for acceptable localization.

Over the next few sections we attempt to model the Reference Sequence in terms of a well known and often utilised variant of a Bayesian Net, the Hidden Markov Model. In the next section, procedures to match a single view with each view in the Reference Sequence and a special layout of the Markov Model to represent the transition between places will be presented along with results. Finally, in section 4, we shall discuss proposals to create compact representations of the Topological Path that still contain enough information for localisation.

2. Using HMMs to accumulate inferences

One of the earliest application of Bayesian Networks in the form of HMMs was to solve the problem of speech recognition. Rabiners tutorial [10] continues to provide a good introduction to the the problem of speech recognition (and to the application of HMMs in general). This area has also seen the application of a number of modifications to the vanilla HMM to account for the complexity of the problem and the robustness that is required of any successful application. Modifications include the use of continuous variable duration HMMs [11]. The recognition of passages of music or the retrieval of the complete music using a shorter passage is another extension of this problem [12].

Recognising what is written by one or multiple persons hand is a difficult problem because of the ambiguity that is introduced in the actual writing procedure. Besides spelling mistakes and the need to recognise a persons own style of writing, there is the difficult problem of identifying where a character ends and where another one begins. Besides methods developed originally for speech recognition, one of the techniques that have found considerable success in the recognition of handwriting are the so called Variable Duration HMMs that allow for increases in the distance between hand written characters[13] and extensions of the same to more complex Bayesian Networks [14]. These techniques have been refined over the years to allow for larger dictionaries and lesser restrictions on the handwriting styles that can be recognised [15].

The importance and sheer scale of the Human Genome Mapping project provides just one example of the importance of the application of HMMs in Biology. The problem of matching described earlier is known in this field, of Bioinformatics, as protein sequence alignment. While there have existed alternatives to the HMM profile-based methods such as pairwise matching and Neural-Network based methods, HMMs 'now provide a coherent theory for profile[matching] methods'[16]. With the human genome completely sequenced, Profile based (HMM) methods have been often used to compare a DNA sequences of as yet unknown function with others having known function [17].

The problem of sequence alignment of protein sequences actually bears much resemblance to the problem of robot localization within a sequence of observations. These are enumerated below.

- A protein sequence, as in the case of our Reference Sequence, is seen as a first-order Markov Chain where only the symbol sequence (corresponding to the view sequence) is visible. The chains are modeled as directed(left-to-right) graphs, where the transition between 'consensus nodes' are represented using a transition probability Matrix.
- Profile HMMs allow the incorporation of information regarding the predisposition of protein sequences to contain spurious subsections (called insertions) as well as missing proteins (called deletions). Given that the Reference Sequence is composed of sampled views of the environment, insertions are expected in between the views in the Reference Sequence. Additionally, limitations in the perception process, incorrect executions of the required behaviours, and possible changes in dynamic environments might result in an unmatched (reference-sequence) view.
- Suited to short-term correlations in the views [16] rather than long-term correlations. This would be analogous to applying the HMMs to detect localization using short-term correlations but not to the problem of loop-closures.
- As in the case of the localisation with the Reference Sequence problem, the profile HMM method forces us to assume what would be the emission probabilities for the insertion states. We assume that the insertions have the same emission probability as the views in the Reference Sequence as a whole.

In the field of Robot Localization itself, probabilistic methods have become the de-facto standard given their flexibility and the development of Bayesian Inference methods that have popularized the use of Bayesian Networks. Although it is quite impossible to mention even the most important we mention a few works because of their relation to our work [18] [19] [20] [21] [22].

2.1. The Reference Sequence of Sampled Views

As described earlier, a Reference Sequence can be created by sampling the environment according to some sampling plan. The critical point of any successful plan is that such a sample plan should be suited to the type of environment, the nature of the sensor features that are gathered and the frequency of data acquisition. The sampling plan must simply
not miss out the views that are critical (essential) for the completion of the mission.

Sampling plans, in general can be based on one of the following:

- Time-Based Sampling
- Distance-based sampling
- Sensed-Event based sampling

Time-based sampling is the usually the easiest to implement and does not require any special hardware or sensors. Usually, such a system will take sensor data from multiple sensors at regular intervals, usually corresponding to the slowest sensor (sensor with the lowest frequency of acquisition).

