



**HAL**  
open science

## Location models for visual place recognition

Elena Stumm

► **To cite this version:**

Elena Stumm. Location models for visual place recognition. Robotics [cs.RO]. Université Toulouse III Paul Sabatier, 2015. English. NNT: . tel-01376134

**HAL Id: tel-01376134**

**<https://laas.hal.science/tel-01376134>**

Submitted on 7 Nov 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# THÈSE

En vue de l'obtention du

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

---

**Présentée et soutenue par :**  
**Elena STUMM**

le 23.11.2015

**Titre :**

Location models for visual place recognition

(Modèles probabilistes pour la reconnaissance visuelle de lieux)

---

**École doctorale et discipline ou spécialité :**

EDSYS : Robotique 4200046

**Unité de recherche :**  
LAAS-CNRS

**Directeur/trice(s) de Thèse :**

Simon LACROIX  
Christopher MEI

**Jury :**

David FILLIAT  
José NEIRA  
Ingmar POSNER  
Patrick DANÈS



DOCTORAL THESIS

Université de Toulouse,  
Université Paul Sabatier

and

LAAS-CNRS

Laboratory for Analysis and Architecture of Systems

---

# Location Models For Visual Place Recognition

*Author:*

Elena Stumm

*Supervisors:*

Simon Lacroix,  
Christopher Mei

*Research Group:*

Robotics & Interactions (RIS)

*Institute:*

LAAS-CNRS, Toulouse

*Reviewers:*

David Filliat,  
José Neira

*Jury:*

Ingmar Posner,  
Patrick Danès,  
Christopher Mei,  
Simon Lacroix

May 2016



this page intentionally left blank

page laissée intentionnellement vide

## Abstract

This thesis deals with the task of appearance-based mapping and place recognition for mobile robots. More specifically, this work aims to identify how location models can be improved by exploring several existing and novel location representations in order to better exploit the available visual information. Appearance-based mapping and place recognition presents a number of challenges, including making reliable data-association decisions given repetitive and self-similar scenes (perceptual aliasing), variations in view-point and trajectory, appearance changes due to dynamic elements, lighting changes, and noisy measurements. As a result, choices about how to model and compare observations of locations is crucial to achieving practical results. This includes choices about the types of features extracted from imagery, how to define the extent of a location, and how to compare locations.

Along with investigating existing location models, several novel methods are developed in this work. These are developed by incorporating information about the underlying structure of the scene through the use of covisibility graphs which capture approximate geometric relationships between local landmarks in the scene by noting which ones are observed together. Previously, the range of a location generally varied between either using discrete poses or loosely defined sequences of poses, facing problems related to perceptual aliasing and trajectory invariance respectively. Whereas by working with covisibility graphs, scenes are dynamically retrieved as clusters from the graph in a way which adapts to the environmental structure and given query.

The probability of a query observation coming from a previously seen location is then obtained by applying a generative model such that the uniqueness of an observation is accounted for. Behaviour with respect to observation errors, mapping errors, perceptual aliasing, and parameter sensitivity are examined, motivating the use of a novel normalization scheme and observation likelihoods representations. The normalization method presented in this work is robust to redundant locations in the map (from missed loop-closures, for example), and results in place recognition which now has sub-linear complexity in the number of locations in the map. Beginning with bag-of-words representations of locations, location models are extended in order to include more discriminative structural information from the covisibility map. This results in various representations ranging between unstructured sets of features and full graphs of features, providing a tradeoff between complexity and recognition performance.





## Résumé

Cette thèse traite de la cartographie et de la reconnaissance de lieux par vision en robotique mobile. Les recherches menées visent à identifier comment les modèles de localisation peuvent être améliorés en enrichissant les représentations existantes afin de mieux exploiter l'information visuelle disponible. Les problèmes de la cartographie et de la reconnaissance visuelle de lieux présentent un certain nombre de défis : les solutions doivent notamment être robustes vis-à-vis des scènes similaires, des changements de points de vue de d'éclairage, de la dynamique de l'environnement, du bruit des données acquises. La définition de la manière de modéliser et de comparer les observations de lieux est donc un élément crucial de définition d'une solution opérationnelle. Cela passe par la spécification des caractéristiques des images à exploiter, par la définition de la notion de lieu, et par des algorithmes de comparaison des lieux.

Dans la littérature, les lieux visuels sont généralement définis par un ensemble ou une séquence d'observations, ce qui ne permet pas de bien traiter des problèmes de similarité de scènes ou de reconnaissance invariante aux déplacements. Dans nos travaux, le modèle d'un lieu exploite la structure d'une scène représentée par des graphes de covisibilité, qui capturent des relations géométriques approximatives entre les points caractéristiques observés. Grâce à cette représentation, un lieu est identifié et reconnu comme un sous-graphe.

La reconnaissance de lieux exploite un modèle génératif, dont la sensibilité par rapport aux similarités entre scènes, aux bruits d'observation et aux erreurs de cartographie est analysée. En particulier, les probabilités de reconnaissance sont estimées de manière rigoureuse, rendant la reconnaissance des lieux robuste, et ce pour une complexité algorithmique sous-linéaire en le nombre de lieux définis. Enfin les modèles de lieux basés sur des sacs de mots visuels sont étendus pour exploiter les informations structurelles fournies par le graphe de covisibilité, ce qui permet un meilleur compromis entre la qualité et la complexité du processus de reconnaissance.



# Contents

<b>1</b>	<b>Premise</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Overview . . . . .	3
1.3	Contributions . . . . .	6
1.4	Related Publications and Presentations . . . . .	7
<b>2</b>	<b>Background</b>	<b>9</b>
2.1	Introduction . . . . .	9
2.2	Robot Navigation and Mapping . . . . .	9
2.2.1	The SLAM Problem . . . . .	9
2.2.2	Appearance-Based Localization and Mapping . . . . .	12
2.3	Visual Place Recognition with Global Image Attributes . . . . .	13
2.3.1	Direct Image Comparisons . . . . .	14
2.3.2	Global Image Descriptors . . . . .	15
2.3.3	Place Recognition with Global Image Attributes . . . . .	16
2.4	Visual Place Recognition with Bags of Features . . . . .	17
2.4.1	Interest Point Detection and Description . . . . .	18
2.4.2	Visual Words . . . . .	20
2.4.3	Information Retrieval . . . . .	22
2.4.4	Place Recognition with Visual Words . . . . .	24
2.5	Defining the Notion of Places . . . . .	25
2.5.1	Working with Single Images . . . . .	26
2.5.2	Working with Sets of Images . . . . .	26
2.5.3	Covisibility Graphs . . . . .	27
2.6	Visual Place Recognition with Geometric Features . . . . .	28
2.6.1	Weak Geometric Constraints . . . . .	28

2.6.2	3D Geometric Constraints . . . . .	29
2.7	Outlook . . . . .	30
<b>3</b>	<b>Covisibility Mapping</b>	<b>33</b>
3.1	Introduction . . . . .	33
3.2	Building the Covisibility Graph . . . . .	33
3.2.1	Feature Extraction . . . . .	34
3.2.2	Graph Creation . . . . .	34
3.2.3	Discussion on Graph Structure . . . . .	35
3.3	Identifying Places . . . . .	36
3.3.1	Inverted Index . . . . .	36
3.3.2	Retrieval by Landmark Covisibility . . . . .	37
3.3.3	Retrieval by Graph Clustering . . . . .	38
<b>4</b>	<b>Modelling Locations with Bags of Words</b>	<b>41</b>
4.1	Introduction . . . . .	41
4.2	Modelling Scene Elements . . . . .	42
4.3	Estimating Observation Likelihoods . . . . .	43
4.4	Modelling Visual Observations . . . . .	44
4.4.1	Observation Given Existence . . . . .	44
4.4.2	Existence Given Observation . . . . .	45
4.4.3	Empirical Evaluation of Observation Models . . . . .	45
4.5	Normalization using Sample Locations . . . . .	47
4.5.1	Discussion on Normalization Models . . . . .	48
4.5.2	Empirical Evaluation of Normalization Models . . . . .	49
4.6	Location Priors . . . . .	51
4.7	Sampling . . . . .	51
4.8	Experimental Evaluation and Results . . . . .	53
4.8.1	Comparison with State-of-the-Art . . . . .	53
4.8.2	Investigating Relative Scoring Methods . . . . .	57
4.8.3	Investigating Trajectory Invariance . . . . .	59
4.8.4	Investigating Graph Clustering . . . . .	62
4.9	Outlook . . . . .	62

<b>5</b>	<b>Modelling Locations with Graphs of Words</b>	<b>65</b>
5.1	Introduction . . . . .	65
5.2	On the Complexity of Graph Comparison . . . . .	66
5.3	Location Graphs of Landmarks . . . . .	67
5.4	Graph Kernels . . . . .	69
5.4.1	Primer on Graph Kernels . . . . .	69
5.4.2	Applications to Location Graphs . . . . .	71
5.5	Location Graphs of Visual Words . . . . .	73
5.5.1	Reduction to Visual Word Representation . . . . .	74
5.5.2	Estimating Observation Likelihoods . . . . .	75
5.5.3	Sampling . . . . .	77
5.5.4	Relation to $tf \times idf$ . . . . .	77
5.5.5	Relation to Graph Kernels . . . . .	78
5.6	Experimental Evaluation of Visual Word Location Graphs . . . . .	78
5.6.1	Comparison with State-of-the-Art . . . . .	78
5.6.2	Investigating Weighting Schemes . . . . .	81
5.6.3	Investigating Behaviour with Respect to Noise . . . . .	82
5.7	Outlook . . . . .	83
<b>6</b>	<b>Closing Remarks</b>	<b>87</b>
6.1	Conclusion . . . . .	87
6.2	Extensions and Open Questions . . . . .	88
<b>A</b>	<b>Implementation</b>	<b>95</b>
A.1	CovisMap framework . . . . .	95
A.2	Various Parameter Settings . . . . .	97
<b>B</b>	<b>Datasets</b>	<b>99</b>
<b>C</b>	<b>Precision-Recall Metrics</b>	<b>103</b>
<b>D</b>	<b>Preliminary Version of CVPR16 Paper on Graph Kernels</b>	<b>105</b>
<b>E</b>	<b>French Summary of Manuscript</b>	<b>117</b>
	<b>Acknowledgements</b>	<b>147</b>

**Bibliography**

**148**

# Chapter 1

## Premise

### 1.1 Motivation

*“Essentially, all models are wrong, but some are useful.”*

*– George E.P. Box*

Long-term autonomous navigation in unknown environments is becoming increasingly important for a variety of mobile robotic platforms and applications. For this purpose, the navigation platform must be robust to errors, with localization working even in the case of unexpected, dynamic, and possibly self-similar environments. One important requirement for reliably maintaining error bounds on a robot’s position during simultaneous localization and mapping (SLAM) is visual place recognition for performing loop-closure, due to its accessibility and ability to work in a wide range of environments [Cummins and Newman, 2011, Maddern et al., 2012]. Without loop-closure, the robot’s understanding of the world quickly diverges from the true state as small errors accumulate over time, becoming inadequate for navigation (see Figure 1.1, for example) [Cummins, 2009]. In addition to loop-closure, place recognition can be an important function, as a stepping stone towards semantic mapping and scene understanding, as well as map fusion or multi-robot navigation. Incorrect place recognition can cause large mapping errors in a SLAM or map fusion context, or totally erroneous reasoning based on incorrect scene interpretation, emphasizing the importance of avoiding false-positive associations. As a result, the goal in achieving this task is to detect as many correct location associations as possible, without returning any false ones. This problem re-



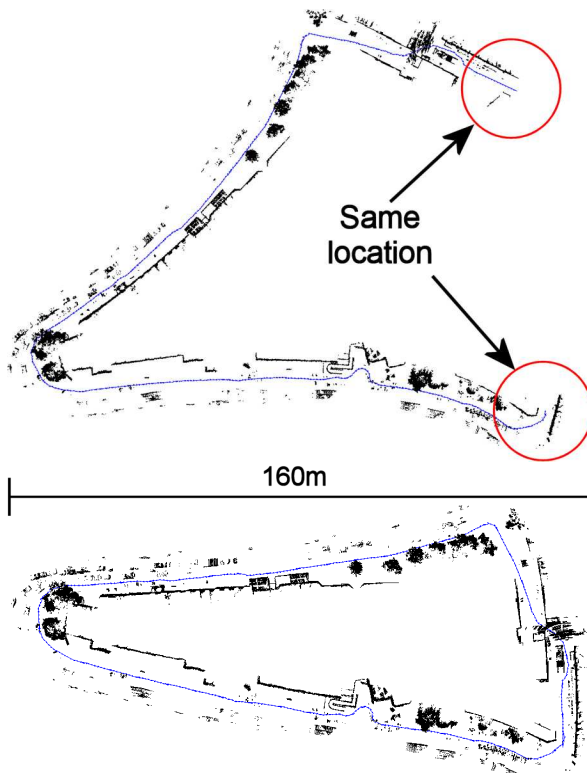


Figure 1.1: As small errors accumulate, mapping diverges from the true state and must be corrected by loop-closure based data associations. [Cummins, 2009]

mainly challenging due to the difficulty of representing places in a way which is invariant to trajectory and environment changes, discriminative enough to distinguish repetitive features, and allows for efficient inference. Some examples of the kinds of visual difficulties that can occur are presented in Figures 1.2 and 1.3.

