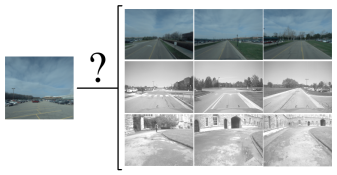
Environment Selection And Hierarchical Place Recognition

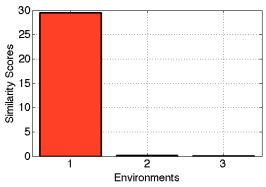
Mahesh Mohan*, Dorian Gálvez-López*, Claire Monteleoni* and Gabe Sibley†

*Department of Computer Science, The George Washington University
Email: {mahesh_mohan, dorian, cmontel}@gwu.edu
†Department of Computer Science, University of Colorado Boulder
Email: GSibley@colorado.edu

***Abstract*— As robots continue to create long-term maps, the amount of information that they need to handle increases over time. In terms of place recognition, this implies that the number of images being considered may increase until exceeding the computational resources of the robot. In this paper we consider a scenario where, given multiple independent large maps, possibly from different cities or locations, a robot must effectively and in real time decide whether it can localize itself in one of those known maps. Since the number of images to be handled by such a system is likely to be extremely large, we find that it is beneficial to decompose the set of images into independent groups or *environments*. This raises a new question: Given a query image, how do we select the best environment? This paper proposes a similarity criterion that can be used to solve this problem. It is based on the observation that, if each environment is described in terms of its co-occurrent features, similarity between environments can be established by comparing their co-occurrence matrices. We show that this leads to a novel place recognition algorithm that divides the collection of images into environments and arranges them in a hierarchy of inverted indices. By selecting first the relevant environment for the operating robot, we can reduce the number of images to perform the actual loop detection, reducing the execution time while preserving the accuracy. The practicality of this approach is shown through experimental results on several large datasets covering a combined distance of more than 750Km.**



(a) Query



(b) Environment selection

(c) Execution time

Fig. 1. Outline of the ideas proposed in the paper. (a) The figure on the left shows a potential query image being compared against a set of database images grouped into environments shown on the right. Different rows represent images obtained from different datasets. (b) The similarity scores computed using the proposed criterion to compare the query image to the existing environments. The plot of the scores shows that it is sufficiently discriminative to be used to determine the environment the image is most likely to be from. (c) Selecting the best environment helps reduce the number of images to be compared against, thereby improving the search time. We show the potential savings in computation time while querying a database of 85,000 images.

I. Introduction

The problem of place recognition has received considerable attention in the past decade. In its most general form, the problem can be formulated as searching for a query image in a database of existing images. Consequently, a large number of techniques for describing and comparing images and for retrieving the best matches have been proposed in the literature [3], [15]. One such popular framework is the Bag Of Words framework introduced in [22]. Here descriptors are extracted from the images and quantized using a vocabulary or code book of visual words. A visual word is a cluster that results from some discretization of the descriptor space. This results in a term frequency (tf) vector based representation of the given images. Queries can then be answered computing a similarity value using an Inverted Index, a structure that stores the list of images where each visual word appears.

This work was made possible with generous support from Toyota Motor Engineering & Manufacturing North America, Inc.

In the context of place recognition using the Bag of Words framework, an increase in the number of images increases the query time, since the underlying system is essentially a linear search. This has motivated recent work that aggregates term frequency (tf) vectors of consecutive frames and constructs a tree based lookup approach; this results in a significant improvement in query time at only a small loss in precision and recall [12]. For example, MacTavish & Barfoot [12] build accumulated bag of words of consecutive images to match groups of images with a single inverted index. This paper follows a similar vein, but it differs from [12] in two important ways. First, we show that the increase in query time for a simple inverted index is directly addressable by maintaining multiple sparse inverted indices. In other words, as the number of images increases, the sparsity of a single large inverted index is reduced. This reduces the efficiency and accuracy of the query retrieval process as more candidates need to be examined. However, separating the set

of places into subsets describing environments and building separate smaller inverted indices preserves the sparsity that we seek, thereby speeding up the querying process.

This decomposes the search problem into two parts: searching for the best subset and finding the best match within that subset. Since each subset is assumed to describe an environment, we call this problem environment selection. The second difference between [12] and this paper is the choice of the similarity criterion to select the best subset. At a conceptual level, environments can be treated as a probability distribution over all possible word sequences. MacTavish & Barfoot [12] use the first order statistics of this distribution, i.e. aggregated tf-vectors, for this purpose. In this paper we propose the use of the second order statistics of this distribution, which are well represented by the frequencies of co-occurrence of word pairs in the environment. This word-word co-occurrence (WWC) matrix representation of the environments motivates a novel similarity criterion that is much more effective for environment selection. This is important because the aggregated tf-vectors cease to be discriminative at larger temporal scales [5] [12], leading to significant loss in accuracy.