Distance-based sampling refers to a sampling plan in which new sensor data is triggered by having an odometry system detect that a certain distance or rotation has been covered.

A Sensed-Event based sampling is possible only when the system is capable of detecting or tracking high-level, identifiable features. In such a system, when the robot successfully segments such a high level feature for the first time or rediscovers it at a new position, sensor acquisition is triggered. Such a sampling plan cannot be used with robots equipped only with an appearance based localization system as such a system is usually not meant to detect specific, pre-defined, high-level features.

Needless to say, in order to be able to repeat the actions required to get to the goal, the environment must also be sampled whenever the robot changes the behaviour that it executed.

3. Modeling the Reference sequence as an HMM

The traversal of the robot along the sampled views can be modeled as a Finite State Machine. The order in which the views were collected in the Reference Sequence, should, if all goes well be repeated as the robot retraces is path on a latter occasion. Since we know only one route between any pair of places the action/behaviour that must be taken is the one recorded in the Reference Sequence. Any other action will take us to somewhere else (where, we do not know!).

In section 3.1, we describe how Place recognition is actually performed using each view of the Reference Sequence. In subsection 3.3, we show how the Reference Sequence is created. We shall then describe the structure that we have adopted for the HMM so that the a standard Dynamic Programming algorithm, the Viterbi algorithm, can be used to match the observation in the context of the Reference Sequence.

3.1. Single View Feature Representation

The idea of converting continuous or discrete-valued features into binary form is not new [23],[24]. Converting features into binary form presents a straight-forward integration of disparate features irrespective of the nature and physical model of the sensor. An integration of such disparate features using a filter-based method would require physical models for each type of feature. This usually is not a feasible task; note for example how would one integrate the presence of a intensity image feature (from an single uncalibrated camera) with, say, the distance from a prominent landmark detected using a laser range finder? Additionally, the ability of binary data to represent both Quantitative (Continuous, Discrete) and Categorical (Ordinal, Nominal) variables is extremely helpful to represent features of different types.

In Fig. 3a we show how features are extracted from the different sensors and converted into a binary form to create a single View. The Views are collected from all along the sequence to generate the Feature Incidence Matrix (FIM) shown in Fig. 3b.

When a View form the Query Sequence must be matched to the Reference sequence a straight-forward comparison with the Reference Sequence is not desirable, since the features in the FIM will, most probably contain strong, unknown correlations. To perform this matching we have use a mixture of multi-variate Bernoulli distributions to model the FIM. When the View from the Query Sequence is compared with the Bernoulli Mixture Model, the problem of the correlation between features is reduced or eliminated.

The robotic platform that was employed is shown in Fig.4. In the course of the experiments described in this article we have used only the two Cameras 1 and 2. The features that were employed are the popular and robust SIFT features [25]. The method for extracting the features is described in [26].

3.2. Formulation of the Bernoulli Mixture-Model

Mixture models assume that there exist a finite number of parametric distributions which, when mixed together in a particular proportion, result in a distribution that best describes the data we wish to characterize. In this case we model any observation $V^{obs}$ as a vector of binary features {0, 1}^N which is obtained from a particular mixture of Bernoulli distributions, as in (1).

$$P(V^{obs} | \Theta) = \sum_{i=1}^{M} \alpha_i P_i(V^{obs} | \Theta_i)$$ (1)

where $\Theta$ denotes the parameters of the distribution of the views that compose our Mixture Model. These parameters include
Fig. 4: Image of the Robot Platform equipped with two Cameras and one Laser Range Finder. In the work described in this article, only features from Camera 1 and Camera 2 were utilised.

the $M$ component vectors, the $\Theta_j$s, and the proportions in which these are mixed, the $\alpha_j$s. Each $\alpha_i$ represents the prior probabilities of the component $i$ in the mixture model, subject to the constraint $\sum_j \alpha_i = 1$. The term $P_i(V_{obs} | \Theta_i)$, can be determined using (2) where each $\Theta_i$ is a multivariate vector of Bernoulli probabilities each of whose $N$ components indicate the probability of success for a particular feature.

