Robust Visual Place Recognition with Graph Kernels

Elena Stumm, Christopher Mei, Simon Lacroix,
LAAS-CNRS,
University of Toulouse
{estumm, cmei, simon}@laas.fr

Juan Nieto, Marco Hutter, Roland Siegwart
Autonomous Systems Lab
ETH Zurich
{nietoj, mahutter, rsiegwart}@ethz.ch

Abstract

A novel method for visual place recognition is introduced and evaluated, demonstrating robustness to perceptual aliasing and observation noise. This is achieved by increasing discrimination through a more structured representation of visual observations. Estimation of observation likelihoods are based on graph kernel formulations, utilizing both the structural and visual information encoded in covisibility graphs. The proposed probabilistic model is able to circumvent the typically difficult and expensive posterior normalization procedure by exploiting the information available in visual observations. Furthermore, the place recognition complexity is independent of the size of the map. Results show improvements over the state-of-the-art on a diverse set of both public datasets and novel experiments, highlighting the benefit of the approach.

1. Introduction

Efficient and reliable place recognition is a core requirement for mobile robot localization, used to reduce estimation drift, especially in the case of exploring large, unconstrained environments [7, 20]. In addition to robotics, place recognition is increasingly being used within tasks such as 3D reconstruction, map fusion, semantic recognition, and augmented reality [9, 11, 15, 28]. This paper examines appearance-based place recognition approaches which combine visual and structural information from covisibility graphs for achieving robust results even under large amounts of noise and variety in input data. For instance, dealing with appearance changes, self-similar and repetitive environments, viewpoint and trajectory variations, heterogeneous teams of robots or cameras, and other sources of observation noise make the task particularly challenging. Figure 1 shows an example that illustrates how different cameras affect the appearance of a location.

By representing locations with their corresponding covisibility graphs, pseudo-geometric relations between local visual features can boost the discriminative power of observations. Covisibility graphs can be constructed as the environment is traversed, by detecting local landmarks, and connecting those landmarks which are co-observed in a sparse graph structure [24]. Candidate locations resembling a given query can then be efficiently retrieved as clusters of landmarks from a global map, using visual word labels assigned to each landmark and an inverted index lookup table. Such a location-based covisibility subgraph will be referred to as a location graph. Using this representation, inspiration is taken from the field of graph theory, more specifically graph kernels, for computing the similarity between the corresponding query and candidate location graphs. As a result, inference can be achieved using more spatial and structural information than bag-of-words or word co-occurrence approaches to visual place recognition.

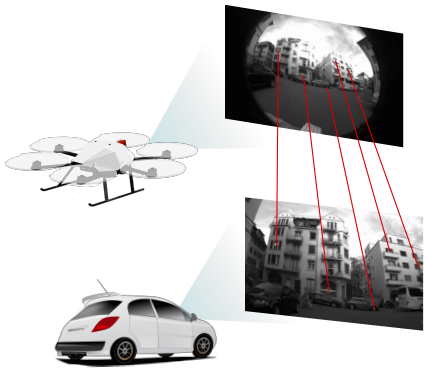


Figure 1: In an effort to move towards robust mapping and localization in unconstrained environments, this paper investigates graph comparison approaches to visual place recognition. Structural and visual information provided by covisibility graphs is combined, in order to cope with variations and noise in observations, such as those coming from heterogeneous teams of robots.

The presented approach does not require any detailed prior representation of the environment, using only rough priors on feature occurrences as additional input. Furthermore, computation does not scale with the size of the map. The approach is therefore well suited to applications including exploration and mapping of unknown areas.

2. Background

State-of-the-art localization methods typically rely on visual cues from the environment, and using these, are able to be applied even on large scales of several hundreds or thousands of kilometers, and sometimes under changing conditions [12, 21, 33]. However, the recent trend is to rely on localizing within a prior map, or relying on enough training and sample data, as in the works of [23, 21]. One of the main goals of this work is to achieve visual place recognition using no prior data from the environment, in a way which is robust to repetitive scene elements, observation changes, and parameter settings.