An overview of our approach is presented in Figure 1. Our proposal is based on considering subsets of images describing environments (represented by bags of binary words [22], [19]). Each environment is described in terms of its co-occurrent words. Similarity between environments is established by comparing their co-occurrence matrices. Final image-to-image loop detection is computed by using the environment-specific database. The important contributions of this paper are:

- We introduce a new problem called *Environment Selection* and propose a similarity criterion to determine similarity between image sets at the environment scale by comparing their corresponding word-word co-occurrence (WWC) matrices. We also provide its derivation from a direct product graph kernel. This is described in Section III.
- A hierarchical approach to place recognition based on the proposed similarity criterion, as described in Section IV.
- Experimentally we verify the improvement in the query time on datasets that span a distance of more than 750Km.
- In addition we also provide an experimental analysis of the properties of the proposed similarity criterion.

II. Related Work

Bags of words are a popular choice when the aim is long-term mapping and large scale place recognition [12], [17], [18], [6], [4]. This involves partitioning an image descriptor space given some training data to build a vocabulary whose words are the resulting cluster centers. Feature descriptors obtained from an image can be quantized to their nearest words using this vocabulary. Images can then be represented by a vector indicating the proportion of each word in the image. This together with an inverted index (or inverted file) allows the creation of a place database for efficient image retrieval.

FAB-MAP 2 [4] is one of the most notable works that makes use of words and an inverted index to find loops in a 1000 Km trajectory. The appearance information of the vocabulary is enriched with the spatial relationships of the words that co-occur in the training data. This is encoded in a Chow Liu tree to approximate the co-occurrence joint distribution and to reduce the memory requirements. They use a probabilistic formulation to compute the probability of a new frame being a revisited place or a new one. The matches that reach a fixed threshold are considered loop closures. In our work, we build co-occurrence graphs of words present in full environments to compare them online with the words that co-occur when querying the database.

The initial motivation for using co-occurrences to determine similarity came from [14] which leverages covisibility information to improve the accuracy of place recognition. The covisibility graph, being topometric in nature results in landmark-centric notion of places. While we do not harness the power of the covisibility graph in this paper, we note that it inherently relies on the same contextual information that we seek. This is because, foreground elements (like cars and people) are less likely to be covisible with the same background features, resulting in weak edges in the covisibility graph.

The problem of establishing similarity between environments is closely related to scene categorization described in the computer vision literature. For example, in [5], the authors show that there are distinct stylistic elements that describe a city and can be used to distinguish it from other cities. This is achieved through the use of discriminative learning and iterative clustering of image patches obtained from various images of the environment. In [11] images are classified according to the depicted scene by augmenting visual words with spatial information. However, these works do not address the final problem of detecting a loop in a map. For mapping, the authors of [17] present an approach to incrementally discover topics that can describe and group together images collected in the same place. Then, to address scalability, the most distinctive areas are retained by clustering the topic space.

Scalability is one of the main issues to address in long-term mapping. A large collection of images affect not only to the computational requirements but also the accuracy of the system due to the confusion that additional data causes. An efficient management of the available memory can achieve real-time performance in large maps [10]. However, this requires disregarding observation of places, leading to miss matches in future operations. When considering all the processed places, hierarchical approaches tackle scalability by dividing the image collection into tractable groups, so that only the most promising ones are inspected. For example, the work in [13] approaches the problem in two steps: a global localization yields a subset of candidate images by matching hue histograms, and a final single-image match is obtained by comparing SIFT features. However, they access

the images sequentially, which is less efficient than using a data structure as an inverted index [22].

Similarly, the authors of [12] aim to achieve logarithmic complexity localization by grouping images together. They analyze the optimal group size to compute accumulative bag-of-words vectors and obtain matches between groups. Nevertheless, they do not compute an image-to-image match required to close a loop in a SLAM map.

III. Similarity Criterion

We begin by assuming that two visual words co-occur if they are visible in the same image and the distance (in pixels) between their corresponding key points in the image is less than a threshold. We define a word-word co-occurrence (WWC) matrix as a symmetric matrix W where entry $W(i,j)$ contains the co-occurrence frequency of words w_i and w_j.