The problem being addressed in this thesis is that of evaluating whether a mobile robot is revisiting a place in which it has already been, or is in a previously unknown location, by analysing the visual appearance of the current scene. Ultimately, this should be achieved with minimal assumptions about the behaviour of the environment and robot, in order to remain applicable in the general case. Additionally, this work relies exclusively on visual information captured through a camera. Many different ways of modelling visual observations for the task of place recognition already exist. However, each model comes with its own simplifications and assumptions, leading to various advantages and disadvantages. This thesis sets out to investigate the behaviour of various commonly used models, along with several new ones that are proposed in this work. By making careful model choices when representing places, the aforementioned challenges related to making reliable data-association decisions in the case of repetitive and self-similar scenes (referred to as perceptual aliasing), as well as appearance changes due to dynamic elements, variations in view-point, and lighting changes are simplified.



Figure 1.2: Examples of difficult recognition tasks due to environmental changes.

In this work, we propose a Bayesian framework which uses feature-based location models built from sets of covisible scene elements for evaluating place recognition. Location models are built up by grouping features based on visual connectivity, allowing for more context than single images, while maintaining invariance to trajectory variations. Additionally, working with feature-based models provides a built-in robustness towards view-point and lighting changes through the use of locally invariant feature descriptors. Incorporating such location models into a Bayesian framework can allow for an integrated and probabilistically sound way to handle perceptual aliasing, dynamic elements, and uncertainty about detections.

The next sections provide a more detailed overview of what concepts will be presented within this thesis, and what scientific contributions have been made.

## 1.2 Overview

This thesis focuses on visual place recognition for mobile robots, primarily building on prior concepts given in [Mei et al., 2010, Cummins and Newman, 2008] by establishing generative location models using covisibility maps. This section provides a brief overview of the structure of the thesis and the place recognition fundamentals which are presented. For more understanding of the background work and terminology, please refer to Chapter 2.

Locations are commonly represented by visual landmarks which are detected within the scene. These landmarks are then described by visual words which are quantized versions of the original feature descriptors, in order to ease analysis, and reduce memory and complexity requirements. As a result, locations can be retrieved and compared using ideas



Figure 1.3: Examples of difficult recognition tasks due to self-similar structures in the environment (perceptual aliasing).

inspired from the text-document retrieval field [Sivic and Zisserman, 2003, Manning et al., 2008, Cummins and Newman, 2008, Angeli et al., 2008b, Botterill et al., 2011]. Hence relatively simple but sound probabilistic models can be created and used to achieve effective place recognition, such as in the work of Cummins and Newman [2008]. The idea is that by modelling each location by its set of visual words, the representation is invariant to a certain degree of lighting and view-point changes, while the probabilistic model incorporates notions of noisy observations and dynamic elements. However, existing works have trouble defining the extent of a location, generally relying on an arbitrary discretization of the space, either defined by single image poses or pre-defined lengths of image sequences. In addition, these models do not incorporate 3D information about the location’s landmarks, due to the difficulty in using the additional information in an efficient way. The above references work with bag-of-words models which do not take into account any structural relationships between landmarks.

In this work, we stress the importance of these structural relationships on multiple levels. Firstly, it is useful for defining the extent of a given place, and can be used for location retrieval [Stumm et al., 2013]. Secondly, it is an important source of discriminative information, helping to reduce perceptual aliasing and false-positive matches [Stumm et al., 2015b]. We argue that these relationships can be incorporated in a simple and intuitive way, which achieves a more truthful and continuous representation of the environment. This is done using a covisibility map which is constructed as the robot explores the environment, by noting

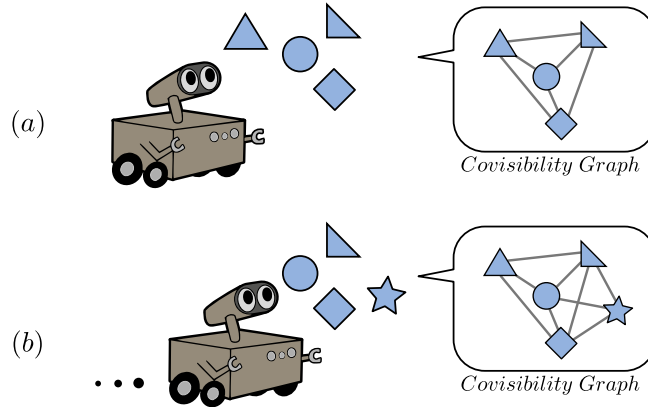


Figure 1.4: As the robot moves, it makes observations, detects landmarks, and notes which ones were seen together in a graph structure. Here you can see a simple example of two steps as a robot moves forward through the environment, and the resulting covisibility graph.

which landmarks are observed together in a graph structure [Mei et al., 2010]. The basic mapping concept is depicted in Figure 1.4. This covisibility graph is able to implicitly capture the underlying structure of the environment without modelling complete 3D position information; as landmarks belonging to the same structures will be co-observed more often than landmarks belonging to different structures. From the covisibility map, relevant “*virtual locations*” which resemble a given query can be retrieved as clusters [Stumm et al., 2015a]. Places are now defined using direct properties of the environment (landmarks), and become less dependent on variations in trajectory while eliminating the problem of pose selection.

Once the virtual locations are retrieved, a probabilistic framework is used to identify any potential matches between the query and previously seen locations. Development of a proper generative model is a key factor for providing useful results, especially in challenging environments, and is therefore the main focus of this thesis. For instance, a rigorous probabilistic method allows for inherent confidence thresholds, and can handle problematic situations such as perceptual aliasing by understanding the likelihood of scene elements. This relies on a careful treatment of probability normalization, done here using a set of sample locations which are used to model the unknown world. The method presented in this thesis allows the system to search for all matching locations, rather than the one most probable, giving the potential to cope with erroneous maps which may contain more than one instance of the same location (for example where loop-closures were missed). The developed models also improve the stability with respect to parameter selections, in comparison to previous work. The resulting posterior probabilities represent an intuitive measure of place similarity, with values varying smoothly as a queried location is traversed. This thesis develops var-

ious ways of calculating observation likelihoods for locations, utilizing varying amounts of structural and spatial information from the covisibility graph, and examines the results.

The remainder of the manuscript is divided into five chapters: Chapter 2 gives a broad overview of background information related to the research done during this thesis, Chapter 3 introduces the notion of covisibility maps and how locations are retrieved as graphs of covisible landmarks, Chapters 4 and 5 then discuss and develop several generative probabilistic models for visual place recognition, first using bag-of-words representations and then using graph-based representations, and finally Chapter 6 provides an outlook on the impact of the work and addresses several topics for future work. In addition, since the theoretical developments of this thesis are largely backed by experimental evaluation, many test results will be presented throughout the thesis for clarity. In order to understand the implementation and testing procedure, appendices depict implementation choices, (Appendix A), the used datasets (Appendix B), and the precision-recall metrics (Appendix C).

### 1.3 Contributions

The work done during this thesis has resulted in the following contributions to the field:

- presentation and analysis of a unified framework for defining, retrieving, and recognizing places in a robust manner
- improved generative models for appearance-based place recognition, including
  - improved stability with respect to parameter settings for visual word observation models
  - a novel normalization scheme allowing for redundant locations in the map and sub-linear performance with respect to locations in the map
  - an introduction of efficient, structured comparison techniques for location graphs, diminishing the effect of perceptual aliasing
- thorough evaluation and analysis of behaviour and performance characteristics compared alongside the state-of-the-art

## 1.4 Related Publications and Presentations

- E. Stumm, C. Mei, S. Lacroix, “Recognizing Places using Covisibility Maps,” in IEEE International Workshop on Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM), 2013
- E. Stumm, C. Mei, and S. Lacroix, “Probabilistic place recognition with covisibility maps,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 2013. [Stumm et al., 2013]
- E. Stumm, C. Mei, and S. Lacroix, “Building location models for visual place recognition,” in The International Journal of Robotics Research (IJRR), 2015. [Stumm et al., 2015a]
- E. Stumm, C. Mei, S. Lacroix, and M. Chli, “Location Graphs for Visual Place Recognition,” IEEE International Conference on Robotics and Automation (ICRA), Seattle, USA, 2015. [Stumm et al., 2015b]
- E. Stumm, C. Mei, S. Lacroix, M. Hutter, J. Nieto and R. Siegwart, “Robust Visual Place Recognition with Graph Kernels,” Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA, 2016. [Stumm et al., 2016]



## Chapter 2

# Background

### 2.1 Introduction

The visual place recognition problem, whether semantic or metric, is in general a problem of understanding patterns in the environment for the purpose of data association. Solving this problem typically draws from research in varied fields, such as estimation, computer vision, probability theory, decision theory, and graph theory to name a few. As a result, a wide array of approaches exist, and this chapter aims to identify the main concepts and approaches, before providing a prospective outlook at how this thesis continues from existing work.

### 2.2 Robot Navigation and Mapping

Visual place recognition has many valuable applications within the field of mobile robot navigation and mapping. This includes building topological maps in the space of appearance, performing loop-closure during SLAM tasks, aiding in the recovery from the kidnapped robot problem, combining maps and observations from multiple robots, localizing with respect to a documented database of imagery (such as Google Streetview), and making important inferences about the environment. This section will provide an overview of these tasks, and the respective role that place recognition plays.

#### 2.2.1 The SLAM Problem

Simultaneous Localization And Mapping (SLAM) is one of the fundamental challenges of autonomous mobile robots. This task requires the robot(s) to navigate through a previously unknown environment, making observations in order to simultaneously build a map of the en-



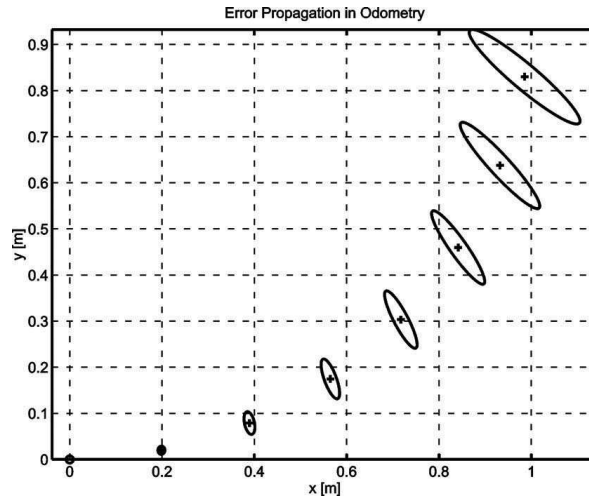


Figure 2.1: As the robot moves, the pose uncertainty (represented by ellipses) from odometry estimation grows. [Siegwart et al., 2011]

environment and localize within the map. Numerous and varied approaches exist, with some of the main solutions to be listed here. However it should be noted that one universal paradigm to all these solutions, is that of incorporating uncertainty. Emphasis should be placed on the importance of probabilistic methods when dealing with uncertainty. Uncertainty lies at the core of why SLAM is a difficult problem. Noisy measurements of the robot’s own ego-motion, as well as observations of features in the environment introduce errors which compound over time, increasing the uncertainty of each subsequent pose and map estimation. Errors in pose estimation lead to further errors in the map estimation, and errors in the map estimation analogously lead to further errors in the pose estimation.