Visual place recognition can be achieved using global image attributes, as in the work of [25]. By comparing sequences of images, global image descriptors can produce astounding results using relatively simple methods [33], but rely on strong assumptions about view-point consistency. Alternatively, methods using locally-invariant features (such as SIFT [22], SURF [8], or FREAK [3]) are commonly applied when such assumptions do not hold. Furthermore, relative positions of these visual features can be used to perform geometric reconstruction and localization, such as in the work of [2]. The efficiency of these methods can be substantially improved by using techniques including hamming-embedding [16], product-quantization [18], inverted multi-indices [4], and descriptor projection [23] for efficient and accurate descriptor retrieval and matching. However, problems with these approaches appear in the case of repetitive elements and scenes, a common occurrence especially in large environments. Repetition can happen on several scales, such as burstiness of visual elements within a scene (*e.g*. plant leaves, windows on building facades) [17, 34] causing difficulty for descriptor lookup and matching with the ratio test; and repetitive scenes themselves (*e.g*. streets in a suburb) causing perceptual aliasing during geometric matching. On the other hand, other approaches quantize local features into visual words, providing a useful representation for probabilistic and information theoretic formulations to avoid the aforementioned issues. Typically, geometry is no longer explicitly used during inference, rather relying on more sophisticated location models in order to avoid perceptual aliasing due to the loss of global structure [12, 21, 31].

In order to incorporate relative spatial information from geometric constraints into observation models, a number of methods have been investigated. For example, the work of [27] incorporates learned distributions of 3D distances between visual words into the generative model in order to increase robustness to perceptual aliasing. In [19], features are quantized in both descriptor and image space. This means that visual features are considered in a pairwise fashion, and additionally assigned a spatial word, which describes their relative positions in terms of quantized angles, distances, orientations, and scales. In recent years, graph comparison techniques have become popular in a wide array of recognition tasks, including place recognition. Applied to visual data, graphs of local features are created and used to represent and compare things such as objects. The work of [36] uses graph matching techniques which allow for inclusion of geometric constraints and local deformations which often occur in object recognition tasks, by introducing a factorized form for the affinity matrix between two graphs. This approach explicitly solves for node correspondences of object features. Alternatively, the works of [14] and [5] apply graph kernels to superpixels and point clouds in order to recognize and classify visual data in a way which does not explicitly solve the node correspondence problem, but provides a similarity metric between graphs by mapping them into a linear space. In the described approaches, graph comparison was applied on relatively small graphs consisting of only tens of nodes due to complexity. For the case of graph kernels, random walk and subtree kernels applied in [5, 14], scale with at least $O(n^3)$ with respect to the number of nodes n [35]. Other types of graph kernels have since been proposed, which strengthen node labels with additional structural information in order to reduce the relative kernel complexity [6, 29] and open the door for applications to larger graphs. For example, in [29], Weisfeiler-Lehman (WL) graph kernels scale with $O(m)$ with respect to the number of edges m. Further details regarding graph kernels will be discussed in Section 3.2.2. In regards to visual place recognition, graph comparison has been applied in works such as [26, 32] which make use of landmark covisibility to compare locations based on visual word co-occurrence graphs, and also scale with the number of edges. The work of [26] demonstrates how the defined similarity measures can be interpreted as simplified random-walk kernels.

In this work, we take further inspiration from existing work on graph kernels and the graph-based location interpretation to boost the reliability of visual place recognition in difficult scenarios. Specifically, this paper offers the following contributions:

- an analysis into using graph kernels for visual place recognition – with the development of a novel graph kernel which is both efficient, and robust to noisy observations and perceptual aliasing
- insight into the Bayesian normalization term – with the introduction of a constant normalization scheme which greatly reduces computational cost without compromising results

The following section will outline how visual observations are represented as graphs of visual words, and how efficient inference can be done using such observation models. The proposed methods are additionally validated through experimental analysis in Section 4.

3. Methodology

3.1. Location Graphs

Given a query location (*e.g.* the current position of a robot), the idea is for the system to be able to evaluate if and where the same location was seen before. The approach developed in this paper relies on location descriptions comprised of sets of visual words (also referred to as bag of words) [30, 10], enabling efficient comparison of the query with a set of candidate locations retrieved from the current map. Quantized visual words are therefore used to represent feature descriptors provided by each landmark (distinct visual features in the image). A map is then constructed as an undirected covisibility graph, with these landmarks as nodes, and edges representing relationships between landmarks. In this work we choose the number of times features are seen together as the edge information, following the procedure described in [24, 31]. For place recognition, edges are additionally weighted according to the amount of information their corresponding landmarks convey, which can be estimated using visual word priors for each landmark: $I = -\log[P(w_u)P(w_v)]$ [32]. At query time, the graph can be searched for clusters of landmarks which share strong similarity with the query using an inverted index, extracting subgraphs which represent candidate locations for further analysis. These candidate locations are not predetermined, but depend on the information in the query, providing some invariance to the sensor trajectory and image frame-rate [31].

The average size of each retrieved location is typically on the order of hundreds of nodes, depending on the environment and feature detector. Location graphs tend to be densely structured, with each node being connected to roughly one hundred other nodes on average. Furthermore, the size of the label set associated to nodes in the graph corresponds to the size of the visual vocabulary used (in our case roughly 10,000 words). The size and structure of these graphs are an important factor when considering the methods of analysis which can be applied, as it drives subsequent approximations and complexity.