Formally, suppose we are given two environments, E_1 and E_2, with WWC matrices W_{E_1} and W_{E_2} respectively and a vocabulary of cardinality $|V|$. We define an environment as a sequence of images collected by an agent around some physical area.

In order to account for the various word-word inter-dependencies, which is a characteristic attribute of any environment, we consider the second order statistics of the distribution. In order to account for co-occurrence of a pair of words, we propose the following expression to compute the similarity:

$$S(E_1, E_2) = \sum_{i,j=1}^{|V|} \min[W_{E_1}(i,j), W_{E_2}(i,j)]. \quad (1)$$

Before we present the justification for this criterion, we briefly outline graph kernels that are used to compare two graphs.

A. Graph Kernels

Graph kernels are symmetric, positive semi-definite functions which measure the similarity between two graphs or between two nodes of a graph. In this paper, we use the direct product graph kernel. In this kernel, random walks are generated from each candidate graph being compared and the degree of similarity is determined by counting the number of common random walks [9], [7].

Before we describe the direct product graph kernels, we specify the notation being used [24]. Specifically, a graph G consists of an ordered set of vertices ζ and a set of undirected edges Υ. This is represented as a matrix where $\Upsilon_{ij} = 1$ if there is an edge between vertices v_i and v_j, and 0 otherwise. Each edge (v_i, v_j) is also associated with a weight w_{ij}. The adjacency matrix of the graph is represented as A (such that $A_{ij} = w_{ij}$). D is the diagonal matrix of node degrees, i.e. $D_{ii} = \sum_j A_{ij}$. The direct product graph $G_\times = \{\zeta_\times, \Upsilon_\times\}$ of two graphs G_1 and G_2 is given by $\zeta_\times = \zeta_1 \times \zeta_2$ and $\Upsilon_\times = \Upsilon_1 \otimes \Upsilon_2$, where $\otimes$ denotes the Kronecker product of two matrices. Let p_1 and p_2 be the starting probabilities of the random walks on G_1 and G_2, respectively. The starting probabilities of the random walks on the product graph is given by $p_\times = p_1 \otimes p_2$. The stopping probabilities q_1, q_2 and $q_\times$ are defined similarly. The graph kernel can now be described as

$$K(G_1, G_2) = \sum_{i=1}^{\infty} \lambda_i q_\times \Gamma_\times^i p_\times, \quad (2)$$

where λ_i is a decay factor. Each entry in $\Gamma_\times$ corresponds to a pair of edges $((e^1 = v_i^1, v_j^1)$ from graph G_1 and $(e^2 = v_i^2, v_j^2)$ from graph G_2) such that,

$$\Gamma_\times(i,j) = f(v_i^1, v_i^2, v_j^1, v_j^2, e^1, e^2). \quad (3)$$

Note that no assumptions have been made about the sizes of G_1 and G_2. The basic idea is that each walk in the direct product graph is equivalent to a joint walk in the two graphs G_1 and G_2. Thus the similarity of the two graphs can be determined using the weight of the walks in the direct product graph.

B. Theoretical Justification

In this section, we describe how the similarity criterion described earlier can be derived from the direct product graph kernel. In the context of place recognition using the bag-of-words approach, we assume that the environment is a stochastic process that generates a document based on an unknown underlying probability distribution over all the words in the vocabulary, i.e.

$$P(w_1, w_2, ..., w_{|V|}|E)$$

where $|V|$ is the number of words in the vocabulary.

Now for two environments to be similar, we impose the restriction that their corresponding joint distributions should be similar.

However, we note that computing the similarity between two joint probability distributions is prohibitively expensive in practice. We also note that the results in [5] indicate that comparing first order marginals $P(w_i|E_1)$ and $P(w_i|E_2)$ is not sufficiently accurate. Hence in this paper, we propose the use of second order marginal distributions, i.e $P(w_i, w_j|E_1)$ and $P(w_i, w_j|E_2)$ for comparing the similarity between two environments.

Comparing the second order marginal distributions is equivalent to comparing their corresponding graphical structures. To visualize this, consider a weighted undirected graph where the words w_i are the nodes and the weight of the edge between w_i and w_j represents $P(w_i, w_j|E_1)$. We can construct a similar graph for environment E_2. In practice, we do not have access to the underlying probability distributions. However we can determine the empirical distribution using the observed co-occurrence frequencies. The WWC matrices W_{E_1} and W_{E_2} are specific instances of these probability distributions. Now the problem of determining the similarity between the two second order marginal distributions $P(w_i, w_j|E_1)$ and $P(w_i, w_j|E_2)$, is approximately equivalent to determining the similarity between the two corresponding graphs denoted by the WWC matrices. To

Algorithm 1 Construct WWC Matrix

Data: I : dataset containing images $I_1, I_2 ... I_n$
Result: W : The word-word co-occurrence matrix for I
for $j = 1 : n$ **do**
  $words = \text{ExtractWords}(I_k)$
  foreach *pair* $\langle w_1, w_2 \rangle$ *in words* **do**
    if *co_occur*(w_1, w_2) **then**
      if $w_1 < w_2$ **then**
        $W(w_1, w_2) = W(w_1, w_2) + 1;$
      end
    end
  end
end

compare the two graphs, we use the graph kernel described in the previous section.