$$P_i(V_{obs} | \Theta_i) = \prod_{j=1}^{N} \Theta_{ij}^{V_{obs}^j} (1 - \Theta_{ij})^{(1 - V_{obs}^j)} \tag{2}$$

To obtain the mixture parameters that explain a particular FIM $\mathcal{V}$, consisting of $K$ observations it is assumed that the views are independent and the likelihood of the mixture satisfying the FIM is expressed thus (3).

$$P(\mathcal{V} | \Theta) = \prod_{k=1}^{K} P(V_k | \Theta) = \mathcal{L}(\Theta | \mathcal{V}) \tag{3}$$

The optimisation task to find the mixture that best explains this $\mathcal{V}$ can be expressed as in (4), i.e. find the value of $\Theta$ that best satisfies the distribution of features in $\mathcal{V}$.

$$\Theta^* = \arg\max_{\Theta} \mathcal{L}(\Theta | \mathcal{V}) \tag{4}$$

The preferred method of solving the Mixture Model problem is the Expectation Maximisation algorithm. McLachlan ([27], page 19) states '...It will be seen that conceptualization of the mixture model (hidden data + component distributions)... is most useful in that it allows the Maximum likelihood estimation of the mixture distribution to be computed via a straightforward application of the EM algorithm.' The EM method applied to the Mixture problem assumes that the data is only partially available. It becomes fully known through the use of a vector of coefficients denoted henceforth as $Z$, called the 'missing data' or the 'hidden data' or still the 'unobserved data'. If we introduce this $Z$ to expression (1) we can now express the likelihood of the observations given the entire data as in (5) and further simplify it to (6).

$$\mathcal{L}(\Theta | \mathcal{V}, Z) = \sum_{k=1}^{K} \sum_{i=1}^{M} z_{ki} \log(\sum_{i=1}^{M} \alpha_i P(V_{obs} | \Theta_i)) \tag{5}$$

$$\mathcal{L}(\Theta | \mathcal{V}, Z) = \sum_{k=1}^{K} \sum_{i=1}^{M} z_{ki} [\log(\alpha_i) + \log(P(V_{obs} | \Theta_i))] \tag{6}$$

The EM algorithm proceeds in two stages: the Expectation stage attempts to reach the best value for the missing data $Z$, by keeping the parameters of the Mixture model constant(7), while the subsequent Maximization stage attempts to optimise the components and mixing parameters themselves by using the values of the 'missing data' obtained in the expectation step just performed (8), (9). The method then alternates between the two steps until some termination criteria is satisfied.

$$z_{ki} = \frac{\alpha_i P_i(V_k | \Theta_i)}{\sum_{j=1}^{K} \alpha_j P_j(V_k | \Theta_j)} \tag{7}$$

$$\alpha_i = \frac{\sum_{k=1}^{K} z_{ki}}{K} \tag{8}$$

$$\Theta_i = \frac{\sum_{k=1}^{K} z_{ki} V_k}{\sum_{k=1}^{K} z_{ki}} \tag{9}$$

Fig. 5: Sample binary FIMs, where the FIM at left shows views sampled from two populations. The FIM at right shows the same views with some added noise (the binary features in some views have been flipped). The black squares (value of 1.0) indicate the presence of features. $k$ denotes the index of the views.

To obtain a feel for the Bernoulli mixture model, two toy FIMs are presented in Fig. 6. The FIM at left is made up of what we can see are two distinct sets of features. The FIM at right contains the same two distinct sets of features but a few of the bits have been flipped.

After running the EM algorithm for mixture models, with two and four components respectively, we end up with the components as shown in Table I and Table II and the mixture
coefficients as shown in Table III and Table IV, respectively. As can be seen, most of the layout of the noisy FIM is explained by components \( \Theta_1 \) and \( \Theta_2 \) (the original components in the noiseless data) and the distribution of these components is quite similar to the corresponding components for the noiseless FIM.