Figure 2.1 shows an example of how the robot’s position uncertainty increases, as it estimates its pose from odometry measurements. Additionally, Figure 2.2 shows an illustration of the SLAM problem, mapping the environment via a set of important features in the environment, referred to as landmarks. The graphical model of the Bayes network which corresponds to this illustration is given in Figure 2.3. SLAM aims to either solve the problem of maintaining the posterior distribution of the current robot state and map state:  $P(x_k, m_1, \dots, m_N | z_1, \dots, z_k, u_1, \dots, u_k)$ , or the posterior distribution of the entire trajectory and map state:  $P(x_1, \dots, x_k, m_1, \dots, m_N | z_1, \dots, z_k, u_1, \dots, u_k)$ . In large environments, solving the full posterior is typically not feasible, and therefore the problem is usually tackled using a number of assumptions and simplifications. Commonly used approaches include extended Kalman filter (EKF) SLAM which works with local linearizations and Gaussian covariance distributions [Smith et al., 1990], particle filter SLAM (FastSLAM) which approximates the distribution of the position of the robot by a set of particles [Montemerlo et al.,

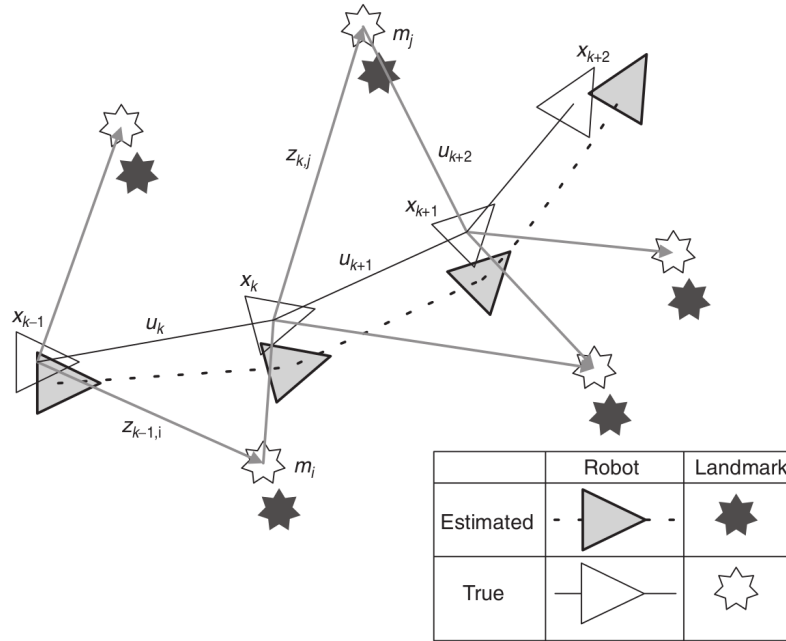


Figure 2.2: Illustration of the SLAM problem, showing the relative measurements of robot poses and landmarks, with robot states (position and orientation)  $x_k$ , control inputs  $u_k$ , landmark locations  $m_i$ , and measurements of landmarks from a given pose  $z_{ik}$ . Errors from both odometry and landmark observations lead to uncertainty in both localization and the map. [Durrant-Whyte and Bailey, 2006]

2003], and pose graph methods which exploit sparsity to perform optimization on the full graph structure [Dellaert and Kaess, 2006, Kümmerle et al., 2011]. We will not discuss the state estimation problem in more detail in this thesis, but it is important to take note of the structure of the problem. In this work we will introduce the notion of covisibility graphs in Section 2.5.3, and later in more detail in Chapter 3, highlighting the parallel relationship with the structure required for SLAM in Section 3.2.3. For more context about solving the SLAM problem, readers may refer to [Bailey and Durrant-Whyte, 2006, Durrant-Whyte and Bailey, 2006, Siegwart et al., 2011, Thrun et al., 2005].

In order to maintain reasonable error bounds over long trajectories, so-called loop-closures need to be detected, in order to perform data-association between several observations. It is by re-observing features in the environment and reasoning on multiple measurements, that errors can be reduced. This concept is crucial to all SLAM implementations, and generally provided through place recognition systems, and is therefore of particular interest in this thesis. Figures 1.1 and 2.4 show examples of mapping results with and without incorporating loop-closures. In addition to loop-closures, place recognition can enable recovery from the kidnapped robot problem (an otherwise unobservable displacement of the robot), as well as enabling data association between maps from multiple robots or mapping sessions.

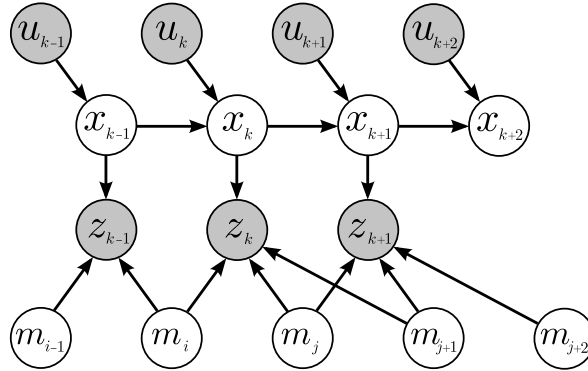


Figure 2.3: The Bayes network graph corresponding to the SLAM illustration of Figure 2.2. Observable variables are shaded grey, while hidden variables are white.

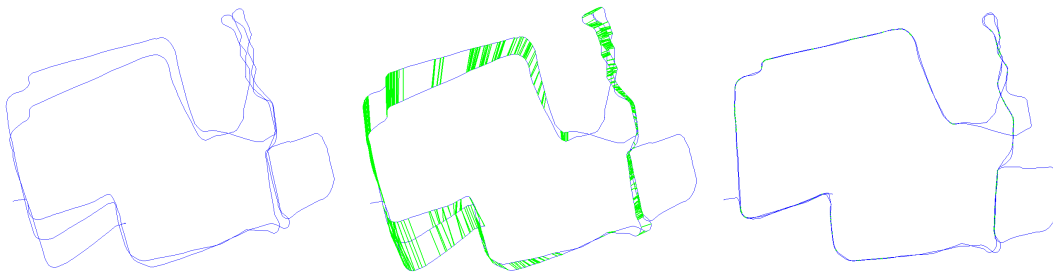


Figure 2.4: An example of how loop-closures can drastically improve estimation results. The figure on the left shows results without using loop-closures, the centre figure shows loop-closures detected using visual place recognition, and the figure on the right shows the updated results from incorporating loop-closures. [Cummins, 2009]

### 2.2.2 Appearance-Based Localization and Mapping

Vision has a long history in robotics, and the variety of techniques and applications continue to grow rapidly. Cameras are an accessible tool which can provide rich, high-frequency information about the environment. Images can be analyzed on either a local or a global scale, giving information about the colour and texture of the scene, as well as 3D positions of features in the scene. In addition, working in the space of appearance generally provides a complimentary type of information and uncertainty, compared to that of other sensors. Using cameras alongside other sensors can therefore be especially useful in mapping and localization tasks.

The previous subsection explained the importance of data associations such as loop-closure or multi-map fusion for autonomous mapping and localization. Appearance-based techniques have been brought to the forefront recently, as popular ways of robustly detecting data associations for tasks such as loop-closures, map fusion, or relocalization. The works of Cummins [2009] and Milford [2013] are common examples of purely vision-based approaches, while the works of Angeli et al. [2008b] and Maddern et al. [2012] use observations from both vision

and odometry or inertial sensors. In addition to this, appearance-based techniques can also be used for segmenting topological regions and performing localization and mapping using this topological space, such as the works of Angeli et al. [2008a, 2009] and Liu and Siegwart [2014]. More recently, large-scale SLAM using only a monocular camera is becoming feasible due to advanced applications of structure from motion, visual odometry, and visual place recognition [Weiss et al., 2013, Engel et al., 2014]. As visual place recognition is the topic of this thesis, work related to this topic will be developed more thoroughly in the remaining sections of this chapter.

Localization and mapping in the space of appearance is difficult for a number of reasons, including: repetitive features which cause perceptual aliasing, dynamic elements in the environment (moving objects, lighting changes, seasonal changes, etc.), occlusions, and appearance changes due to viewpoint. Furthermore, factors such as low image quality and blur can also often obfuscate the appearance of a scene.

In addition, the impact of false data associations can have very serious consequences, such as unknowingly corrupting maps with errors in an irrecoverable manner. Therefore, the importance of the system’s ability to make correct decisions should be stressed, only making high-confidence data associations and avoiding incorrect associations at all costs.

## 2.3 Visual Place Recognition with Global Image Attributes

Approaches to visual place recognition vary between using information from the entire image, or focusing on some collection of local features within the images. The dimensionality of the image space and the amount of information that can be contained in an image make it difficult to extract and utilize this information in an efficient way. Examples of working with the entire image include directly calculating differences between the pixel intensities of each image, or choosing alternate ways of representing the images, such as comparing histograms of pixel values, or other vectors designed to capture the nature of the image. Each of the methods generally have trade-offs between robustness and discriminative power, and complexity. Typically though, working with global image attributes tends to function better under poor image quality (e.g., low resolution, low image depth, and/or blur) in comparison to feature-based techniques. On the other hand, the balance between their discriminative power and invariance is more difficult, and retrieving images from large collections is, generally speaking, also more difficult. A few examples of image comparison techniques and localization frameworks which use such global image attributes will be outlined now.

### 2.3.1 Direct Image Comparisons

There are numerous ways of computing similarity measures between images, under many different contexts. Because directly comparing pixel intensities between images of scenes under varying viewpoints produces very low similarity scores, such techniques have generally not been applied to tasks such as place recognition in the past, but rather used in tasks like basic object detection through template matching. However, these techniques have recently become more popular in the place recognition community, due to the introduction of new algorithms which are very simple to implement and have been shown to be robust against extreme lighting and weather changes [Milford, 2013]. The details behind this and other algorithms will be discussed in Section 2.3.3, while this section will mention a few generic image comparison methods.

One of the simplest ways to compute the difference between two images (or two meta-images of image sequences for example) is by summing the difference in pixel intensities, either by the sum of absolute differences (SAD):

$$SAD = \sum_{i,j} |X_{ij} - Y_{ij}| \quad (2.1)$$

or the sum of squared differences (SSD):

$$SSD = \sum_{i,j} (X_{ij} - Y_{ij})^2 \quad (2.2)$$

where  $X$  and  $Y$  are the two images, and  $i$  and  $j$  are the rows and columns respectively. Another method is to calculate the cross-correlation between the images (CC):

$$CC = \sum_{i,j} (X_{ij} \cdot Y_{ij}) \quad (2.3)$$

There are also normalized versions of each method, which tend to produce better results, especially in the case of the normalized cross-correlation (NCC):

$$NCC = \frac{\sum_{i,j} X_{ij} \cdot Y_{ij}}{\sqrt{\sum_{i,j} (X_{ij})^2 \cdot \sum_{i,j} (Y_{ij})^2}} \quad (2.4)$$

All these methods tend to be sensitive to slight variations in the images (such as view-point changes or lighting changes), therefore images are usually smoothed before comparison, for instance by applying Gaussian blur.

### 2.3.2 Global Image Descriptors

Due to the sensitivity of the similarity measures introduced in the previous subsection, much work has been done on finding ways of capturing the discriminative information from images in a robust way. These global image descriptors also generally result in a decreased dimensionality of the problem, and therefore reduce the complexity and memory needed for analysis.

Image histograms have been widely used in the fields of image retrieval and place recognition. Histograms can be made for things such as pixel intensities, colours, edge intensity, etc. These essentially represent the likelihood distribution of the values in the image (or location), and can additionally be used to index similar images from a database in a fairly efficient way [Swain and Ballard, 1991]. Histograms provide a compact description of an image, and maintain a good degree of invariance to rotations, occlusions, and small translations which may occur (for example rotations of a robot equipped with a panoramic camera remain invariant).

Texture-based features also exist, and can provide a substantial amount of information about an image or scene. A common method is to convolve the image with a bank of filters which capture the spatial frequency and directionality of the texture of the image, and combine the results in a vectorial descriptor. Examples include Gabor filters and wavelet decomposition [Torralba et al., 2003].

For each of the global descriptors mentioned, images can always be subdivided into a few regions, in order to maintain some information about the spatial layout of the image. Of course, when this is taken too far, the same problems of invariance to rotations and translations mentioned in the previous subsection, begin to manifest themselves again.

Taking the ideas a bit farther, the work of Oliva and Torralba [2006] tries to find a global image descriptor which is able to capture the gist of the scene in the image. Using the same kinds of texture features described above, applied to many images of varying types of scenes, the principal components can then be extracted and used as weights for computing a set of global features. The global features are calculated by applying the same set of filters on the given image and then projecting the results into the pre-computed principal components. These global descriptors can then provide a vector which should capture the discriminative features of scenes.

### 2.3.3 Place Recognition with Global Image Attributes

With image descriptors in place, the question remains of how to exploit the information that they contain in order to perform visual place recognition effectively.

The work of Ulrich and Nourbakhsh [2000] performs topological localization using colour histograms of panoramic images. They worked with learned models of each location in the map (solving the localization problem, rather than the SLAM problem). In addition, because the map is known in advance, computation is reduced by comparing only a query image with its immediate neighbours in the topology. Six one dimensional histograms are constructed for each image (one for each channel of the hue-luminance-saturation, HLS, colour spaces, and one for each channel of the red-green-blue, RGB, colour spaces). The distance between query and candidate location histograms are then computed, and the ratio of the minimum distance to the second smallest distance is used to gauge the confidence of the match.