Specifically, we set the function f as follows:

$$f(v_i^1, v_i^2, v_j^1, v_j^2, e_i, e_j) = \begin{cases} \min(e_i, e_j) & \text{if } v_i^1 = v_i^2 \wedge v_j^1 = v_j^2 \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Also, we set both $p_\times$ and $q_\times$ to be the uniform distribution, i.e. $p_\times(i) = 1/\|\zeta_\times\|$ (and similarly for $q_\times$). This reduces the expression for $K(G_1, G_2)$ to the sum of the entries of the matrix $\Gamma_\times$. This choice is motivated by the observation that the different images in the environment are likely to have a different number of words and without prior information, all document lengths are equally likely.

The final approximation is that we restrict ourselves to only random walks containing one edge (or two vertices). In other words, we set $\lambda_1 = 1$ and $\lambda_i = 0$ for all $i > 1$. This reduces the graph kernel in the previous section to the proposed similarity criterion. In order to show that this is a valid kernel, we note that the $\min()$ function is a valid kernel (from the Histogram Intersection Kernel). The sum of kernel functions is still a kernel. Since the proposed criterion is just the sum of a set of $\min()$ functions, the validity follows.

C. Constructing the WWC Matrix

The algorithm is shown in Algorithm 1. Coarse words are extracted from each image in the dataset. The explanation of the coarseness is given in Section IV-B. Co-occurrence counts are incremented for a pair of words $\langle w_1, w_2 \rangle$ if the distance between the corresponding features lie within a certain distance of each other in the image. If a pair of words occurs multiple times in the same image, the corresponding co-occurrence counts are incremented multiple times.

The threshold used in the definition of co-occurrence used here has two major purposes:

- It enforces a weak geometric consistency check, and
- ensures greater sparsity in the WWC matrix.

D. Computational Considerations

From an implementation standpoint, we note that the matrix $\Gamma_\times$ is extremely large, of size $O(|V|^2)$. However, since the tf-vector for each image is generally sparse, the WWC matrix is reasonably sparse. Consequently, $\Gamma_\times$ is also relatively sparse. This means that the proposed criterion can be computed fairly efficiently in practice (since it only depends on the number of non-zero terms in the WWC matrices). Also, the matrix is maintained in a upper triangular form for efficiency. However, as the size of the vocabulary increases, the WWC matrices are likely to contain more word pairs. In such cases, we note that the pairs of features that have a low frequency of co-occurrence are usually less descriptive of the environment and hence can be disregarded from the similarity computation. Specifically, this is done by setting entries in the WWC matrix to zero, if the corresponding entries are below a threshold λ (we call this threshold the sparsification threshold in the rest of this paper).

IV. Hierarchical Place Recognition

Our loop detection approach follows the loop detection algorithm for single environments DBoW2 presented in [6]. For the reader's convenience, we start by giving a brief description of that technique.

A. Single Environment

DBoW2 uses a fixed-size visual vocabulary tree of binary words to represent images. This is the result of a hierarchical k-means clustering of the descriptor space, so that clusters at deeper levels of the tree correspond with finer discretization levels. The leaves of the tree are the vocabulary words and are given a *term frequency - inverse document frequency* (tf-idf) weight that depends on their discriminative power (according to the training data). Images are indexed by an inverted index that stores for each word in the vocabulary the images that contain it, allowing fast retrieval of those images with words in common. Finally, a direct index stores the features of the images at any level of the vocabulary tree, i.e. at any descriptor discretization level.

Given a query image q, a normalized similarity score η_{qi} based on the Bhattacharyya coefficient is computed for each i-th candidate match. Those that exceed a threshold α are grouped together when they were taken at close positions, yielding a cumulative group score H_q. The best-ranked image of the group that maximizes H_q is selected as a loop candidate. If this candidate is consistent with k previous loop candidates, it is selected as a loop detection. This test is called *temporal consistency* and provides robustness to the results. In [6] a final geometrical verification based on epipolar geometry was done before accepting a match. However, we deactivated that functionality in our work because it veils our analysis.