3.2. Place Recognition

3.2.1 Probabilistic Framework

The posterior probability of being in a certain location, $\mathcal{L}_i$, given a query observation, $\mathcal{Z}_q$, can be framed using Bayes' rule as follows,

$$P(\mathcal{L}_i|\mathcal{Z}_q) = \frac{P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i)}{P(\mathcal{Z}_q)} \tag{1}$$

Typically, the normalization term, $P(\mathcal{Z}_q)$, is either computed by summing likelihoods over the entire map and/or sampling observation likelihoods from a set of representative locations; or often skipped entirely and the observation likelihood is used directly (at the loss of meaningful probability thresholds) [31]. This normalization term can be formulated as the marginalization over the particular location of interest, $\mathcal{L}_i$, and the rest of the world, $\overline{\mathcal{L}_i}$:

$$P(\mathcal{Z}_q) = P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i) + P(\mathcal{Z}_q|\overline{\mathcal{L}_i})P(\overline{\mathcal{L}_i}) \tag{2}$$

resulting in the following equation for the posterior probability:

$$P(\mathcal{L}_i|\mathcal{Z}_q) = \frac{P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i)}{P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i) + P(\mathcal{Z}_q|\overline{\mathcal{L}_i})P(\overline{\mathcal{L}_i})} \tag{3}$$

In this work, we propose that the representation of visual observations is unique enough such that the average observation likelihood of the observation coming from a place which does not match the query, $P(\mathcal{Z}_q|\overline{\mathcal{L}_i})$, remains approximately constant. As a result, this value can be estimated once and then used in the posterior normalization step without the need of its costly calculation for each query. This assumption arose from the difficulty in actually producing reliable results using sampling. This is due to the fact that the sample space for such complex observation models becomes too large to sample effectively. However, upon further introspection, and based on the selected representation of locations, it can be seen that the dependence on sample locations becomes unnecessary as our assumption provides an effective approximation. Perceptual aliasing, can of course still happen, if scene similarity is very high. However without having a prior map of the environment, this cannot easily be avoided. In essence, normalization by a sample set typically prevents perceptual aliasing due to common sets of scene elements, while in this paper we argue that given enough context and structure, the confusion between locations containing similar elements is greatly reduced.

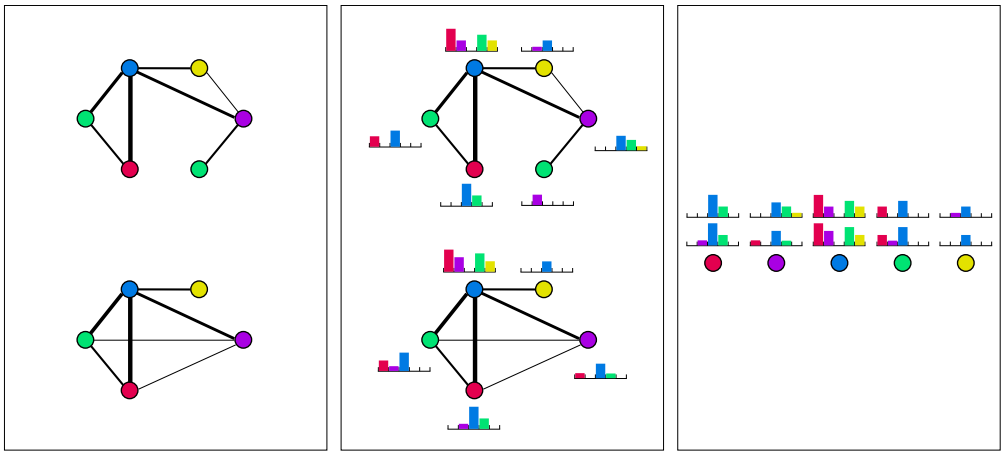
The following section will now explain how graph comparison techniques can be used to estimate observation likelihoods by locations using their covisibility graphs, and later Section 4 will validate the proposed assumptions with experimental results.

3.2.2 Graph Comparison

As previously discussed, graph kernels can provide an efficient means of graph comparison. A graph kernel function,

$$k(\mathcal{G}, \mathcal{G}') = \langle \phi(\mathcal{G}), \phi(\mathcal{G}') \rangle \tag{4}$$

defined between two graphs, $\mathcal{G}$ and $\mathcal{G}'$, effectively maps the graphs into a linear feature space, and can act as a similarity measure. In this work, we investigate the use of graph kernel representations to define similarities between location graphs and estimate the observation likelihood of being in a given location, $P(\mathcal{Z}_q|\mathcal{L}_i)$. Kernels can be defined in a



(a) input graphs with original labels

(b) node re-labelling with neighbourhood vectors

(c) node-to-node comparison with neighbourhood vectors

Figure 2: Illustration of the graph comparison process. The input graphs with node labels are shown, followed by the re-labelled graphs including each corresponding neighbourhood vector, and a node-to-node comparison of neighbourhood vectors from each graph. Colours in the node labels represent elements from the given vocabulary, and edge values are represented by line thickness.

number of different ways, and kernel choice is often important for achieving useful results, as it acts as an information bottleneck. Therefore, in kernel selection, prior knowledge about data types and domain patterns is valuable.