The work of Torralba et al. [2003] uses the texture-based features described in the previous subsection to do a similar task. Location models are also learned ahead of time using different views of various types of scenes, and the system uses a Hidden Markov Model (HMM) to estimate the probability of being in a certain type of location given the observation and a transition probability from the previous type of location. The authors demonstrate the importance of the HMM in boosting results, but had to rescale observation likelihoods in order to balance the weighting of the prior and the likelihoods.

Work by Murillo and Kosecka [2009] uses the more sophisticated Gist descriptors to perform visual place recognition for the task of loop-closure detection in SLAM. The authors adapt the Gist descriptor for use on panoramic imagery. The described system is able to return a set of candidate locations which may match a query, but the authors propose verifying matches in more detail to accept candidates, as many incorrect locations are often returned. Similarly, the work of Sünderhauf and Protzel [2011] uses Gist-like descriptors to represent each scene. However, rather than using Gist, they apply the BRIEF descriptor [Calonder et al., 2010] to the entire image (rather than the local interest points it was originally designed for). The BRIEF descriptor is fast to compute, and especially fast to compare the distances between descriptors. Despite its simplicity, this method has reported useful results on several datasets.

Taking the simplicity of this approach a bit further, the work of [Milford and Wyeth, 2012, Milford, 2013] (SeqSLAM) compares sequences of down-sampled images to detect loop-closures. In order to increase the discriminative nature of the observation and avoid the

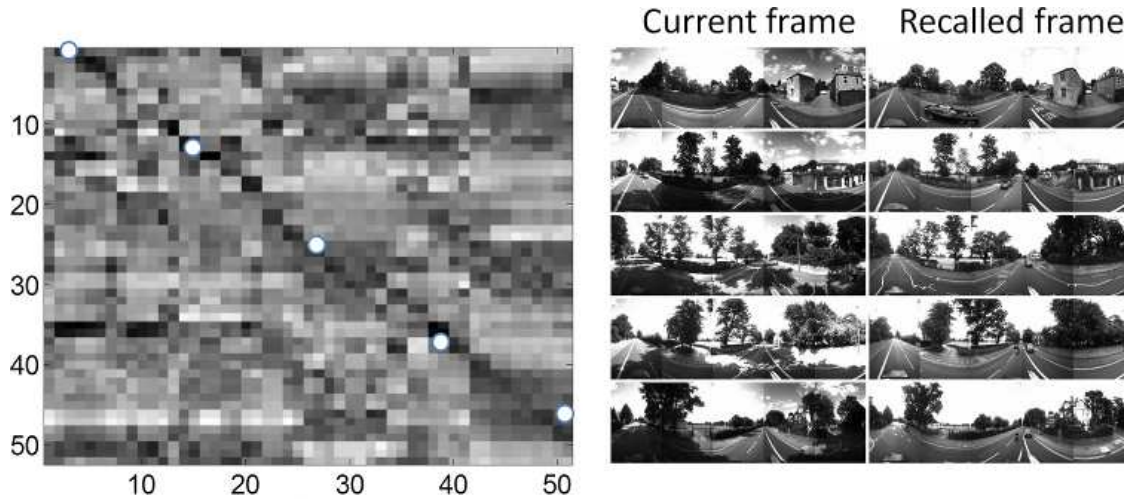


Figure 2.5: An example of the difference matrix (left) described in [Milford, 2013], comparing two streams of images. The strong dark diagonal line in the difference matrix indicates a match between the corresponding images sequences. The query and recalled images are shown on the right, with white dots in the difference matrix indicating where in the sequence the image pairs come from.

problem of false-positives, this work represents each location as a long sequence of images, rather than just using the images from the one pose. The authors create a difference matrix comparing all previously seen locations to each other using SAD scores (Equation 2.1), perform local normalization, and then search for diagonals of low difference values over the defined sequence length. An example can be seen in Figure 2.5, showing the resulting difference matrix between two streams of images, along with the corresponding query and matched location images. Due to the use of sequences rather than single images, and the local normalization of the difference matrix, this approach is able to provide good results, even in the case of low quality imagery (low resolution, low depth, and image blur). However, due to the use of SAD, this approach suffers severely in the case of variances in trajectory (non-constant speeds and changes in view-point). Some add-ons have been developed, to cope with multiple traversal directions of a path by checking panoramic images adjusted by an off-set [Milford, 2013], and to cope with speed variations using local odometry measurements [Pepperell et al., 2013]. However, the functionality of the approach is still restricted to applications where the same path is traversed each time due to the lack of view-point invariance.

## 2.4 Visual Place Recognition with Bags of Features

Rather than working with features which describe an entire image, images are often represented by a set of local features which they contain. The idea behind this representation is to



pick out the important features that embody the meaning in the image, and describe these local features in a way which is robust to things such as lighting and view-point changes. As with global image descriptors, a number of different ways of defining and describing these interest points exist. Furthermore, by working with quantized feature descriptors, image retrieval and analysis inherits many analogies from the field of text document analysis, allowing the application of many well-studied tools.

### 2.4.1 Interest Point Detection and Description

In order to represent an image by a set of local features, so-called interest points need to be detected and described. The goal of an interest point detector is to find points or regions in the image which are representative of the scene, and can be robustly identified in new images of the same things, even in the case of changing lighting conditions or viewing angle. In general, detectors look for either corners or uniform blobs in the images. Many options exist, with a few common examples described here, and some example results shown in Figure 2.6.

Once detected, the interest points also need a descriptor to capture the information about the corresponding region, in order to find correspondences between interest points in other images and in order to analyze a scene. Both the detector and descriptor should ideally also be invariant to any expected deviations such as varying lighting conditions and geometric transformations from varying viewpoints. A number of descriptors have been developed, and the choice depends on the types of interest points we want to detect, the type of environments we want to operate in, the desired task, and the computational restrictions involved.

Harris corners search for places in the image where the gradient changes significantly in more than one direction (corners), which then serve as the interest points in the image [Harris and Stephens, 1988]. Harris corner detection is not invariant to scale changes, but has been adapted in the methods of Harris-Laplace and Hessian-Laplace, and can be further extended to detect areas which are invariant to affine transformations, found through iterative algorithms [Mikolajczyk and Schmid, 2004]. Maximally Stable Extremal Regions (MSER) [Matas et al., 2004], is used to detect regions of stable pixel intensities. These regions are also invariant to affine transformations, and have been shown to additionally cope well with illumination changes [Mikolajczyk and Schmid, 2004]. Due to the types of regions it detects, MSER tends to perform well in easily segmentable scenes, rather than highly textured scenes, and is quite sensitive to noise and occlusions.

The Scale-Invariant Feature Transform (SIFT) [Lowe, 1999] algorithm is another solution to the problems related to scale when using Harris corners. As the name implies, it extracts



Figure 2.6: Examples of various interest point detectors: original image (top-left), Harris corners (top-right), SIFT (middle-left), MSER (middle-right), SURF (bottom-left), and BRISK (bottom-right).

scale-invariant interest points and computes their descriptors by finding extrema across both scale and space in the outputs of varying scales of difference of Gaussian filters. SIFT features also define an associated orientation, according to the gradient of the surrounding neighbourhood. The SIFT descriptor is given by combining histograms of intensity and orientation values in the region surrounding the interest point into a 128-dimensional vector.

In order to speed up computation time, Speeded-Up Robust Features (SURF) [Bay et al., 2008] uses simplified box filters instead of the difference of Gaussian filters. Additionally,

upright SURF (U-SURF) skips the process of finding the orientation of the feature, in order to speed things up even more, in the case where feature orientations are not expected to change. SURF features are described using Haar wavelet responses to capture gradient information in two directions. SURF descriptors are then usually given by a 64-dimensional vector, but can be extended to a more descriptive 128-dimensional version.

Binary Robust Independent Elementary Features (BRIEF) [Calonder et al., 2010] is an alternative descriptor which reduces memory requirements and significantly reduces the computation required to calculate and compare descriptors. BRIEF descriptors are binary vectors (the chosen dimension can vary depending on the requirements) of the results of simple intensity comparisons between pixels in the local neighbourhood of the interest point. In order to cope better with scale changes and rotations, but maintain the increase in efficiency provided by BRIEF, Binary Robust Invariant Scalable Keypoints (BRISK) [Leutenegger et al., 2011] was developed. It also creates a binary vector, this time by a more formalized set of pixel comparisons sampled from a circular ring pattern around the interest point.

The concepts developed in this work are independent of interest point detector and descriptor choice. For this work we use U-SURF for its performance characteristics, ease of integration based on prior work, and ease of comparison with many state-of-the-art approaches which use SURF. However, although SURF is good at dealing with things like image blurring, it is documented as not coping well with lighting and viewpoint changes [Bay et al., 2008]. Lighting and viewpoint invariance is critical to the task of place recognition, and therefore there may be a good deal of room for performance improvement by working with alternative features, but this is not tested in this work.

### 2.4.2 Visual Words

Working with sets of feature descriptors to represent each image has allowed for significant improvements in tasks such as object detection, as one can search for similar descriptors in other images, finding objects even in the case of cluttered and occluded environments. However, as the number of images involved in the search grows, the task quickly becomes difficult. Images can contain hundreds or thousands of interest points, and searching for similar descriptors in (for example) 128-dimensional space is complex. In order to drastically increase the efficiency of this, the work of Sivic and Zisserman [2003] introduces quantized versions of descriptors, now giving the systems the ability to create inverted indexes for efficient image retrieval. Quantization of descriptors is typically done by clustering large amounts of descriptors contained in sample images, capturing the appearance of common visual elements.

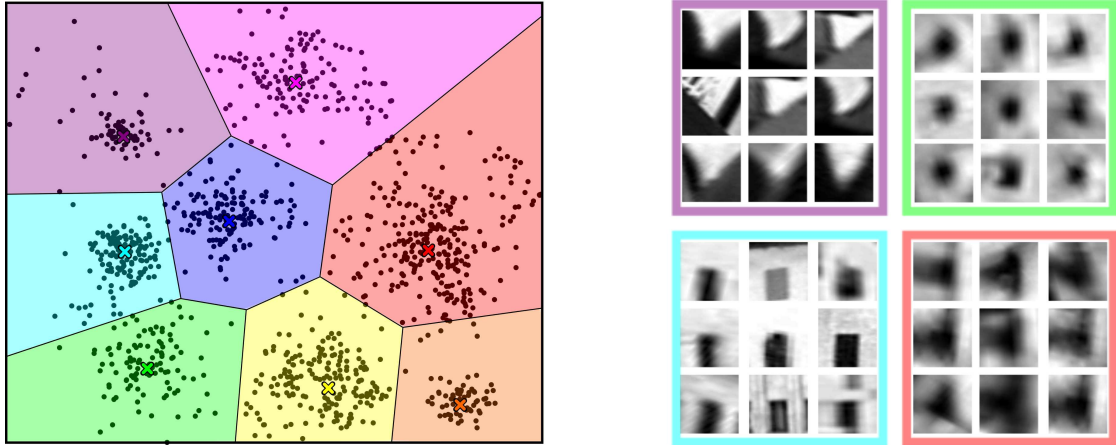


Figure 2.7: Illustration of descriptor clustering in an oversimplified 2-dimensional space and examples of features belonging to the same clusters, and therefore being represented by the same visual words. (Example word image patches taken from [Sivic and Zisserman, 2003])

These quantized descriptors are referred to as visual words, and this representation establishes analogies between text analysis and visual analysis. As a result, many theoretical developments from the field of text retrieval become applicable to images.

Descriptor quantization is typically achieved by a clustering algorithm such as K-means [Sivic and Zisserman, 2003, Bishop, 2007]. Each descriptor is then assigned to the visual word represented by the closest cluster centre in descriptor space. A simplified 2-dimensional illustration of the kind of results achieved by k-means clustering to partition the space is shown in Figure 2.7. Figure 2.7 also shows examples of groups of image features which have been assigned the same visual words. The entire set of resulting visual words are referred to as the visual vocabulary, contained in a visual dictionary. In the case of large descriptor spaces or large vocabularies, methods like approximate k-means [Philbin et al., 2007] or hierarchical k-means [Nister and Stewenius, 2006] are often used instead.