B. Hierarchical Approach

Our loop detection approach in this paper is depicted by Figure 2. We use a single static vocabulary tree of branching factor 10 and 6 depth levels (resulting in 10^6 words) trained with 99M ORB features obtained from 108K independent and generic images of the SUN397 dataset [25]. This ensures a very descriptive configuration and fast access time. As an example, we show in Table I the execution time of

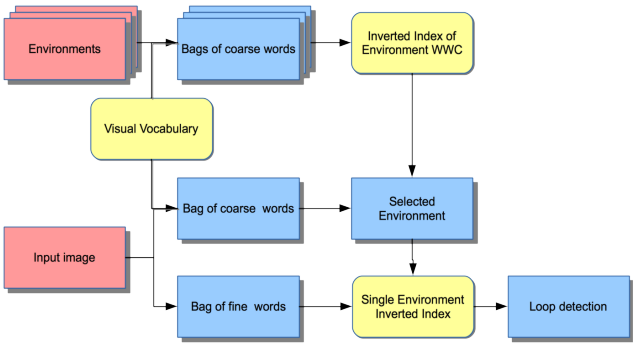


Fig. 2. Proposed algorithm for Hierarchical Place Recognition.

TABLE I
MEDIAN OF EXECUTION TIMES IN NORDLANDSBANEN WITH 71534 IMAGES (MS)

Vocabulary size	Insertion	Detection
10^4	1.3	453.0
10^5	1.8	100.1
10^6	2.5	22.0

performing image matching with different vocabulary sizes in a sequence of 71K images. The larger the vocabulary, the more sparse its inverted index, requiring fewer comparison when computing candidate matches. On the other hand, a very high number of words can make them too specific, so that similar image features lie in different words, preventing them from matching and decreasing accuracy. The vocabulary sizes we selected do not exhibit this problem, as we show in Section V.

We assume that we are provided with a set of environments obtained from other robots' maps. In an offline stage, we compute the ORB features and the the bag of words of all the images of each environment to create its WWC matrix. Since the size of this matrix is quadratic in the number of words, we consider two different discretization levels for the words: a fine level of 10^6 words to represent single images, and a coarser level of 10^5 words for environments. Selecting a coarse level can be done effortlessly and without demanding another vocabulary because it only requires to pick the parent node of the fine word in the vocabulary tree. Although the coarseness level trades off speed and accuracy, 10^5 words provide a good balance as we show in Section V.

To provide a fast environment score computation, we encode all the WWC matrices in a high-level inverted index. At the same time, each environment stores the bag-of-words of its images in an environment-level inverted index. Note that the high-level inverted index indexes entries by pairs of words and the environment specific inverted index by single words. All of them compose the hierarchy of inverted indices of our approach.

During the online stage, given a query image, two bag-of-words vectors are computed at fine and coarse levels. The coarse vector is used to build a WWC matrix I of that single image. The high-level inverted index is inspected then to compute the score $S(I, E_i)$ (equation 1) for any environment E_i with pairs of co-occurrent words in common with I. The environment $E^* = \arg\max_{E_i} S(I, E_i)$ is selected as the most similar one. Finally, the environment-level inverted index related to E^* is accessed to obtain an image-to-image loop detection. At this stage, the temporal consistency check explained above is applied to enhance the results.

V. EXPERIMENTAL RESULTS

A. Hierarchical Place Recognition

To show the versatility of the place recognition, we used 7 datasets collected by other authors, shown by Table II. These are long image sequences taken by mounting a camera on some kind of vehicle (robot, car, train) taken with different cameras and under different conditions, producing a total of 650K images of different sizes.

In each experiment, we create two subsets of images for each dataset. One is added to the image database of the tested loop detector and the other is used to perform queries to obtain loop matches. Each subset is created by reading the images of each dataset at a certain frequency f, applying a time offset to obtain disjoint subsets. In order to ensure a fair comparison between the methods, no geometric verification was used in any of the experiments to verify the matches obtained.

We performed two sets of experiments. The first experiment sampled the datasets at approximately 2Hz to obtain a total of 85,508 images, while the second experiment used a sampling frequency of 5Hz to obtain 214,483 images. The first experiment was used to verify the accuracy of the system, since fewer training images were used. whereas the second experiment was used to verify the computational scalability of the proposed approach. We compared our hierarchical approach with a detector baseline [6] that adds all the images into a single common database. Table III shows the execution time obtained on a MacBook Pro 2.7 GHz Intel Core i7 with 16 GB DDR3 memory. We used a visual vocabulary of 10^6 words for performing place recognition and a vocabulary of 10^5 words for building the WWC matrices. The sparsification threshold was set to $m(r)$ (the mean of the non-zero elements in the row r).