Mixture models used for classification make use of both, the Mixture parameters and the posterior probabilities over the components, the \( Z \) are used to evaluate the likelihood in the space of the views in the Reference Sequence as in (10) where \( P(k) \) represent the prior probabilities on each index \( k \).

\[
P(k|V^{obs}) = \frac{\sum_{j=1}^{M} P(k)z_{kj}u_j P(V^{obs}|\Theta_j)}{\sum_{k=1}^{K} \sum_{j=1}^{M} P(k)z_{kj}u_j P(V^{obs}|\Theta_j)}
\]

(10)

The Maximum Likelihood Estimation approach is used to obtain the matching view, the index \( k^* \), in \( V \) that best describes the view to be matched, \( V^{obs} \).

\[
P(k = k^*|V^{obs}) = \max_k P(k|V^{obs})
\]

(11)

3.3. Introducing the Hidden Markov Model

What is the best way of representing the Reference Sequence for the purpose of localisation?

Consider that a sequence of sensor data is obtained when an action is performed. This sequence can be seen as a signature of some signals, Fig. . This signature can be defined in time or in some other, more appropriate, variable derived from the time.

As mentioned in the earlier section, the place is not directly visible. We attempt to match the current observation with each view in the Reference Sequence (using the Mixture Model), to infer the place at which the robot is situated. To improve the quality of the localization, specifically, to include the information that the order of the views in the Reference Sequence, we have tried to model the Reference Sequence as a Markov Chain. Localization using the context of the Reference Sequence has been modelled as a Hidden Markov Process where the actual position occupied by the robot is modeled as the hidden states and the results of place recognition, as described in the earlier section, are modeled as the observation.

The Reference Sequence assumes that a behaviour executed at each view propels the robot to the next view in the Reference Sequence. This is expressed as a Markov chain as in Fig. 8. Actions other than the one executed in the Reference Sequence will result in the robot seeing things differently as compared to the Reference Sequence. In the case of robots in dynamic environments this problem is made worse since the original behaviour from the Reference Sequence may no longer take the robot to the next view or, worse, the behaviour may no longer be valid. A view-based localisation for a robot must, at least, recognise that the robot is lost.

In the case of mobile robot localisation the signals could be represented in terms of the distance traveled - a noisy quantity that is not always controllable in certain environments.

We utilise the notation used in Rabiner's tutorial on HMMs for speech recognition, [10]. The parameters \( \lambda \) of the HMM are specified as in (12), where \( N \) corresponds to the number of states, \( M \) the number of observations, \( \pi \) represents the initial probability on the states, the \( \alpha_{ij} \) correspond to the transition probabilities between a pair of states \( i \) and \( j \) and \( b_i(n) \) represents the probability of viewing symbol \( n \) at state \( i \). Rabiner [10], also specifies three basic problems for HMMs, as reproduced in Fig. 6. We are interested in the second type of problem: what is the probable path (defined in terms of views in the Reference Sequence) that the robot took, given what the robot has seen so far?

\[
\lambda = \{N, M, \{\pi_i\}, \{a_{ij}\}, \{b_i(n)\}\}
\]

(12)
3.4. Modifying the Markov Chain for localization

In our case, for localisation, since we know only one route between any pair of places the action/behaviour that must be taken is the one recorded in the Reference Sequence. Any other action will take us to somewhere else (where we do not know!). As a result the HMM depicted in Fig. 8 will be modified to Fig. 9. It is important to note that, if inference is to be made across multiple actions or behaviours, it will be reflected in terms of an increased complexity of the graph in Fig. 9.

Fig. 7: A classic representation of a HMM for localization in a Reference Sequence showing the [same] action that will propel the robot from one state to another over a total of 4 observations.

Keeping the discussion of the last section in mind we have modeled the Reference Sequence, for localisation purposes into another Markov chain, with some additional Places, a fragment of which is shown in Fig. 9. The Reference Sequence consists of K sampled views in all. To this discrete set K, are added K - 1 'Lost' Places that so that the set of values that the state can take is now described in the discrete set M which includes the K sampled Places and the K 'Lost' Places (there is a lost place before the first state too). This representation is similar to the model use for profiling gene sequences[28]. An example of a 'profile HMM' used in such application is shown in Fig. 10. Our graph lends itself to the application of profiling HMM to the robot localization problem. In our case the lost Places correspond to the 'deletion' nodes in the profile HMM and hold special meaning as described below:

1) Indicating the last position at which robot was localized and directly indicating the probability of the robot being lost.
2) Taking into account the fact that the robot might see non-sampled views.