Inspection of the clustering results shown in the left side of Figure 2.7 shows that there are some problems with the k-means clustering approach. For example, many descriptors near the boundaries of two Voronoi cells are often assigned to the wrong visual word. The lack of consideration about the underlying word distributions and relative variances results in sub-optimal word assignments. Construction of visual dictionaries is not limited to such clustering methods, but visual word assignment in such large spaces is a complex task, especially considering the difficulty in accumulating enough descriptors to fill the space and capture the true distribution of visual elements. A number of concepts have been developed to deal with problems of traditional vocabulary clustering methods, including the already mentioned hierarchical vocabularies [Nister and Stewenius, 2006], soft word assignments [Philbin et al.,

2008], hamming embedding [Jégou et al., 2010], and weak scale and orientation consistency checks [Jégou et al., 2010]. In general, empirical studies have suggested that larger visual vocabularies tend to outperform smaller ones [Cummins, 2009]. However, even in the case of low quality clustering results, useful experimental results have been obtained. For example visual dictionaries generated by k-means and trained using outdoor imagery can be successfully applied for indoor localization.

### 2.4.3 Information Retrieval

When working with visual words, relevant images can be retrieved quickly using an inverted index system, where a list is maintained for each word in the dictionary, containing the reference number of each image in the database where that visual word has been seen [Sivic and Zisserman, 2003]. Using this, images related to a given query are simply given by those other images from the lists associated to the words from the query. Compared to previous methods where images had to be retrieved by more complex descriptor matching techniques, this greatly simplifies the procedure.

Once a list of other images containing query words are retrieved, they often undergo further analysis to perform tasks such as image classification, object detection and recognition, or place recognition. In order to compare images further, a variety of approaches borrowed from the field of text retrieval can be used. One common scoring method that has been used is known as Term Frequency - Inverse Document Frequency ( $tf \times idf$ ) [Manning et al., 2008]. This method creates vectors for each document, where each element of the vector is the ratio between how common a word is within that document and how common the word is within the entire set of documents. Therefore, the  $tf \times idf$  vector is the length of the visual dictionary, where each term is given by:

$$t_i = \frac{z_{id}}{z_d} \times \log \frac{N}{z_i} \quad (2.5)$$

and where  $z_{id}$  is the number of times that word  $i$  has appeared in document  $d$ ,  $z_d$  is the total number of words in document  $d$ ,  $z_i$  is the number of times that word  $i$  has appeared in the entire database of documents, and  $N$  is the number of documents in the database. These terms therefore give weight to words occurring in the particular document, but down-weights terms that are also commonly occurring throughout many documents. Document similarity can then be measured by finding the distance between their corresponding  $tf \times idf$  vectors (usually given by the dot-product).

Another common approach, involves modelling the probabilities of word occurrences using the observed frequencies. This can be used to compare documents or assess what class or category a document belongs to. For example, according to Bayes' rule, the probability of a document belonging to a certain class can be written in terms of the likelihood of those words belonging to a certain class, and the prior probability of each class:

$$P(C|z_1, z_2, \dots, z_N) = \frac{P(z_1, z_2, \dots, z_N|C)P(C)}{P(z_1, z_2, \dots, z_N)} \quad (2.6a)$$

$$= \frac{P(z_1, z_2, \dots, z_N|C)P(C)}{\sum_{\{C\}} P(z_1, z_2, \dots, z_N|C)P(C)} \quad (2.6b)$$

where the value of  $C$  represents the given class, and  $z_1, z_2, \dots, z_N$  represent the set of words. Furthermore, when conditioned by the document class, the probabilities of the occurrence of each word are often assumed to be independent of each other, reducing the likelihood to a product of individual word likelihoods:

$$P(z_1, z_2, \dots, z_N|C) = \prod_{i=1}^N P(z_i|C) \quad (2.7)$$

This independence assumption is referred to as the Naive-Bayes assumption, and is common in practical applications, in order to greatly simplify the inference problem [Manning et al., 2008]. When it comes to modelling the distribution of word occurrences, either a Multinomial or a Bernoulli model is typically used. In the case of a Multinomial model,  $z_1, z_2, \dots, z_N$  represents the set of words which exist in a document, and is therefore a different size for each document. In this case, each word likelihood is estimated by the number of times the word appears in all documents of the conditioned class, divided by the total number of words in the documents of the conditioned class. Alternatively, when using the Bernoulli model, word occurrences are binary, and  $z_1, z_2, \dots, z_N$  represents all words in the dictionary, taking values of 0 or 1, depending on their presence in a document. Thus, the word likelihoods are given by the number of documents in the class which contain the word, divided by the number of documents in the class. Under both models, the class priors are given by the relative fraction of documents which belong to each class. In addition, as Equation 2.7 is arbitrarily zero as soon as one likelihood value is estimated to be zero (which is common in frequency based estimations using small sample sets), word likelihoods need to be smoothed to achieve practical results. A number of methods exist, with the most common being plus-one smoothing, but readers may refer to [Manning et al., 2008] for more details.

Furthermore, the most complex part of Equation 2.6 is in the calculation of the denominator which correctly normalizes the posterior probability. As a result, this normalization step is often skipped in practical application, especially when only the maximizing class is desired. This is referred to as maximum a posteriori estimation:

$$C_{MAP}(z_1, z_2, \dots, z_N) = \arg \max_C P(z_1, z_2, \dots, z_N|C)P(C) \quad (2.8)$$

Alternatively, many applications even focus on maximizing the likelihood only, referred to as a maximum likelihood estimation.

These types techniques such as *tf* × *idf* and Naive-Bayes are commonly referred to as bag-of-words (BOW) techniques due to the fact that the structural relationships between words are removed, meaning that an image or document is simply represented by its set of words, with no particular structure or order.

#### 2.4.4 Place Recognition with Visual Words

After the introduction of visual words for efficient retrieval and comparison of images in the work of Sivic and Zisserman [2003], bag-of-words techniques were subsequently applied to the task of visual place recognition. For example, *tf* × *idf* scoring has been successfully used in work such as [Angeli et al., 2008b], [Mei et al., 2010], and [Botterill et al., 2011].

Calculating *tf* × *idf* scores is relatively easy to implement and computationally efficient. However, it requires threshold tuning when detecting matches, and (as in text retrieval), has been shown to be sensitive to location (document) characteristics such as repetitive structures [Schneider, 2004, Jegou et al., 2009] and size [Mei et al., 2010]. Alternative weighting schemes also exist, which try to compensate for bursts in word frequencies which overwhelm the relative word frequencies [Jegou et al., 2009, Torii et al., 2013].

In the work of Angeli et al. [2008b], loop-closures are detected using a Bayesian framework, where observation likelihoods are estimated using *tf* × *idf* comparisons, and location priors are evaluated based on a transition model between states. Loop-closures are provided by the previously seen image which maximizes the posterior probability of the loop-closure hypothesis given an image of the current position. Images directly before the current image are not included in the evaluation, as the system would otherwise simply close the loop with the position immediately before the current one. In addition, in this system, perceptual aliasing is primarily resolved using epipolar geometry checks on any retrieved loop-closures. By incorporating the *tf* × *idf* scores into a probabilistic framework, problems related to score

thresholding is made easier, as data association decisions can be made based on probabilities rather than arbitrary scores.

Other scoring methods rely on more detailed probabilistic models, such as the work of Cummins and Newman [2008, 2011], and Cadena et al. [2012]. In the FAB-MAP (Fast Appearance-Based MAPping) [Cummins and Newman, 2008] framework, locations are composed of sets of visual words, where the existence of words is modelled as a hidden variable, with the observed words representing noisy measurements of the existing word set. Loop-closures are then provided by the location which maximizes the posterior probability of the current query observation. Observation likelihoods are provided by a model based on the detection probabilities of observations given existence, under a Naive-Bayes scheme. In order to then compute posterior probabilities, normalization is done by marginalizing across all locations in the map, plus an extra location which represents the unseen world. This generative framework allows for a natural and principled way to incorporate dynamic environments and perceptual aliasing, and only requires two parameters representing feature detection probabilities. The inclusion of the unseen world is accomplished by the use of sample images, and is most important for combating perceptual aliasing. Furthermore, decision thresholds for matching locations are clear probabilities. Like in [Angeli et al., 2008b], locations immediately prior to the current location are excluded from the comparisons. The work of [Cummins and Newman, 2008] also investigated the relaxation of the Naive-Bayes assumption, removing some of the independence assumptions based on results of a Chow-Liu tree constructed from the visual vocabulary. Alternatively, discriminative models used in [Cadena et al., 2012] have also been shown to produce good results without explicit knowledge of hidden variables, but using training data to learn the model directly. This approach also makes use of 3D landmark positions, unlike FAB-MAP.

## 2.5 Defining the Notion of Places

The scope of a location generally varies between either using discrete poses or loosely defined sets of poses. Location models built using specific poses in the robot’s trajectory limit the view of the world, and often imply that the robot must visit the same arbitrary pose in order to recognize any relevant loop-closures. In addition, single images lack the necessary context due to their limited view, often resulting in perceptual aliasing. Abstraction from single image location models is often achieved by using sequences of images in time, or sets of images based on a topological world, therefore increasing the available information about a



given scene. Both methods deal with the problem of defining where one location ends and the next location begins in a consistent manner. Furthermore, dealing with trajectory invariance can be problematic when place representations depend so strongly on the particular traversal of the environment. As an alternative, the approach of [Mei et al., 2010] simultaneously limits the dependence on the observation trajectory and solves for the scope of a location, by defining scenes as clusters of features in a covisibility graph. The next three subsections describe further details about these different methods of defining and representing places, with examples of where they have been applied in place recognition frameworks.

### 2.5.1 Working with Single Images

As visual place recognition is usually used within a SLAM framework for loop-closure, there are many examples of methods which define locations to be given by each keyframe pose in the SLAM graph (see Figure 2.2). The methodology behind selecting keyframes varies from simply selecting poses based on time or distance intervals [Konolige and Agrawal, 2008], to using image similarity metrics or feature covisibility thresholds to pick out representative poses [Zhang et al., 2010, Strasdat et al., 2011]. Examples where these kinds of single-image location representations have been used for performing visual place recognition include the works of Cummins and Newman [2008] and Angeli et al. [2008b]. Although these works produce successful results, such narrow views of each location lack the context to differentiate between self-similar structures in the environment, resulting in perceptual aliasing. As a result, each method relies on extensive probabilistic methods and additional geometric checks to achieve the required precision. In order to better cope with this problem, other systems consider a larger scope when defining places in the world, as will be discussed in the next subsection.

### 2.5.2 Working with Sets of Images

In order to extend the representation of a place to a more continuous notion, many frameworks make use of sequences of images. Most vision-based systems rely on sampling images at a given time interval [Gálvez-López and Tardós, 2012, Milford, 2013], but when available, some incorporate local metric position estimates to sample images based on distance intervals instead [Maddern et al., 2012, Pepperell et al., 2013]. More recently, the work of MacTavish and Barfoot [2014] gives hints at how hierarchical sizes of image sets can be used to achieve more efficient place recognition, but still relies on fixed scales of sequences, and does not compare sets across different scales. Such methods using image sequences still rely

on strong assumptions about the observation trajectory. Generally working with continuous, fixed sizes of image sets, these methods have difficulty coping with inconsistent trajectories and defining the extent of a location.

Rather than working with a complete representation of the world, some mapping and place recognition systems focus on defining topological places. The difficulty in this approach is how to automatically segment places into meaningful and consistent representations, referred to in the field as landmark detection (note that here landmark is not used in the same sense as this work). The work of Ranganathan and Dellaert [2009, 2011] deals with topological mapping and place segmentation using Bayesian surprise. However, much of the existing work on topological mapping and place recognition rely on prior definition and training images of each place, such as the work of Pronobis et al. [2006]. Alternatively, work by Chli and Davison [2009] is able to automatically infer these kinds of substructures within maps using monocular images and mutual information measures between visual features. Such topological approaches are often restricted to more structured environments, usually indoors, where each region has more clear boundaries.

### 2.5.3 Covisibility Graphs

In order to diminish the reliance on the robot trajectory and remain applicable in all environments, the work of Mei et al. [2010] represents the world as a graph of its visual features (landmarks), where edges between landmarks represent whether landmarks have been observed together (covisibility). As a result, connectivity between landmarks summarizes the implicit structure of the features in the world. The covisibility representation additionally mirrors that needed for bundle adjustment, and can thus be applied directly in SLAM frameworks [Mei et al., 2010, Strasdat et al., 2011]. Furthermore, covisibility structure has shown to be linked to the normalized information matrix (inverse of the covariance matrix) in SLAM frameworks [Neira et al., 2003, Williams et al., 2009], indicating that features that are co-observed also have high values of co-information.

Using this graph, places can be retrieved as connected clusters of landmarks from the map. These location models are then based on the underlying environmental features, rather than the discretization of the robot's trajectory in the form of individual images, or sequences of images. Working with the covisibility structure also allows for the application of techniques from the field of graph theory to aid in place recognition. Since this covisibility map methodology is strongly related to the developments of this thesis, it is developed more thoroughly within Chapter 3.