In order to accommodate the results for all the datasets, we summarize the results as follows: for each dataset, we compute the maximum precision obtained and report the corresponding value of recall. These results are shown in Table IV for the case $f = 2$Hz. The results clearly show that the Hierarchical Inverted Index achieves slightly lower but comparable accuracy to the single Inverted Index in most cases, and requiring a much lower execution time, as shown in Table III. In addition, there are two interesting observation we make:

- The first observation is that for the Whitmore dataset, the Hierarchical Index actually outperforms the single Inverted Index. This is attributed to the fact that by selecting the correct environment, we can eliminate a significant number of ambiguous matches.

TABLE II

DATASETS

Dataset	Description	Length (Km)	Average speed (m/s)	Image size (px×px)
New College [23]	Outdoors, dynamic	2.3	1.5	512×384
Bicocca 2009-02-25b [21]	Indoors, static	0.8	0.5	640×480
Ford Campus 2 [20]	Urban, slightly dynamic	4	6.9	407×621
Malaga 2009 Parking 6L [2]	Outdoors, slightly dynamic	1.2	2.8	1024×768
Whitmore (own)	Urban, slightly dynamic	-	-	640×480
St Lucia 19-08-09 08:45 [8]	Urban, dynamic, JPG artifacts	17.6	12.1	640×480
Nordlandsbanen Spring [1]	Outdoors, static, JPG artifacts	729	20.4	1920×1080

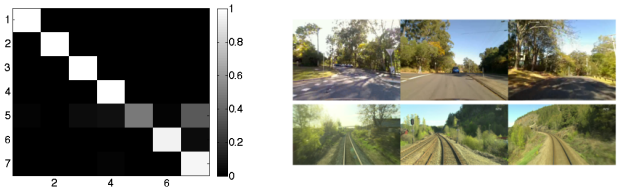
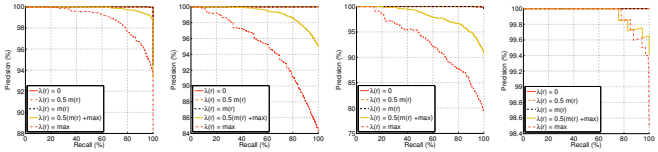


Fig. 3. Figure showing the Confusion matrix for the Environment Selection problem. The environments are {Bicocca25b, NewCollege, Whitmore, Malaga6L, StLucia, Ford2, Nordlands}. The confusion matrix shows that images from StLucia are often classified as Nordlands. From the images in the first row (StLucia) and second row (Nordlands), we see that foliage in both datasets occupy a significant portion of the features obtained, leading to ambiguity.



(a) Bicocca25b (b) NewCollege (c) Ford2 (d) Whitmore

Fig. 4. PR Curves for Environment Selection for different values of the sparsification threshold λ. A lower value of λ implies fewer elements are discarded, leading to higher accuracy. The interesting thing to note here is that for all the environments, setting $\lambda = m(r)$ gives us the same performance as the full matrix (i.e, when $\lambda = 0$). This implies that words that co-occur less frequently are less informative for environment selection. This is in contrast to place recognition where words that occur rarely are considered more informative.

TABLE III

COMPARISON OF MEAN EXECUTION TIME FOR DIFFERENT METHODS (MS)

Images	Index Type	Environment Selection	Query	Total
85,508	Hierarchical	7.09	10.7	17.79
	Single	-	24.51	24.51
214,483	Hierarchical	6.89	37.73	44.6
	Single	-	66.37	66.37

- The second observation is that both the Single Inverted Index approach and the Hierarchical Inverted Index approach perform poorly on the StLucia dataset. Further examination of the confusion matrix, shown in Figure 3, reveals that some images from the StLucia and Nordlands datasets look alike. Images from both the datasets are also shown in Figure 3. Although this is a challenge for the similarity criterion, the results of our Hierarchical Inverted Index are still similar to those of the Single Inverted Index.