1) The observation model, the b_t(n): We perform the observation using the mixture model, described in the earlier section. The components of the mixture model are estimated and the current view is compared to these components.

For the observation model of the HMM, the observations 'O_t(n)' at each an every state of the HMM are defined in the same space as the sampled views (13) from the Reference Sequence. This is not a property of HMMs but rather a result of our adoption of the Bernoulli Mixture model for matching

Fig. 8: This figure models the 'K' Places in the Reference Sequence as a Markov Chain with transitions that reflect the order in which these Places were sampled. The sequence represents all the transition knowledge that is available of the environment and represents a Topological path in that environment.

Fig. 9: This figure shows a fragment of the 'complete' Reference Sequence for localization as a Markov Chain with lost places inserted within the original Reference Sequence shown in Fig. 8. The robot is now assumed to start out at the Place 'P_{Lost,0}', where, it has never localized in the Reference Sequence. The dotted lines indicate the transitions to each of the Places in the original Reference Sequence which have not been drawn to avoid cluttering the figure.

Fig. 10: The figure describes a profile HMM [28]. The lower-most layer of states correspond to the 'main' states, the diamond-shaped states represent the 'insertion' states and, the circles represent the 'deletion' states. Such a sequence is more appropriate to model a Markov Chain in which additional information is available on the probabilities of 'missing' and of 'additions' occurring at specific positions. Such a problem is somewhat analogous to our representation of the Reference Sequence where variations object occlusions and variations in robot speed may result in some observations being missed out or new, previously unseen views being obtained.
any pair of views. Since the posterior probability for the Bernoulli Mixture Model is calculated in the 'space' of the views of the Reference Sequence and not in the space of the components, the observations have the corresponding number of events as in $K$.

Also, since the environment is sampled, during the localization phase, not every observation might be assigned to a place in the Reference Sequence. Thus, when the robot is between places that were sampled earlier, it sees an un-defined view. We call this undefined view the 'Lost View' and any place generates this view is defined as a 'Lost Place' Fig. 9. In the absence of any information, we arbitrarily define the observation probability over the set $K$ at any Lost Place as an uniform distribution. This statement is made on the assumption that the distribution of the features is is maintained throughout the environment. In practice the probability emission matrix will have non-zero probabilities off the diagonal (for the original places). In addition it will have to be regularised (probabilities made non-zero) to avoid over fitting to the Reference Sequence.

$$O_{i}(n) \in K$$ (13)

(a) The Emission Probability Matrix.

(b) The Probability Transition Matrix.

Fig. 11: The figure at left represents the Observation Probability (emission) matrix for our HMM. The observation from the lost Places is now expected to result in a LostView. This additional observation is 'seen' whenever the entropy of the posterior probability over the views of the mixture model is greater than some threshold. At right, is an image of the Transition Probability matrix for the hidden states of the HMM (the complete Reference Sequence). The values of the transition probability are obtained from a model of the robot motion as explained in subsection 3.4.

2) The transition probabilities on the places, the $a_{ij}$ s: The transition probabilities reflect on the probability of moving from one place to another. The transition probability is always conditioned on the behaviour or behaviours being executed. If we allow each place to transition to the neighbouring places and to the respective lost place we get a transition probability matrix which can be represented in terms of Fig. 11, where white represents a transition probability of 0, black a transition probability of 1 and grey values represent the continuum of values in 0,1.

A robot motion model that applies a probability on every view, as a function of the estimated amount of travel is adopted . We refer to the probability on each view $k$ as $P(k)$ and the value of the probability on the view at any time $t$ is defined as $P(k^t)$.

The distribution displaces the current probability distribution from a Lost Place to the remaining Views in the Reference Sequence according to (14), where $d(i-j)$ is the distance from each original place corresponding to the Lost place and $d_{odo}$ is the estimate of the distance travelled. The transition probability from any of the original places in the Reference Sequence is uniformly split between the next original place (to the right) and to the corresponding Lost Place. The one-step transition probability from a Lost Place to another Lost Place is zero.

Representative images and Laser scans from the Query/Working sequence are shown in Fig. 14.

$$P(i|d_{odo}) = \frac{\sum_{j=1}^{K} P(i^{t-1}) \times d(i-j)/(2 \times d_{odo})}{\sum_{i=1}^{K} P(i|d_{odo})}$$

if $|d(i-j)| < d_{odo}$

$$P(i|d_{odo}) = 0 \text{ otherwise}$$ (14)

This transition probability also reflects the prior probability that must be applied to the place recognition given that the robot is lost.