## 2.6 Visual Place Recognition with Geometric Features

Reflecting on the bag-of-features approach to place recognition outlined in Section 2.4, one drawback of these approaches is that they discard most of the geometric information when comparing feature sets, therefore reducing the discriminative nature of the model and typically resulting in either perceptual aliasing or reduced recall. Following this realization, previous works have investigated ways of incorporating some geometric information into the location models. The following subsections discuss some of these approaches.

### 2.6.1 Weak Geometric Constraints

Rather than relying on explicit 3D information, many existing methods look into more abstract geometrical features, which we will refer to here as weak geometric constraints. Advantages of such methods include that full 3D geometry does not need to be extracted or stored, and therefore the computational requirements are typically lower. In addition, these methods can often cope with certain types of noise and errors more easily, due to their relative simplicity. At the same time, they remain less discriminative than more complete geometric observations would be.

As an example, the work of Jégou et al. [2010] incorporates quantized angle and scale information about visual features, along with the quantized descriptors. This is then used to reduce scores of images where scale and orientation of features is not consistent with a valid image transformation. Another example is the work of Johns and Yang [2013, 2014], which quantizes features 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 and distances, as well as possibly their relative orientations and scales. One possible drawback of this approach is the difficulty in quantizing spatial features in a robust way, as well as possible memory and computation constraints.

A slightly different approach is given by Cao and Snavely [2013], where a database of images is represented by a large graph, where images containing overlapping places are connected by edges. Using this, the position of a query image in this graph can be found using information about neighbouring images, providing larger scale geometric constraints. However, this method requires a pre-existing graph of images, as well as training in order to calibrate each neighbourhood model. Also working with graphical structures, the work of Fisher et al. [2011] defines semantic and spatial relationships between objects in virtual scenes, creating

scene graphs. Scene similarity can then be evaluated using fixed-length walk graph kernels. In a somewhat similar fashion, work by Yoon et al. [2014] defines an appearance graph for panoramic scenes based on omnidirectional images. This is done by detecting features in the images and describing the pairwise relationship between features as the angle between them (based on a coordinate frame centred on the image), which can then be used to compare locations using spectral graph matching. However, this method has only been demonstrated for very small maps of 14 locations.

### 2.6.2 3D Geometric Constraints

In order to obtain more discriminative representations of scenes, or in order to achieve accurate 3D localization, a number of other systems look at 3D geometric constraints during place recognition. For example, in [Paul and Newman, 2010] (FAB-MAP 3D), locations are represented by both visual landmarks and a distribution of the 3D distances between, given by range-finders or stereo cameras. The work of Cadena et al. [2012] also considers both appearance and geometric information of features. First, an appearance-only bag-of-words approach searches for candidate loop-closures, and then unclear candidates are verified with a comparison of minimum spanning trees built over the 3D coordinates of features (given by stereo imagery) under a trained discriminative model. These approaches are able to achieve better recall than appearance-only systems like FAB-MAP due to their ability to avoid perceptual aliasing, however they require acquisition, storage, and analysis of 3D position data.

The work of Agarwal et al. [2011] aims at reconstructing the 3D geometry from thousands of images collected from online public repositories. Using structure from motion and multiview stereo reconstruction, they can reconstruct 3D models of cities in less than a day of computation using parallel computing. The works of Sattler et al. [2012] and Choudhary and Narayanan [2012] also use structure from motion on sets of images to perform 3D localization. Sattler et al. [2012] uses an active search method to reduce and refine the correspondence task, while Choudhary and Narayanan [2012] defines a probabilistic framework to increase the triangulated point density while remaining efficient. Overall, such methods have become drastically more efficient and reliable over last few years, and are an active area of research. The drawback of these approaches is the high storage requirements for maintaining more information about features, and collecting many images from varying viewpoints.

## 2.7 Outlook

Visual place recognition algorithms have come a long way in recent years, enabling significant improvements in long-term localization and mapping. Various techniques have been successfully employed in datasets up to distances of 1000 km, containing drastic lighting changes and many self-similar locations which cause perceptual aliasing [Cummins and Newman, 2011, Milford, 2013]. Impressive as these systems are, there is still room for improvement in terms of how maps and locations are modelled. In this thesis, emphasis is placed on exploring several existing and novel location representations in order to better exploit the available visual information for the task of place recognition.

Learning from the work summarized throughout this chapter, the value of generative models with locally invariant feature descriptors is recognized, alongside the inclusion of sample locations to represent the unseen places in the environment. These models incorporate uncertainty about observations of the world, allow for varying observation conditions due to the use of local features, and also provide more intuitive thresholds for making data association decisions [Cummins and Newman, 2011]. Furthermore, quantizing these local features to visual words, facilitates efficient search and retrieval of scenes from vast amounts of images [Sivic and Zisserman, 2003].

At the same time, the importance behind the notion of defining the extent of a place should also be recognized. Methods which extend locations beyond single viewpoints are able to improve performance drastically, due to the additional context which helps to differentiate between similar places [Milford, 2013]. However, existing methods rely on images sequences of predefined lengths, and have been shown to be very sensitive to how the scene is traversed and especially the chosen sequence length.

As a result, this thesis aims to identify how these location models can be improved, notably by incorporating information about the underlying structure of features in the scene. To do so, covisibility graphs are used, providing implicit information about the underlying physical structure of the environment and how it is observed; representing locations as clusters of landmarks from the global map. A number of generative models for location comparison are studied, and new ones are proposed. The methods developed in this thesis remain general to the type of feature detectors and descriptors which are used, meaning that although the choice of features is extremely important, this choice is not the focus of the thesis, as one can simply be substituted by another. In order to evaluate the applicability of various models, tests are performed on several real-world datasets, mandating robustness to dynamic environments,

repetitive structures in the environment (perceptual aliasing), and trajectory and viewpoint variations.



## Chapter 3

# Covisibility Mapping

### 3.1 Introduction

Given a query location (usually corresponding to the current position of the robot), the idea is for the system to be able to evaluate if and where the same location was seen before. The primary approaches developed in this thesis rely on location descriptions comprised of sets of visual words [Sivic and Zisserman, 2003], enabling efficient comparison of the query with a set of candidate locations retrieved from the current map. This chapter outlines how the environment is represented as a graph of such visual features, where covisibility defines connectivity [Mei et al., 2010], upon which probabilistic location models are later built (the details behind these probabilistic models will be described in Chapters 4 and 5). Quantized visual words are used to represent feature descriptors provided by each landmark (distinct visual features in the image). The map is then constructed as an undirected covisibility graph, with these landmarks as nodes, and edges representing the information that the connected landmarks were seen together. At query time, the graph can be searched for clusters of landmarks which share strong similarity with the query, extracting subgraphs which represent candidate virtual locations for further analysis. These virtual locations dynamically adapt to the query scene, and are inherently less reliant on the robot’s observation trajectory. Section 3.2 explains how the covisibility graph is built and maintained over time, while Section 3.3 discusses how to retrieve the relevant virtual locations.

### 3.2 Building the Covisibility Graph

The covisibility map is built up incrementally, updating nodes and edges in the graph as each new image is processed. This is done by detecting and extracting relevant features,



tracking them between successive images, and recording which landmarks were seen together and where. In addition to tracking, data association from place recognition can be used to update landmark information as well. The graph construction procedure will now be described in more detail in the following subsections.

### 3.2.1 Feature Extraction

As each image is processed, a set of visual features,  $\ell_i$ , are detected and represented by a vector-based descriptor such as SIFT or SURF, as described in Section 2.4.1. Each landmark is furthermore associated with a quantized visual word, which is taken by the closest match in a pre-trained visual dictionary, as described in Section 2.4.2. Thus, each image provides a set of words, which represent an observation  $\mathcal{Z}_k$ , which is able to maintain some invariance to view-point and lighting changes, due to the use of local feature descriptors. In this work, 128-dimensional U-SURF features are utilized, due to their performance characteristics, and ease of integration and comparison with existing approaches. Similarly, visual words are assigned using a pre-defined visual dictionary consisting of 10987 words, given by Cummins and Newman [2008]. However, the methods described throughout this work remain general to any form of quantized features, leaving the choice of feature detection and description open. In fact, results could likely be significantly improved by more careful choices about descriptors and especially dictionary encoding.

### 3.2.2 Graph Creation

Between consecutive images, local data association is implemented through feature tracking, meaning that the identity of landmarks are identified across multiple images. Tracking is performed between features in subsequent image frames by comparing descriptors, and optionally refined using epipolar geometry and RANSAC (RANdom SAMple Consensus). Tracked features are then represented as the same landmark,  $\ell_i$ . A simple example of some observations, the resulting covisibility map, and a given query observation can be seen in Figure 3.1.

The current map,  $\mathcal{M}_k$ , is updated as information from each new image is processed. The map is implemented as a sparse clique matrix,  $C_k$ , with each column representing an observation  $\mathcal{Z}_k$ , and each row representing a particular landmark,  $\ell_i$ . Therefore the value in row  $r$ , and column  $c$  indicates whether or not landmark  $\ell_r$  was seen in observation  $\mathcal{Z}_c$ . An adjacency matrix,  $A_k$ , for the covisibility graph can simply be found by taking  $A_k = H(C_k \dot{C}_k^T)$  (with  $H(\cdot)$  being the element-wise unit step function), but is often not explicitly needed.

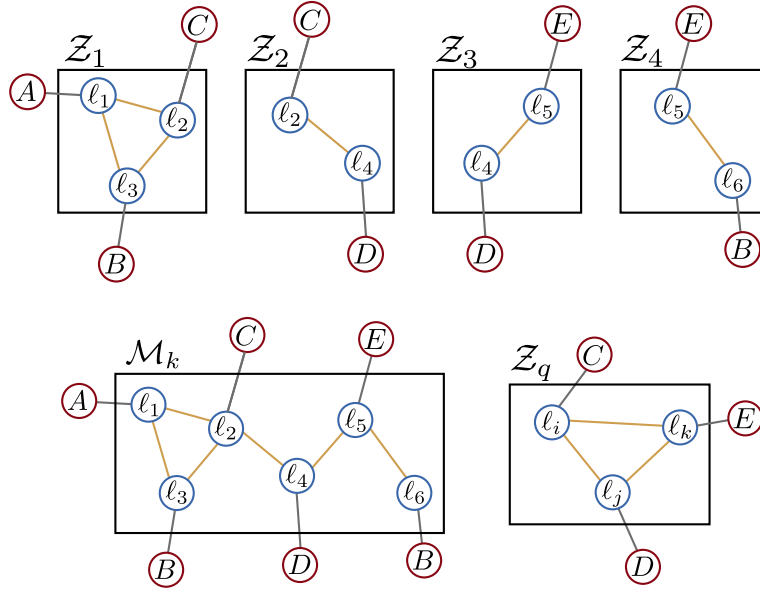


Figure 3.1: A sequence of simplified example observations is shown in the top row ( $Z_1$ ,  $Z_2$ ,  $Z_3$ ,  $Z_4$ ), along with the corresponding covisibility map,  $M_k$ , on the bottom-left, and the current query observation,  $Z_q$ , on the bottom-right. The figure also depicts which word (represented as  $A$ ,  $B$ ,  $C$ ,  $D$  or  $E$ ) is associated with each landmark,  $l_i$ .