The most commonly described graph kernels typically decompose graphs into sets of subgraphs of a given structure, and then compare the sets of subgraphs in a pairwise fashion, for instance by counting the number of matching subgraphs. However, comparing all subgraphs between two graphs is an NP-hard problem, and therefore the types of subgraphs considered are generally limited [35]. Examples of this include random walks, shortest paths, and graphlet kernels (typically enumerating subgraphs of three to five nodes) [35]. When considering subgraphs of even a few nodes, the computational complexity of these kernels remains prohibitive for online place recognition with large and densely connected location graphs.

Alternative approaches consist of relabelling graphs to incorporate additional structural information into simpler structures. For example, in the Weisfeiler-Lehman (WL) kernel, node labels are updated to include the labels of their neighbours in an iterative scheme. At each iteration, each node is represented by a new label based on the combination of its own label and those of its neighbours, propagating information from further nodes. By augmenting node labels in this way, the WL kernel can achieve practical results by simply counting the number of matching labels between two graphs at each iteration. Computation therefore scales only linearly in the number of edges in the graph [29].

In this work, inspiration is taken from the WL kernel, attempting to find a way which is better suited to noisy observations. In the WL kernel, a single noisy node label or missing edge in the original graph will result in a difference in each further node label iteration which incorporates information from the noisy label, since only the number of exactly matching node labels between two graphs contribute to the final score. In our approach, rather than relabelling nodes with a single new value, node labels are augmented by a vector corresponding to their neighbourhood. The length of the vector is equal to the size of the label vocabulary (in this case the visual dictionary), and each element is weighted by the strength of the connecting edges in the covisibility graph. This concept is illustrated in Figures 2a and 2b. After one iteration of re-labelling, graph similarity can be measured by taking the dot product between the neighbourhood vectors of corresponding nodes in each graph (illustrated in Figure 2c), and summing the results. This process remains efficient, as only neighbourhood vectors from nodes with the same base-labels (original node label) are compared. In the case where more than one node in a graph have the same base-labels, comparison is done between all available pairs and the maximal value is used in the sum. As a result, nodes are not strictly matched one-to-one, but similarity scores remain symmetric by ensuring that the graph with fewer nodes of a given base-label is used to form the sets of node pairs for comparison. In order to obtain a normalized similarity measure between 0 and 1, the sum of neighbourhood comparisons is divided by the sum of total neighbourhood comparisons of each input graph to itself.

Figure 3: Example images from each of the datasets used for testing.

.

The final metric is therefore normalized, symmetric, and can be used to create a positive-definite kernel matrix between location graphs. The resulting complexity of the observation likelihood calculation is on the order $O(nd)$ (bounded by $O(n^2)$), where n is the number of common nodes, and d is the degree of the graph, likewise to the methods presented in [29, 32]. Furthermore, due to the sparse nature of visual word observations, a sparse implementation ensures that the complexity does not scale with the vocabulary size (typically on the order of tens or hundreds of thousand words). In addition, the approach inherently includes invariance to observation trajectories, view-points, and rotations, due to the underlying use of locally-invariant features and covisibility clustering. Query retrieval from the covisibility map using an inverted index also ensures that the overall complexity does not scale with the size of the map.

4. Experimental Validation

In order to validate and analyze the approach described in this paper, this section presents experiments on a number of benchmark datasets in varied environments. Evaluation is done on each dataset by incrementally processing monocular images in the sequence, updating the map at each step, and using the current location as a query into the current map. If a matching location already exists in the map, it is expected to be retrieved. The proposed method, referred to here as neighbourhood graph or nbhdGraph, is compared alongside the commonly applied FAB-MAP framework [12], and the word co-occurrence comparisons of [32], referred to here as wordGraph.