3) The initial Probabilities State: It is natural that the probability at the beginning of the localization be concentrated in the first lost Place, before the first state. Thus, the first hidden state is always matched to the Lost Place, $P_{lost,0}$. This first lost place has a non-zero probability of reaching each place in the original Reference Sequence.

In the context of this work, where a mixture model is used to generate the observations, the above arrangement has some useful properties since, a concentration of the probability in the first lost place, $P_{lost,0}$ results in an uniform probability distribution over the places $P_0, P_1, \ldots, P_{K-1}$ for the Bernoulli mixture model.

3.5. Estimating the likely sequence of Places

The Viterbi algorithm is commonly used in the context of HMMs to determine the most probable sequence of hidden states that gave rise to a particular sequence of observations. In the context of the profile HMMs, where the HMMs are used to solve the problem of identifying substrings, the Viterbi algorithm is employed to find the 'best alignment', or the one with the highest probability, between two fragments of a string.

The algorithm itself is a type of Dynamic Programming algorithm. Using one hidden state at a time, the Viterbi algorithm calculates all the outcomes that could be possible for that state - and then keeps only the most likely one. After traversing the length of the HMM, the 'surviving' sequence
TABLE V: Comparison of results showing Place-Recognition failures with and without using HMMs.

<table>
<thead>
<tr>
<th>Localization Attempts</th>
<th>Reference Sequence</th>
<th># failures with No-HMM</th>
<th># failures with HMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

is the sequence that is most likely to have generated the observation.

The results of using the Viterbi algorithm for a set of 6 experiments is shown in Table V. The Hidden Markov model that was 5 observations long. The results for the experiment number 5 has also been shown graphically in Fig. 12.

The benefits of employing the Hidden Markov Model are much more obvious when the place recognition is hard to perform. In Fig. 13, we compare the results for localization using three procedures. The fact that the Reference and Query Sequences were contained under different lighting (light intensity and direction) conditions made Place recognition very difficult. In the figure at top, the place recognition results using only the Bernoulli Mixture Model are shown. A non-informative, Uniform probability over all the views of the Reference Sequence is maintained. The lower number of common features in the two sequence resulted in a high failure rate of 49%. At bottom, in Fig. 13c, the results of applying a 5-observation-long HMM, which resulted in a failure rate of 20% over the entire sequence, are shown. The figure in the middle shows the results of applying only the Bernoulli Mixture Model with an informative, non-uniform prior, which, itself was obtained from the the procedure in Fig 13c, with a success rate falling in between those obtained earlier.

Fig. 12: Comparison of place recognition with and without recourse to HMMs for a single Reference Sequence. Square Markers indicate the Place recognition results obtained by using 5-Observations-long HMMs. Place recognition was performed using a single sensor, Camera 1, over 116 images.

(a) Place recognition using a Bernoulli Mixture Model with a Uniform prior probability over all views of the Reference Sequence for matching each observation. A total of 57 failures were recorded from a total of 117 observations.

(b) Place recognition using a Bernoulli Mixture Model with an informative prior probability for matching each observation. A total of 46 failures were recorded from a total of 117 observations.

(c) Place recognition using a 5-observation-long Hidden Markov model. Each observation is matched using a Bernoulli Mixture Model with a prior given by the transition probability of the HMM. A total of 24 failures were recorded from a total of 117 observations.

Fig. 13: Comparison of place recognition with and without recourse to HMMs. The results represent two sequences taken at different times of the day (morning and late evening), a fact that makes View matching rather more difficult. Only Camera 1 and Camera 2 were utilized in these sequences.
4. Proposing the Creation of a Better Topological Path Representation

Up to this point we have considered sequences that were created by sampling the environment according to some sampling plan. This procedure creates the original Reference Sequence that can become quite large and contains the maximum amount of information (for a particular sampling plan). It is conceivable, however, that in certain cases removing some of the [less informative] views will result in more compact Topological representations of the path that might not, necessarily, provide inferior localization capabilities. Shorter, more compact Reference Sequences are desirable, for example, in applications involving communication between robots/persons having different capabilities and limited computing power or communication bandwidth. Compact representations of the Reference Sequence are also desired when long paths through the environment must be traversed, resulting in faster localisation.