TABLE IV

COMPARISON OF RECALL (R) FOR MAXIMUM VALUES OF PRECISION (MP) FOR DIFFERENT METHODS

	Single		Hierarchical	
Dataset	MP	R	MP	R
Bicocca25b	100	94.49	100	97.14
NewCollege	99.99	90.75	99.87	90.18
Malaga6L	100	94.48	100	90.56
Whitmore	99.49	57.6	99.62	76.32
StLucia	98.75	25.55	97.85	23.49
Ford2	100	95.77	100	91.96

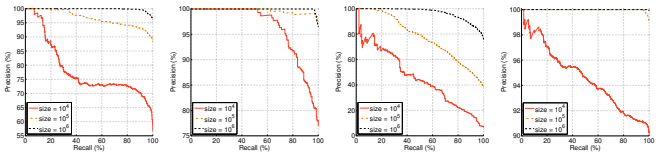
B. Environment Selection

In this subsection we discuss a few properties of the similarity criterion. There are three major factors that affect the discriminative ability (the ability to differentiate the true environment from other environments) of the proposed similarity criterion, namely the quantizing vocabulary V, the sparsification threshold λ and the number of images used to construct the WWC matrix. We proceed to examine each in turn. For all the experiments in this section, we obtain the test and training data in the same way as described in the previous subsection. The training data belonging to a specific environment is used to compute the corresponding WWC matrix. Each of the test images is treated as a separate query. We then use the similarity criterion on the test images to determine the environment they belong to.

1) Sparsification Threshold: Throughout our experiments, we have used $\lambda(r)$ to the mean of the sum of the non-zero entries in the row r. In other words,

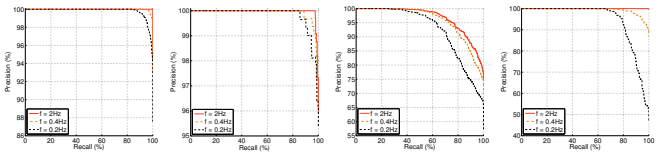
$$\begin{aligned} \lambda(r) &= mean(W(r,j)) \\ &\approx \sum_j W(r,j)/|W(r,:)| \\ &= 1/|W(r,:)| \end{aligned} \tag{5}$$

Here $|W(r,:)|$ indicates the number of non-zero elements in the row r. Since the WWC is row normalized, sum of the elements in each row is 1. This gives us the final expression $\lambda(r) = 1/|W(r,:)|$. Since this is not the true mean, we represent it by $m(r)$. We experimentally justify this choice by comparing it with 4 other thresholds. Specifically we present PR curves for $\lambda(r) = \{0, m(r)/2, (m(r)+max)/2, max\}$ in Figure 4.



(a) Ford2 (b) Whitmore (c) StLucia (d) NewCollege

Fig. 5. PR Curves for Environment Selection using vocabularies of different sizes. This monotonic improvement shows that the discriminative ability of the criterion improves with the size of the vocabulary.



(a) Bicocca25b (b) Whitmore (c) StLucia (d) Malaga6L

Fig. 6. PR Curves for Environment Selection using different sampling rates. This sampling rate controls the number of images used to construct the WWC matrix. Reducing the number of images used to construct the WWC reduces the discriminative ability of the proposed criterion. However, these curves show that environments can be identified correctly even when we use much fewer images. This highlights a dominant underlying structure to these environments.

The results show that choosing $\lambda(r) = 1/|W(r,:)|$gives us the best performance in many datasets. This can be explained by observing that at the environment level, pairs that co-occur very rarely, hardly contribute to the similarity score. This is in contrast to standard place recognition, where the infrequently occurring words contain most of the discriminative information. Another observation is that if a pair of words co-occur too frequently, they are no longer discriminative. This explains the reduced performance for $\lambda > mean$.

2) Vocabulary Size : The discriminative ability of the BoW approach improves with the size of the vocabulary. Since the WWC matrix relies heavily on the vocabulary, the discriminative ability also improves with vocabulary size. This is shown for various environments in Figure 5. However, we note that higher vocabulary sizes imply larger WWC matrices and consequently takes longer to compute the similarity. This offers a nice tradeoff between memory consumed and accuracy.

3) Effect of Number of images: The number of images used to construct the WWC directly affects the discriminative ability of the WWC matrix. In order to verify the impact on the PR curve, we used a subset of the total images in the dataset. The subset was obtained by choosing images at fixed intervals. In separate runs of the experiment, the interval size was adjusted to obtain different number of images in the subset. The results are shown in Figure 6. These indicate that the higher the number of images used to construct the WWC matrix, the better the results. However, we note that even with as few as 1000 images per dataset, we obtain reasonable accuracy for the environment selection problem. This underlines the existence of a distinct dominant latent structure to these environments and thus justifies our use of the criterion.