4.1. Test Sequences

A wide variety of datasets are used, in order evaluate the applicability and robustness of each approach. Example images from each dataset can be seen in Figure 3 to provide an idea of the different environments and image characteristics. Two of the sequences are from the KITTI visual odometry datasets [13] and provide examples of widely used, urban datasets. Specifically, the KITTI 00 and KITTI 05 sequences are used here, as they contain interesting loop-closures. The KTITI 00 sequence is 3.7km long, and the KITTI 05 is 2.2km long, both through suburban streets with good examples of perceptual aliasing. The sFly dataset [1] shows a very different environment. It contains imagery from a multi-copter flying over rubble with a downward-looking camera, and is about 350m long. Finally, the Narrow/Wide Angle datasets demonstrate a challenging localization scenario using different types of camera lenses. In these sequences a few streets are traversed once with a standard camera lens, and once with a wide-angle lens. A large portion of the two traversals overlap, but some areas also exist which are unique to one traversal. These sequences are tested twice, once in each order, providing a Narrow-Wide sequence and a Wide-Narrow sequence.

4.2. Test Configurations

Any parameter settings for each framework are set according to values documented in their respective publications [12, 32], with the exception of the masking parameter in FAB-MAP, as we found a value of 5 images provided better results. FAB-MAP was run using the Chow-Liu tree implementation, and a basic forward-moving motion model. Additionally, the visual word existence parameters were set to $P(z|e) = 0.39$ and $P(z|\bar{e}) = 0.005$. In all tested methods, the same feature detector, descriptors, and visual dictionary were used, namely 128-dimensional SURF descriptors and the 10987-word dictionary provided alongside the available FAB-MAP implementation. In both the imple-

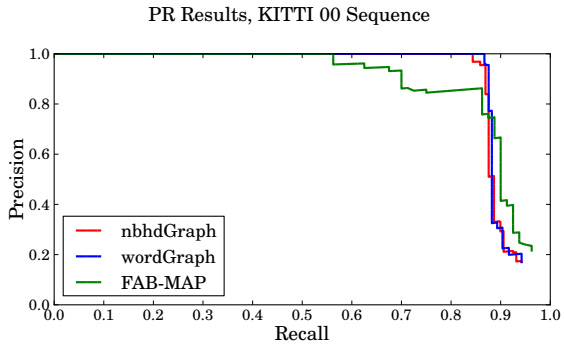


Figure 4: Precision-recall results on the KITTI 00 sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [12].

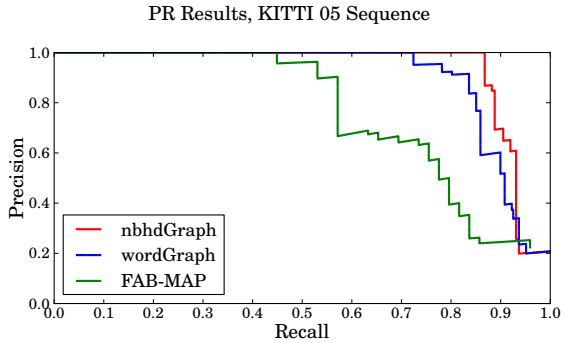


Figure 5: Precision-recall results on the KITTI 05 sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [12].

mentation of nbhdGraph and wordGraph, the same covisibility clustering parameter of 0.05 was used [32]. The effective $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$ was set to 0.002 after estimating it once from samples. Importantly, these parameters are kept constant through testing across datasets. The exception is for the more challenging Narrow/Wide Angle datasets, where configurations were allowed to change slightly. In the case of FAB-MAP the $P(z|\bar{e})$ parameter had to be increased to 0.05 to account for differences in observations, and the masking parameter had to be set to 30 images to account for tighter image spacing. In the nbhdGraph framework, the different extent of observations is simply handled by normalizing graph similarity scores by the sum of neighbourhood comparisons of only the common words between the two graphs, rather than all nodes (in a sense normalizing by the graph intersection rather than union).

Ground truth is given for most datasets by provided metric global position information. As a result, true location matches are those which lie within a given radius of the query position. For the KITTI datasets, a radius of 6m was used, while for the sFly dataset, a radius of 2m was used since the downward-looking images provide a more localized view. However, nearby images to the query (trivial

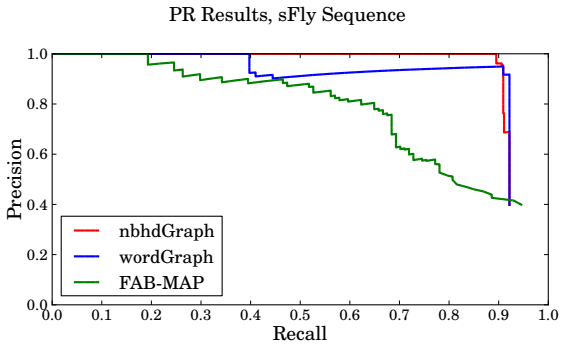


Figure 6: Precision-recall results on the sFly sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [12].

matches) cannot provide to true-positive match scores. For the Narrow/Wide datasets, metric position information was not available, and therefore ground truth was given by geometric feature matching between images which was then hand-corrected to remove false matches and fill in false negatives. Furthermore, for the Narrow/Wide datasets, only location matches from the opposite part of the sequence count toward true-positive matches, however images from the same part of the sequence can provide false-positive matches.