In addition to these matrices, an inverted index between visual words and observations is maintained, for efficient look-up during the creation of virtual locations which will be explained further in Section 3.3. In the simple example of Figure 3.1, at time  $k$  there are 4 observations ( $Z_1 = \{l_1, l_2, l_3\}$ ,  $Z_2 = \{l_2, l_4\}$ ,  $Z_3 = \{l_4, l_5\}$ , and  $Z_4 = \{l_5, l_6\}$ ), then the clique matrix, adjacency matrix, and inverted index are given by:

$$C_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_4 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{array}{l} A : \{Z_1\} \\ B : \{Z_1, Z_4\} \\ C : \{Z_1, Z_2\} \\ D : \{Z_2, Z_3\} \\ E : \{Z_3, Z_4\} \end{array}$$

### 3.2.3 Discussion on Graph Structure

The covisibility graph representation captures the implicit structure of features in the world, with minimal dependence on observation trajectory. Many place recognition and mapping approaches rely on strong assumptions about the motion of the camera. For example, as described in Section 2.5, places are often represented as sequences consecutive images. Figure 3.2 demonstrates how these kinds of assumptions can be problematic. By grouping images

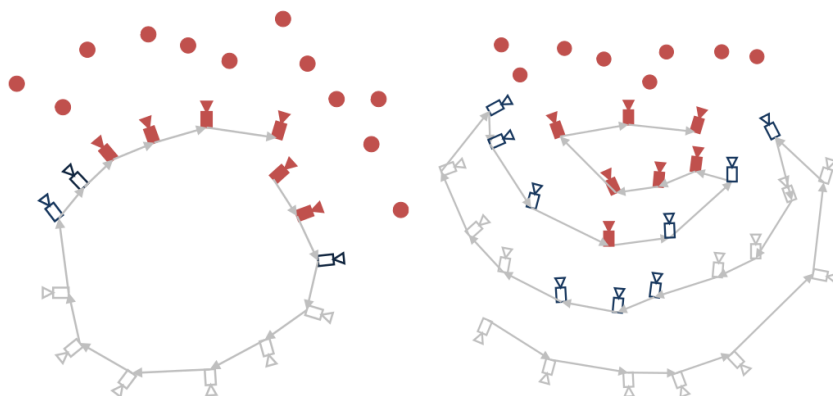


Figure 3.2: Examples of different observation trajectories. In the first, the motion is relatively linear, while in the second, the camera has a much more varied trajectory. Relevant image poses are shown in red. Assumptions about sequential observations clearly do not suffice in the second example. [Strasdat et al., 2011]

together based on their co-observation of landmarks, more consistent place representations can be achieved. The resulting covisibility graph structure additionally corresponds with the information needed for bundle adjustment applications, and can therefore be easily integrated within SLAM frameworks [Mei et al., 2010, Strasdat et al., 2011].

### 3.3 Identifying Places

At query time, virtual locations similar to the query image need to be retrieved from the covisibility graph, in order to be compared as a potential match. The idea is to find any clusters of landmarks in the map, which may have generated the given query. In this work, virtual locations are drawn from the graph for each specific query, and are therefore more closely linked to the actual arrangement of landmarks in the environment than individual images would be. This provides a more adaptable solution to place recognition, compared to methods which rely on pose-based location models. Defining places using covisibility avoids the need for time-based image groupings which rely on prior motion knowledge [Gálvez-López and Tardós, 2012] or more exhaustive key frame detection [Ranganathan and Dellaert, 2009]. This section provides an overview of how candidate virtual locations can be efficiently found in the map, based on a particular query.

#### 3.3.1 Inverted Index

Inverted indices are often used within retrieval applications, such that the complexity of the process does not depend on the size of the database [Manning et al., 2008]. This is important

for place recognition for robotics as well, as the size of the environment is unbounded, and execution time is often limited. When searching through images, index creation is possible due to the quantization of visual features into words [Sivic and Zisserman, 2003]. An example of an inverted index is shown in Section 3.2.2.

### 3.3.2 Retrieval by Landmark Covisibility

The process of finding relevant virtual locations will now be described, with the aid of Figures 3.1 and 3.3, and using the simple example introduced in Section 3.2:

- Using the inverted index, a list of observation cliques (columns in  $C_k$ ), containing words from the current query observation,  $Z_q$ , can be found. *In the example,  $Z_q = \{C, D, E\}$ , and so the relevant observation cliques are  $\{Z_1, Z_2, Z_3, Z_4\}$ .*
- Then, these clusters are extended to strongly connected cliques (sharing a certain percentage of covisible landmarks). This covisibility parameter represents the probability of re-observing landmarks between images. Refer to [Mei et al., 2010] for a discussion on the influence of this parameter, and the next subsection of this thesis for alternative clustering methods. *This will extend clique  $Z_2$  to  $Z_3$ , clique  $Z_3$  to  $Z_2$  &  $Z_4$ , clique  $Z_4$  to  $Z_3$  (which all co-observe 50% of their landmarks), and  $Z_1$  with nothing (because it doesn't share enough landmarks with any other cliques).*
- The result is sets of landmarks/words, which in turn, provide models for a set of virtual locations,  $\{\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_M\}$ . *Four virtual locations are produced for the given example, and are shown in Figure 3.3.*

Note that the set of virtual locations in Figure 3.3 provides a close match to the query shown in Figure 3.1, despite the fact that those landmarks were never directly covisible in any one observation. This emphasizes the adaptive nature of virtual locations, and the continuity

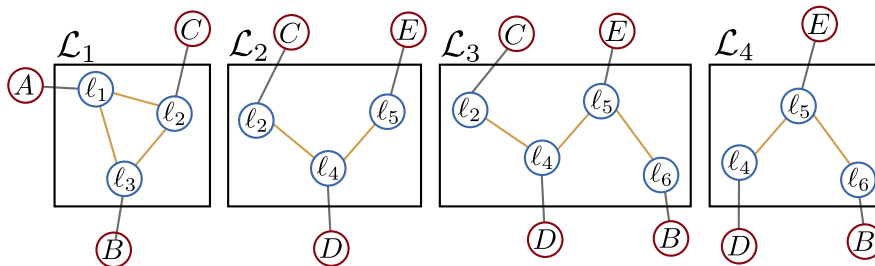


Figure 3.3: Given the query  $Z_q = \{C, D, E\}$  and covisibility map  $\mathcal{M}_k$  (both shown in Figure 3.1), will produce four virtual locations,  $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$  &  $\mathcal{L}_4$ .

which is achieved through covisibility. An experimental illustration of this will additionally be provided in the next chapter, in Section 4.8.3. Perhaps one limiting factor of this approach is the use of a parameter to determine the extent of clustering when forming the virtual locations. The next section discusses an alternative method of clustering, which eliminates the need for a covisibility parameter without changing the chosen probabilistic model.

### 3.3.3 Retrieval by Graph Clustering

The process of retrieving virtual locations requires some method of grouping relevant landmarks together into clusters. In Section 3.3.2 this was described by the use of a covisibility parameter, which extends clusters to strongly connected cliques sharing a certain percentage of covisible landmarks. This technique allows for very efficient clustering, but suffers in the sense that it relies on setting the covisibility parameter. In order to avoid using parameters for the expansion of virtual location clusters, more traditional graph clustering approaches can be used. A review of the theory surrounding graph clustering techniques is given by Schaeffer [2007]. Since clustering aims at grouping together nodes of a graph with similar properties and strong connectivity, it is inherently application dependent, and no single method is universally accepted. In addition, working with real-word image data means that the graphs tend to contain many different sources of error and noise, adding difficulty which the clustering algorithm needs to cope with.

One common metric used in clustering is the ratio of internal edges (those which connect two nodes in the cluster) to external edges (those which connect one node inside the cluster to another node outside the cluster). Therefore, landmarks are added to the cluster in a way which maximizes this parameter. In this application of forming virtual locations, landmarks are allowed to be a member of more than one location (overlap between locations is admissible), because of the normalization method discussed later in Section 4.5, simplifying the clustering process to a local (rather than global) optimization. The advantage of the method of the previous subsection is efficiency, whereas the advantage of using this clustering technique is a parameter-free technique. Tests were carried out to investigate how the different clustering techniques affect the results, and the outcome (seen in Table 4.1 of Section 4.8.4) shows that results are similar for both methods of clustering.





## Chapter 4

# Modelling Locations with Bags of Words

### 4.1 Introduction

Chapter 3 explained how a map of the environment is built, and furthermore how a set of candidate locations can be identified and retrieved from the resulting covisibility graph. The current section aims at developing a probabilistic observation model of places, in order to evaluate if any of the candidate locations match a query location. The clusters of detected visual words which were introduced in Section 3.3 represent observations of places, and here, a probabilistic bag-of-words approach is used to compare the query to the set of candidate virtual locations.

The probability of a location generating the given query observation can be found using Bayes' theorem:

$$P(\mathcal{L}_i|\mathcal{Z}_q) = \frac{P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i)}{P(\mathcal{Z}_q)} \quad (4.1)$$

where  $\mathcal{L}_i$  is a particular virtual location, and  $\mathcal{Z}_q$  is the query observation given by a set of visual words  $\{z_1, z_2, \dots, z_N\}$ . The development of each term in Equation 4.1 is given throughout this chapter. Section 4.2 explains how the existence and observation of visual elements in the scene are modeled, followed by the introduction of a novel model for the observation likelihood given a location in Section 4.3, a discussion of normalization techniques in Section 4.5, an overview of the sampling procedure in Section 4.7, and a description of how location priors are estimated in Section 4.6.



## 4.2 Modelling Scene Elements

An observation of the query location,  $\mathcal{Z}_q$ , is represented as a binary word-observation-vector of length equal to the number of words in a visual dictionary,  $\mathcal{V}$ :

$$\langle z_1^q, z_2^q, \dots, z_{|\mathcal{V}|}^q \rangle$$

And the observation of a virtual location,  $\mathcal{Z}_{\mathcal{L}}$ , is represented analogously as:

$$\langle z_1^{\mathcal{L}}, z_2^{\mathcal{L}}, \dots, z_{|\mathcal{V}|}^{\mathcal{L}} \rangle$$

where each  $z_n$  is set to one if the  $n^{\text{th}}$  word in the dictionary was present in the observation and zero otherwise. The visual dictionary,  $\mathcal{V}$ , is pre-trained with sample features, using a clustering algorithm to define a set of visual words that span the relevant feature space [Sivic and Zisserman, 2003].

Note that the negative information (lack of a word) is explicitly considered; however frequency information (word count) is removed from the observations. The reason for this is twofold. Firstly, ignoring word frequencies is justified by the fact that features tend to appear in bursts, where most of the information is provided by the presence (or lack of presence) of the word, rather than the number of occurrences of the word [Schneider, 2004]. As an example, objects such as bricks or leaves tend to be present in large multiples which would have an overwhelming effect on the outcome of comparison methods between locations, whereas seeing one leaf or brick will already give a good indication of what is present in the scene.

Secondly, these feature vectors are binary and of fixed length, allowing for a simple representation which does not vary depending on the number of words. In the context of text classification, this corresponds to the use of a Bernoulli model, as opposed to a Multinomial model which requires an assumption of fixed document length (i.e. all locations contain the same number of words), which does not hold in this context [Eyheramendy et al., 2003, Schneider, 2004].

Observations are modelled under a generative scheme, where locations are assumed to be composed of a set of existing visual elements which are observed using an imperfect sensor, resulting in our observation vectors  $\mathcal{Z}$ , as is also done in [Cummins and Newman, 2008] (see Figure 4.1 for reference). In this case,  $e_n$  is introduced as a hidden layer, which represents the true existence of scene elements generated by  $\mathcal{L}_i$ . The observations  $z_n$  represent (possibly

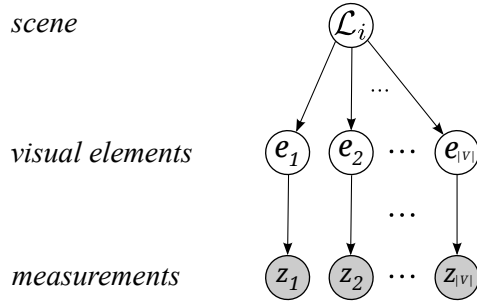


Figure 4.1: Graph of the observation model, with observed variables shaded in gray. A location consists of a set of visual elements  $e_n$ , which are then observed by an imperfect sensor, giving measurements  $z_n$ . The possible set of visual elements is defined by a visual dictionary of size  $|\mathcal{V}|$ , and visual features  $e_n$  and  $z_n$  take on Boolean values based on existence or observation respectively.

imperfect) measurements of these underlying scene elements. The choice for a generative model is driven by its capability to incorporate aspects such as location priors, various types of measurements, and sensor models in a principled way which generalises well to unknown environments [Bishop, 2007].

### 4.3 Estimating Observation Likelihoods

For computational reasons, a conditional independence (Naive-Bayes) assumption is made about individual word observations, simplifying the observation model shown in Figure 4.1, such that when conditioned on the location, the likelihood of one word does not depend on any other words. Although this assumption is generally false, it has been shown to provide meaningful results when compared to more complex models [Cummins and Newman, 2008]. Due to this conditional independence assumption, the observation likelihood,  $P(\mathcal{Z}_q|\mathcal{L}_i)$ , can be reduced to a product of individual word likelihoods, as given in Equation 4.2a.

$$P(\mathcal{Z}_q|\mathcal{L}_i) \approx \prod_{n=1}^{|\mathcal{V}|} P(z_n^q|\mathcal{L}_i) \quad (4.2a)$$

$$\approx \prod_{n=1}^{|\mathcal{V}|} \sum_{\alpha \in \{0,1\}} P(z_n^q|e_n=\alpha, \mathcal{L}_i) P(e_n=\alpha|\mathcal{L}_i) \quad (4.2b)$$

$$\approx \prod_{n=1}^{|\mathcal{V}|} \sum_{\alpha \in \{0,1\}} P(z_n^q|e_n=\alpha) P(e_n=\alpha|z_n^L) \quad (4.2c)$$

The observation likelihood can therefore be written using the sum rule of probability as Equation 4.2b, which simplifies to Equation 4.2c under the assumption that detection is

independent of location:  $P(z_n^q | e_n = \alpha, \mathcal{L}_i) = P(z_n^q | e_n = \alpha)$  (see Figure 4.1), and by estimating the existence of an element using the prior observations of a location:  $P(e_n = \alpha | \mathcal{L}_i) \approx P(e_n = \alpha | z_n^\ell)$ , as in [Cummins and Newman, 2008, 2011].