VI. Conclusions

In this work we have derived a graph kernel to compare graphs of co-occurrent image words. With this, we have formulated a similarity criterion to measure the similitude in appearance and geometrical spaces between environments for place recognition for SLAM. We have also proposed a novel hierarchical place recognition algorithm to detect loop closures in imagery obtained from very large and heterogeneous trajectories, adding up to more than 750Km in length. Leveraging the presented environment similarity criterion, we have showed that by using two nested levels of inverted indices, we are able to discriminate between environments to reduce the search space of image candidates. This leads to a decrease of the required execution time for the full place recognition without affecting the accuracy.

Aiming at a scenario where many robots or users with mobile phones [16] can create, share and need to reuse maps, we consider that this work is a first step to a long-term algorithm for place creation and maintenance. To accomplish this goal, in a future work, we will address issues such as automatic online creation of new environments or environment fusion after successful recognition of places in different environments.

References

[1] Nordlandsbanen: minute by minute, season by season, January 2013.

[2] José-Luis Blanco, Francisco-Angel Moreno, and Javier González. A collection of outdoor robotic datasets with centimeter-accuracy ground truth. *Autonomous Robots*, 27(4):327–351, November 2009.

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

[4] Mark Cummins and Paul Newman. Appearance-only slam at large scale with fab-map 2.0. *The International Journal of Robotics Research*, November 2010.

[5] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei A. Efros. What Makes Paris Look like Paris? *ACM Transactions on Graphics (SIGGRAPH)*, 31(4), 2012.

[6] Dorian Gálvez-López and J. D. Tardós. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, October 2012.

[7] Thomas Gärtner, Peter Flach, and Stefan Wrobel. On graph kernels: Hardness results and efficient alternatives. In *Learning Theory and Kernel Machines*, pages 129–143. Springer, 2003.

[8] Arren Glover, Will Maddern, Michael Milford, and Gordon Wyeth. FAB-MAP + RatSLAM: Appearance-based SLAM for Multiple Times of Day. In *IEEE International Conference on Robotics and Automation*, Anchorage, USA, 2010.

[9] Hisashi Kashima, Koji Tsuda, and Akihiro Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the Twentieth International Conference on Machine Learning*, volume 3, pages 321–328, 2003.

[10] M. Labbe and F. Michaud. Appearance-Based Loop Closure Detection for Online Large-Scale and Long-Term Operation. *IEEE Transactions on Robotics*, 29(3):734–745, June 2013.

[11] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178. IEEE, 2006.

[12] K. MacTavish and T. D. Barfoot. Towards Hierarchical Place Recognition for Long-Term Autonomy. In *IEEE International Conference on Robotics and Automation (ICRA) workshop on "Visual Place Recognition in Changing Environments"*, May 2014.

[13] L. Maohai, S. Lining, H. Qingcheng, C. Zesu, and P. Songhao. Robust Omnidirectional Vision based Mobile Robot Hierarchical Localization and Autonomous Navigation. *Information Technology Journal*, 10(1):29–39, January 2011.

[14] Christopher Mei, Gabe Sibley, and Paul Newman. Closing loops without places. In *International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, October 2010.

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

[16] Jack Morrison, Dorian Gálvez-López, and Gabe Sibley. MOARSLAM: Multiple Operator Augmented RSLAM. In *International Symposium on Distributed Autonomous Robotic Systems*, November 2014. Accepted for publication.

[17] Liz Murphy and Gabe Sibley. Incremental Unsupervised Topological Place Discovery. In *IEEE International Conference on Robotics and Automation*, May 2014.

[18] T. Nicosevici and R. Garcia. Automatic visual bag-of-words for online robot navigation and mapping. *IEEE Transactions on Robotics*, 28(4):886–898, Aug 2012.

[19] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, 2006.

[20] Gaurav Pandey, James R. McBride, and Ryan M. Eustice. Ford campus vision and lidar data set. *The International Journal of Robotics Research*, 30(13):1543–1552, November 2011.

[21] Rawseeds. Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets (Project FP6-IST-045144)., 2007-2009.

[22] Josef Sivic and Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision*, pages 1470–1477, 2003.

[23] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The new college vision and laser data set. *The International Journal of Robotics Research*, 28(5):595–599, May 2009.

[24] S.V.N. Vishwanathan, Nicol N. Schraudolph, Risi Kondor, and Karsten M. Borgwardt. Graph kernels. *The Journal of Machine Learning Research*, 11:1201–1242, 2010.

[25] Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. SUN Database: Exploring a Large Collection of Scene Categories. *International Journal of Computer Vision*, pages 1–20, 2014.