4.3. Results

Figures 4, 5, and 6 show precision-recall plots for the KITTI and sFly datasets as a threshold on the posterior probability $P(\mathcal{L}_i|\mathcal{Z}_q)$ is varied, comparing the proposed method (nbhdGraph), to the methods proposed in [32] (wordGraph), and [12] (FAB-MAP 2.0). All configuration parameters for each framework are kept the same for each of these datasets, and values are provided in Section 4.2. To give a notion of the complexity implications of the algorithm, our prototype code in python results in location comparisons which take $0.041 \pm 0.027s$ for the KITTI 05 dataset, with future capabilities for code optimization and parallelization.

In general, the results show improvements over the state-of-the-art, most notably against the FAB-MAP framework which incorporates far less spatial information about the visual features than the other two methods. Although the results are not strictly better than those from the wordGraph method, they are especially meaningful due to the fact that explicit posterior normalization calculations are not required, therefore simplifying computation and removing the dependency on previously acquired sample locations.

Precision-recall plots for the Narrow-Wide and Wide-Narrow angle sequences are shown in Figure 7. From these plots, one can see how each method can handle heterogeneous observations. Comparing the two plots, results for

RANSAC inliers: 19%
FAB-MAP: $P(\mathcal{L}_i|\mathcal{Z}_q) = 7.6^{-4}\%$
wordGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 98\%$
nbhdGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 96\%$

RANSAC inliers: 47%
FAB-MAP: $P(\mathcal{L}_i|\mathcal{Z}_q) = 7.2^{-7}\%$
wordGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 94\%$
nbhdGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 95\%$

RANSAC inliers: 24%
FAB-MAP: $P(\mathcal{L}_i|\mathcal{Z}_q) = 5.0^{-7}\%$
wordGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 75\%$
nbhdGraph: $P(\mathcal{L}_i|\mathcal{Z}_q) = 38\%$

Figure 8: Example true and false-positive matches from the KITTI 05 dataset. Each column shows one example, where the query locations are shown in the top row in blue, with a candidate location below. True matches are designated in green, while false matches are designated in red. These examples represent some difficult locations for place recognition.

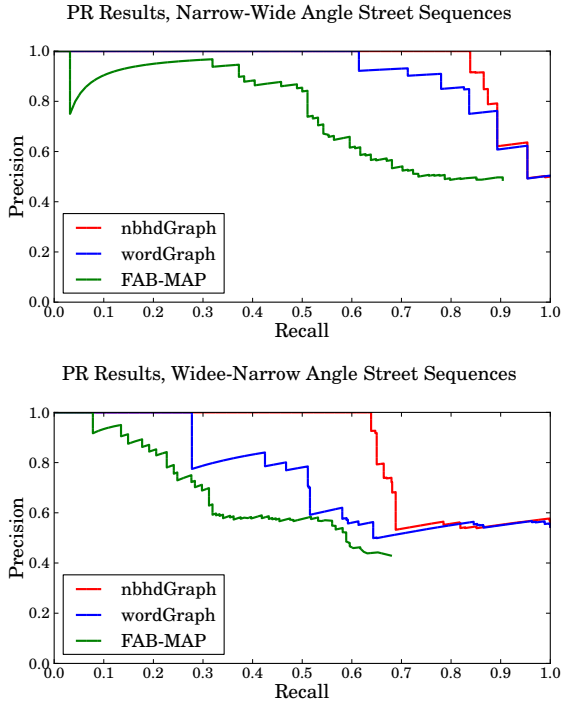
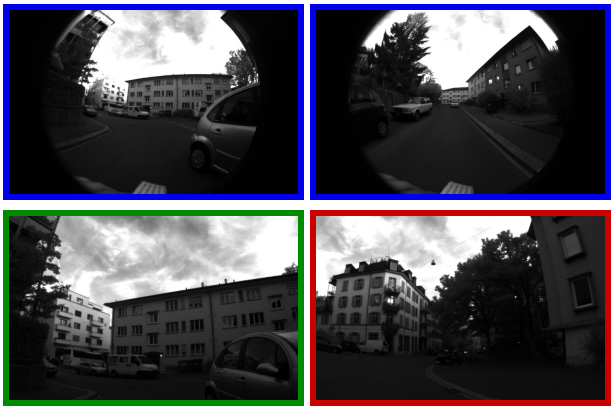


Figure 7: Precision-recall results on the Narrow-Wide and Wide-Narrow Angle sequences for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [12].