A compacting procedure for the Reference Sequence should include two types of Views:

1) Essential Views are those Views at which the robot motion behaviour is altered.
2) Non-Essential Views improve the Topological Path representation.

The localization within the Reference Sequence might be viewed from an Information Theory approach (see [29] for an excellent introduction). There is a uncertainty associated with position occupied by the robot in the Reference Sequence. Each View provides some information that, potentially, could reduce some of the uncertainty of the location within the Reference Sequence.

Let $V$ indicate a view in the Reference Sequence and it can take the values $V_1, V_2, \ldots, V_K$ denoting one of $K$ distinct and detectable Views in the Reference Sequence. $V$ is correlated with the position $k$ of the robot in the Reference Sequence. This correlation may not be perfect however (the function is not injective), indicating that the same value of $V$ can be obtained at more than one position $k$ in the Reference Sequence. The 'mutual information' term $I(V; k)$ denotes the overlap of information between $V$ and $k$, i.e. it specifies how much the appearance of View $V$ reduces the uncertainty of the robot position $k$. The way in which individual views can contribute to the reduction in the uncertainty of the position within the Reference Sequence is shown graphically in Fig. 15.

Over the remainder of this section we shall search for ways to create compact Reference Sequences from the the original sampled Reference Sequence.

4.1. Essential Views in the Reference sequence

The Essential Views in the Reference Sequence or simply the Essential Views are exactly those views that are related to instances where the action being executed by the robot was changed. Essential Views must be maintained in any Reference Sequence since their removal implies that the robot misses (does not correctly identify) the start of a behaviour or the end of the previous behaviour and will get lost. All the other views are called Non-Essential Views.

The Essential Views can be identified in a Reference Sequence using the trivial pseudo code in Algorithm 1. It is clear the first sampled view $V_0$ and the last sampled view $V_{k-1}$ must be included as Essential Views.

4.2. Non-essential Views in the Reference Sequence

While it is essential to keep the Essential Views in the Reference Sequence, depending on the number of factors, it might be preferable to maintain additional Views that improve the probability of the robot successfully executing its mission. Non-Essential Views help the localization in the following way:

1) They increase the probability of identifying Essential Views
2) They increase the probability of correctly detecting that the robot is lost.

Here we look at the possibility of using exact and approximate methods to discover the set/s of Non-Essential Views to keep. This is an optimization task involving a number of factors and in which each factor implicitly affects the others.
Algorithm 1 Setting the Essential Views

\[ \begin{align*}
K &= \text{total number of sampled Views} \\
k &= \text{current sampled View being tested} \\
L &= \text{total number of Essential Views} \\
\text{Require: } & K \geq 2 \\
V_0 &\text{ is 1st Essential View } // \text{the first View is always an essential node} \\
L &= 1 \\
\text{while } (k < K - 1) \text{ do} \\
& \quad \text{if } (\text{Behaviour}_k \neq \text{Behaviour}_{k-1}) \text{ then} \\
& \qquad V_k \text{ is } L \text{th Essential View} \\
& \qquad L = L + 1 \\
& \quad \text{end if} \\
& \quad V_{k+1} \text{ is } L \text{th Essential View } // \text{the last View is always an essential node} \\
& \quad L = L + 1 \\
\text{end while} 
\end{align*} \]

The aim is to select and keep those non-essential views that improve the 'detectability' of the Essential Views so that the transition between these is made more reliable. As shown in Fig. 16, the original, sampled Reference Sequence might be broken down into a number of smaller sequences resulting in a compact representation of the Topological Path.

Choosing the set of Views that will improve the chances for localization of a particular Essential View is not akin to the third problem in HMMs, the problem of Learning, as described by Rabiner [10]. 'Learning' in the context of HMMs involves the use of multiple runs of observations to discover the HMM model (parameters) that is most suited to explain the sequence of observations.

The problem of selecting the Non-Essential Views entails selecting the most successful sub-set of Views from a single sequence. Such a set of parameters will depend on the length chosen between the Views of the HMM and (a related quantity) the number of Non-Essential Views chosen between a pair of Essential Views.