RANSAC inliers: 0%
FAB-MAP:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 1.0^{-3}\%$
wordGraph:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 46\%$
nbhdGraph:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 95\%$

RANSAC inliers: 45%
FAB-MAP:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 5.7^{-5}\%$
wordGraph:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 22\%$
nbhdGraph:
$P(\mathcal{L}_i|\mathcal{Z}_q) = 90\%$

Figure 9: Example true and false-positive matches from the Narrow-Wide dataset. Each column shows one example, where the query locations are shown in the top row in blue, with a candidate location below. True matches are designated in green, while false matches are designated in red. These examples represent some difficult locations for place recognition

the Narrow-Wide sequence are better than the Wide-Narrow sequence. This can be explained by the fact that in the first case, the more complete wide-field-of-view images are used to query the narrow-field-of-view images, making retrieval from the covisibility map more reliable in the case of nbhdGraph and wordGraph, and the observation model parameters more applicable in the case of FAB-MAP.

Figure 8 shows three representative examples of difficult locations for visual place recognition from the KITTI 05 sequence. In each example a query and a candidate location are depicted, and scores corresponding to various comparison methods are shown below. Generally speaking, the nbhdGraph method tends to localize more precisely than the

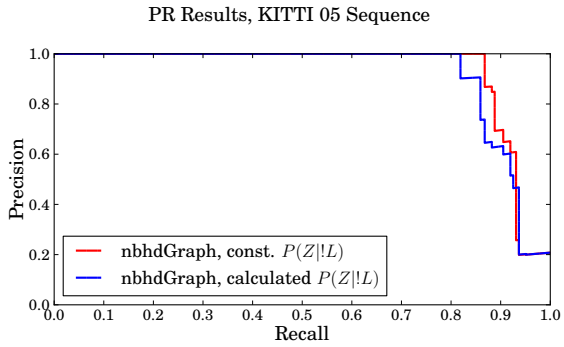


Figure 10: Precision-recall results on the KITTI 05 sequence, comparing the results using a constant value for $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$, and one which calculated $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ using the dataset ground-truth.

wordGraph method, providing better resistance to perceptual aliasing and more tightly located location matches, but possibly reducing recall in locations like the boundaries of overlapping areas. From this figure, one can also see problems with the posterior normalization method of the FAB-MAP framework (presented in [12]), as the posterior probability mass is distributed among all nearby locations in the map, resulting in unintuitive values in most locations.

Similarly, Figure 9 shows examples of difficult areas from the Narrow-Wide dataset. Here one can see that differentiating between true and false matches is more challenging since landmark detection and appearance tends to differ largely between the two camera lenses. The second example of Figure 9 is challenging because the buildings and foliage produce similar features, and in particular, almost all detected features came from the trees in this case, leaving degenerate location graphs.

The validity of the normalization scheme proposed in Section 3.2.1 was also investigated experimentally. In order to do so, the results obtained with a constant value for $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ were compared to results obtained from conducting normalization using the ground truth data, and can be seen in Figure 10. Using the global position information, $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ was calculated for each query, by comparing the given query observation to every other location in the map. It turns out that this normalization using ground truth position information even produces slightly worse results than the proposed constant $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ approach. This could in part be due to the fact segmenting out the query location from the map is non-trivial (for example, distant objects may be observed over large areas). Furthermore, $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ should become more stable as the size of the map increases, and therefore it is possible that not enough locations were used in the estimation. These results confirm the difficulty in accurately normalizing posterior probabilities, and provide support for the assumption that $P(\mathcal{Z}_q|\overline{\mathcal{L}}_i)$ can be approximated as constant.

5. Conclusion

This paper has introduced a probabilistic place recognition framework which combines visual and spatial information in a flexible yet discriminative manner. Efficient approaches of graph comparison have been explored for calculating similarity between locations represented by their corresponding covisibility graphs. As a result, a novel observation likelihood formulation has been developed which analyzes the similarity of local neighbourhoods within each graph. The resulting graph comparison method can be formulated as a symmetric and positive-definite graph kernel, additionally providing the potential for further uses in learning algorithms such as semantic understanding of location graphs.

The inclusion of structural information from the covisibility graph allows the inference algorithm to disambiguate between repetitive and self-similar patterns in the environment using only noisy visual information. Consequently, this allows for a more efficient posterior normalization scheme due to the fact that the average probability of an observation coming from a random location can be effectively estimated as a constant value. This not only reduces the overall computational complexity of the approach, but also eliminates the dependence on detailed sample locations or prior map information that most state-of-the-art approaches rely on. The presented method is therefore well suited to applications which involve exploration of large, unconstrained environments. Experiments on several challenging datasets validate the reliability and applicability of the approach in a number of different environments.

Future work includes extending the application of the framework to long-term place recognition in dynamic environments, and tasks such as semantic scene understanding, or object recognition. In addition, the probabilistic framework could include additional sensory information and more sophisticated location priors based on a motion model. Furthermore, since the approach remains general with respect to the underlying features, visual words could be replaced or used in conjunction with other, possibly higher-level features such as objects.

Acknowledgments

The research leading to these results has received funding from the Swiss National Science Foundation through the National Centre of Competence in Research Robotics. We would also like to thank the reviewers for their valuable feedback.

References

[1] M. W. Achtelik, S. Lynen, S. Weiss, L. Kneip, M. Chli, and R. Siegwart. Visual-inertial SLAM for a small helicopter in large outdoor environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[2] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski. Building Rome in a day. *Communications of the Association for Computing Machinery (ACM)*, 54(10):105–112, 2011.

[3] A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast retina keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[4] A. Babenko and V. Lempitsky. The inverted multi-index. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(6):1247–1260, June 2015.

[5] F. Bach. Graph kernels between point clouds. In *International Conference on Machine Learning (ICML)*, 2008.

[6] L. Bai, L. Rossi, H. Bunke, and E. R. Hancock. Attributed graph kernels using the Jensen-Tsallis q-differences. In *Machine Learning and Knowledge Discovery in Databases*, pages 99–114. Springer, 2014.

[7] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics Automation Magazine*, 13(3):108–117, September 2006.

[8] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

[9] R. O. Castle, G. Klein, and D. W. Murray. Wide-area augmented reality using camera tracking and mapping in multiple regions. *Computer Vision and Image Understanding (CVIU)*, 115(6):854–867, 2011.

[10] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, 2004.

[11] M. Cummins. *Probabilistic localization and mapping in appearance space*. PhD thesis, University of Oxford, Balliol College, October 2009.

[12] M. Cummins and P. Newman. Appearance-only SLAM at large scale with FAB-MAP 2.0. *The International Journal of Robotics Research (IJRR)*, 30(9):1100–1123, August 2011.

[13] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] Z. Harchaoui and F. Bach. Image classification with segmentation graph kernels. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[15] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox. RGB-D mapping: Using kinect-style depth cameras for dense 3D modeling of indoor environments. *The International Journal of Robotics Research (IJRR)*, 31(5):647–663, 2012.

[16] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision (ECCV)*. Springer, 2008.

[17] H. Jegou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[18] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(1):117–128, 2011.

[19] E. Johns and G.-Z. Yang. Generative methods for long-term place recognition in dynamic scenes. *International Journal of Computer Vision (IJCV)*, 106(3):297–314, 2014.

[20] J. Lim, J.-M. Frahm, and M. Pollefeys. Online environment mapping using metric-topological maps. *The International Journal of Robotics Research (IJRR)*, 31(12):1394–1408, October 2012.

[21] C. Linegar, W. Churchill, and P. Newman. Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015.

[22] D. G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision (ICCV)*, 1999.

[23] S. Lynen, T. Sattler, M. Bosse, J. Hesch, M. Pollefeys, and R. Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems (RSS)*, 2015.

[24] C. Mei, G. Sibley, and P. Newman. Closing loops without places. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[25] M. J. Milford. Vision-based place recognition: how low can you go? *The International Journal of Robotics Research (IJRR)*, 32(7):766–789, June 2013.

[26] M. Mohan, D. Gálvez-López, C. Monteleoni, and G. Sibley. Environment selection and hierarchical place recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[27] R. Paul and P. Newman. FAB-MAP 3D: Topological mapping with spatial and visual appearance. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2010.

[28] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A discriminative approach to robust visual place recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[29] N. Shervashidze, P. Schweitzer, E. J. Van Leeuwen, K. Mehlhorn, and K. M. Borgwardt. Weisfeiler-Lehman graph kernels. *The Journal of Machine Learning Research*, 12:2539–2561, 2011.

[30] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, 2003.

[31] E. Stumm, C. Mei, and S. Lacroix. Building location models for visual place recognition. *The International Journal of Robotics Research (IJRR)*, 35(4):334–356, April 2016.

[32] E. Stumm, C. Mei, S. Lacroix, and M. Chli. Location graphs for visual place recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[33] N. Sünderhauf, P. Neubert, and P. Protzel. Are we there yet? Challenging seqslam on a 3000 km journey across all four seasons. In *Workshop on Long-Term Autonomy, at IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[34] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi. Visual place recognition with repetitive structures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

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

[36] F. Zhou and F. De la Torre. Deformable graph matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.