![Reference Sub-Sequence 1](image1)

![Reference Sub-Sequence 2](image2)

![Reference Sub-Sequence 3](image3)

Fig. 16: The Reference sequence is now decomposed into a series of sub-sequences.

Fig. 17: Where should the Next non-essential View go using a two-step process?

If exact methods are not useful for practical Reference sequences of any large size, we might resort to developing non-optimal, albeit workable solutions to the selection of Non-Essential Views.

These approximate methods might use cost functions that seek to maximise one or more of the above mentioned factors. For example, a strategy for addressing the first and second factors mentioned above would aim to select the earlier single view that has the greatest [total] chance of getting selected. The selection of the 'Next' non-essential View is not completely straightforward since the Prior probability \( P(k) \) in (10) implicitly influences the Place recognition using the Bernoulli Mixture Model.

A strategy for identifying Non-Essential Views by addressing the third factor would require the iterative use of a model of the Robot motion to select the next pair of views that have the greatest [total] chance of getting selected. We have attempted to present here how such a view selection process would reduce the size of the original, sampled Reference Sequence in Fig. 18. The transition probability matrix and the observation matrix described in the previous section were used to identify the Non-Essential Views.

While the Essential views depend of the behaviours that are adopted along the path, and on any markers laid down by the user, the Non-Essential views reflect the information content of that part of the environment. It can be quite easily seen from Fig. 18, that the information is not uniformly distributed along the path. Certain sections have views that sufficiently distinct to allow larger spacing between them.

\[ P_{\text{trans}}(V_{t+1} = V_j|V_t = V_i, \text{Behaviour}) \quad (15) \]
5. Conclusions and Future Work

In section 3, we presented the idea of using Hidden Markov Models (HMMs) to achieve robust place recognition by using the context of views. A dimensionality reduction method using Bernoulli Mixture Models was employed to perform unbiased place recognition. Good results place recognition results were obtained by using SIFT image features. The adoption of a Hidden Markov model improves the results of place recognition by constraining the matching in the context of the Reference Sequence and by providing a consistent means of maintaining the Prior Probability for the matching procedure. In the case where the environment has changed between the Reference Sequence and the Query Sequence, the HMM greatly improves the results of localisation. This improvement comes as a result of the fact that a prior probability can be maintained and updated in a consistent manner and because inconsistent transitions between views are ruled out.

In section 4, we suggested ways of compacting the Reference Sequence consisting of sampled views so as to keep a sub-sequence of views that would still be useful for localization. This step will be necessary to extend the application of the Reference Sequence to areas involving Robot Communication and Human-Robot interaction. The initial results reported at the end of this section indicate the fact that information is not distributed equally all along the Topological Path. Experiments that seek to compare the place-recognition performance behaviour of robots that use the original (Sampled) Reference Sequence as compared to the compact Reference Sequence are currently being performed (results not reported in this article).

In the near future, we intend to develop more complex Hidden Markov models that result in more intuitive and more compact Reference Sequences. The transition probabilities of an HMM can also be expressed in terms of the 'duration' or the average value of a quantity (say time or distance) for which the system remains at a states before changing. The Viterbi algorithm for such models must be modified so that these parameters can be factored in. This approach seems to be more intuitive for our problem and has been applied in the case of certain left-right HMMs in areas such model speech, handwriting or music recognition. The variable of interest in these problems is time and often the pace at which the system changes its states is a key aspect of the model. Various distributions have been employed to model these durations including some of the more well-known parametric distributions such as Exponential, Gaussian and Beta distributions [11], [12], [15], [13].

Using Second order HMMs might also be another useful idea, where a number of 'classes' of HMMs are applied [2]. Such a method might be especially useful in applications involving very long paths wherein similar sections of the environment are repeatedly encountered. Using these higher order HMMs could allow a more compact representation of the Topological Path.

References


[18] Jana Konseka and Fayin Li. Vision based topological markov localization. In *Proceedings of the International Conference on Robotics and Automation, ICRA 2004*, 2004. SIFT Utagage: As long as 4 could be matched successfully between the consecutive views they were assigned as belonging to the same location.