Although the complexity grows with the number of words in the vocabulary, the sparse nature of observations can be used to greatly reduce computation in most cases. Note that the model could be further extended to remove the conditional independence assumption between words, for instance by using a Chow-Liu tree as done in [Cummins and Newman, 2008]. Doing so tends to improve results slightly in the presence of minor scene changes, but for simplicity has not been implemented here. See [Cummins and Newman, 2008] for a thorough analysis of such observation models.

## 4.4 Modelling Visual Observations

The final terms in Equation 4.2c are defined by the underlying nature of how visual observations are made and modelled. There are a number of ways to estimate these values, and the details are discussed in the following subsections.

### 4.4.1 Observation Given Existence

In previous work [Stumm et al., 2013, Cummins and Newman, 2011], the term  $P(z_n | e_n = \alpha)$  represents the sensor detection probabilities, which are set as parameters by the user. For example, the true positive probability of observing an element which exists,  $P(z_n = 1 | e_n = 1)$ ; and the false positive probability of observing an element which doesn't exist,  $P(z_n = 1 | e_n = 0)$ , are generally pre-calibrated. This leaves the likelihood of a particular element existing in the location,  $P(e_n = \alpha | \mathcal{L}_i)$ , to be estimated from the observation we have of the virtual location,  $\mathcal{Z}_{\mathcal{L}_i}$ , using the sensor model and prior knowledge about how common the element is [Glover et al., 2012]:

$$P(e_n = \alpha | \mathcal{L}_i) = P(e_n = \alpha | z_n^\ell) \tag{4.3a}$$

$$= \frac{P(z_n^\ell | e_n = \alpha) P(e_n = \alpha)}{P(z_n^\ell)} \tag{4.3b}$$

$$= \frac{P(z_n^\ell | e_n = \alpha) P(e_n = \alpha)}{\sum_{\beta \in \{0,1\}} P(z_n^\ell | e_n = \beta) P(e_n = \beta)} \tag{4.3c}$$

However, implementing this requires estimating  $P(e_n)$ , which is not possible in practise as the true existence is not known. In effect, other works such as [Cummins, 2009, Glover et al., 2012, Stumm et al., 2013], substitute  $P(z_n)$  for  $P(e_n)$ , undermining the observation model they define.

#### 4.4.2 Existence Given Observation

One possible way around this problem of needing an estimate of  $P(e_n)$ , would be to redefine the observation model in terms of  $P(e_n|z_n)$ , rather than  $P(z_n|e_n)$ . This means that the problem is flipped, and now  $P(e_n=\alpha|z_n^e)$  in Equation 4.2c is predefined, and  $P(z_n^q|e_n=\alpha)$  is calculated as follows:

$$P(z_n^q|e_n=\alpha) = \frac{P(e_n=\alpha|z_n^q)P(z_n^q)}{P(e_n=\alpha)} \quad (4.4a)$$

$$= \frac{P(e_n=\alpha|z_n^q)P(z_n^q)}{\sum_{\beta \in \{0,1\}} P(e_n=\alpha|z_n=\beta)P(z_n=\beta)} \quad (4.4b)$$

This removes an explicit inclusion of  $P(e_n)$ , as well as having a number of other consequences. Now, the required parameters are all defined in the sense of having access to  $z_n$ , the *observed* variable, rather than  $e_n$ , the *hidden* variable. This includes the probability of existence *given* the observation,  $P(e_n|z_n)$ .

#### 4.4.3 Empirical Evaluation of Observation Models

In order to investigate the empirical effect of these changes, the FAB-MAP code [OxfordMRG, 2013] was augmented to incorporate the new model, and both versions were tested across a range of parameters (with parameter settings  $P(z_n=1|e_n=1) > P(z_n=1|e_n=0)$  and  $P(e_n=1|z_n=1) > P(e_n=1|z_n=0)$  in each case). The results are seen in Figure 4.2, where the value of maximum recall at 100% precision was plotted for varying parameter settings, for each model, for four different datasets (Begbroke, City Centre, New College, and KITTI). For more information on the datasets and precision-recall metrics, please refer to Appendix B and Appendix C. The plots in the left column show the recall results for fixed values of  $P(z_n|e_n)$  (corresponding to Equation 4.3), and the plots in the right column show the recall results for the novel model where  $P(e_n|z_n)$  is fixed (corresponding to Equation 4.4). From these plots, one can see that recall results remain stable across a wider selection of parameters, reducing

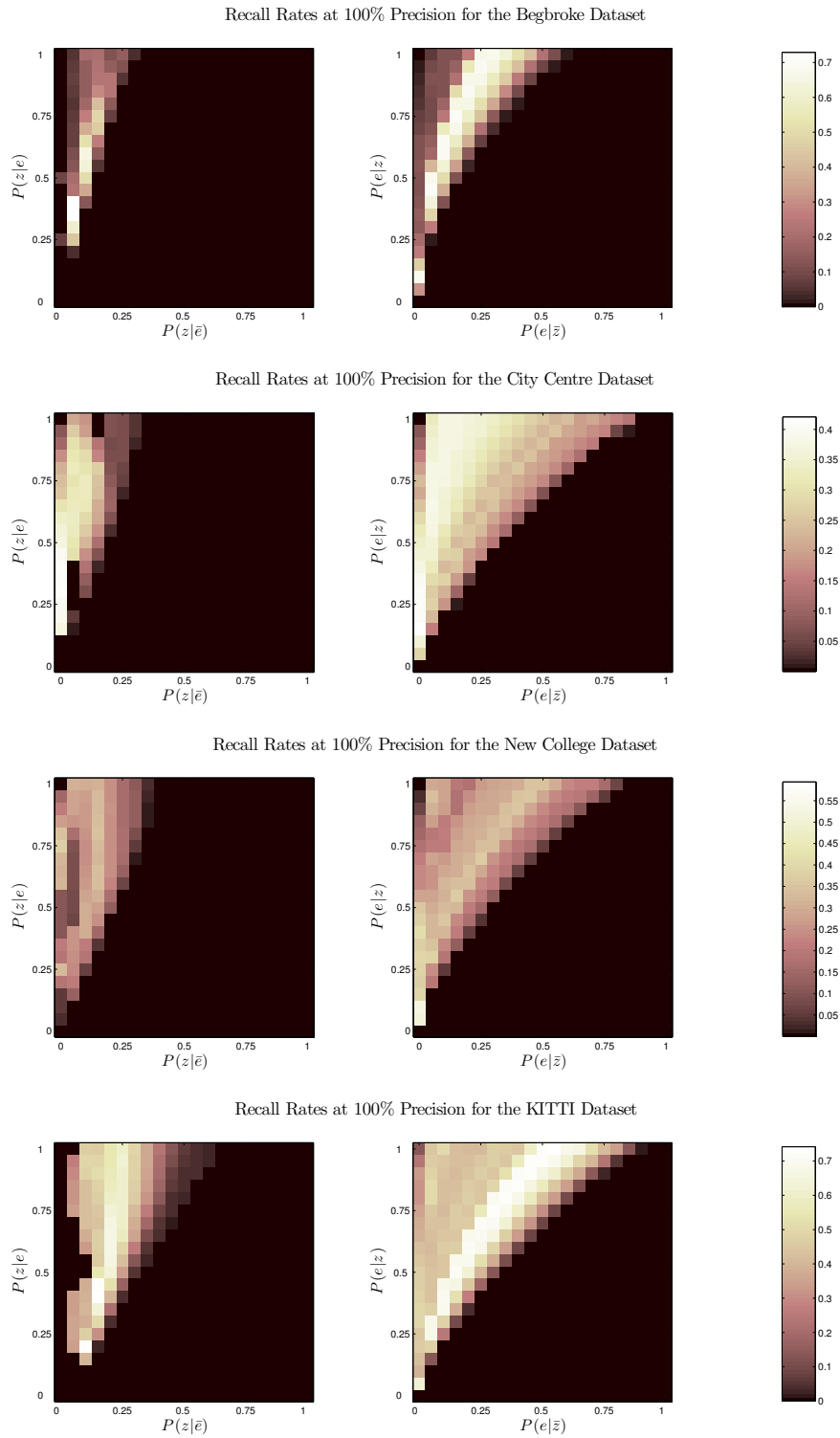


Figure 4.2: A comparison of maximum recall results at perfect precision for a variety of possible parameter settings, computed for four different datasets. The left column shows the results of the model implied by Equation 4.3, whereas the right column shows the results of the model implied by Equation 4.4. Colour bars are also given to indicate recall values for each dataset.

the sensitivity of results to parameter settings while maintaining recall performance. Based on these experiments, the parameters  $P(e_n|z_n) \approx 0.8$  with  $P(e_n|\bar{z}_n) \approx 0.3$  are of interest, with finer scale tests providing  $P(e_n|z_n) = 0.78$  with  $P(e_n|\bar{z}_n) = 0.32$  as the selected parameters. All results shown in Figure 4.2 were obtained using the Naive-Bayes version of FAB-MAP, with the default motion model and default sample data. In addition, the ground truth was given by accepting all loop-closures within a generous radius of 10 m (to compensate for some significant errors in the GPS data), and the 10 most recent images were masked from consideration.

## 4.5 Normalization using Sample Locations

Working with true (normalized) probabilities is essential for the decision making process in the context of loop-closure for mobile robots, as false loop-closures result in fundamental mapping and localization errors. However, accurate normalization in the presence of such complex and high-dimensional observations of scenes requires careful treatment using previously obtained sample observations [Bishop, 2007] which will be explained throughout this section.

The formulation presented here differs from the typical treatment of classification problems, where only the best (maximizing) class is assigned to an observation (as in Equation 4.5, for example), and therefore normalization is not required and rarely calculated in practice.

$$c_{MAP} = \arg \max_{c \in \mathbb{C}} P(c|z) = \arg \max_{c \in \mathbb{C}} \frac{P(z|c)P(c)}{P(z)} = \arg \max_{c \in \mathbb{C}} P(z|c)P(c) \quad (4.5)$$

where  $c \in \mathbb{C}$  is a set of classes,  $z$  is a given observation, and  $c_{MAP}$  represents the maximum a posteriori estimate for the class [Manning et al., 2008].

However, within the application of place recognition and loop-closure, the number of locations which match the query observation is unknown, and there may even be no matches; meaning that it would be incorrect to always associate the maximizing location to the query. In addition, the severity of making any incorrect data associations can be further motivation for basing decisions on the results of posterior probabilities, only fusing locations if the probability lies above a certain confidence threshold. Therefore, the denominator in Equation 4.1,  $P(\mathcal{Z}_q)$ , is required. As previously mentioned, due to the high-dimensional nature of visual observations of places, estimating this term in practice requires the use of sample locations. These sample locations function as a representation of all other locations in the world,  $\bar{\mathcal{L}}_i$ . Using samples, the likelihood of the observation coming from any other place,  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$ , is

calculated and then  $P(\mathcal{Z}_q)$  is found through marginalization:

$$P(\mathcal{Z}_q) = P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i) + P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)P(\bar{\mathcal{L}}_i) \quad (4.6)$$

Implicitly, the sample locations provide an indication as to how common or ambiguous an observation is, and help deal with a problem known as perceptual aliasing. As an example, in a city, something like a brick wall does not provide much information about where one might be since it is commonly seen throughout cities. On the other hand, a distinct fountain or sculpture will. In the same way, a representative group of sample locations should augment the posterior probability according to how often elements of the observation were seen throughout the samples. A more detailed intuition behind working with sample locations is given in Section 4.7.

#### 4.5.1 Discussion on Normalization Models

Note that the presented method of normalization makes no reliance on the robot’s current map, since it makes no assumption on the number of matching locations in the map, or the number of previously seen locations. This provides an advantage to other place recognition frameworks such as FAB-MAP [Cummins and Newman, 2008] which normalize over all locations in the map (plus an unknown location,  $\mathcal{L}_u$ ). In these approaches, probabilities are summed across locations:

$$P(\mathcal{L}_1|\mathcal{Z}_q) + P(\mathcal{L}_2|\mathcal{Z}_q) + \dots + P(\mathcal{L}_u|\mathcal{Z}_q) = 1 \quad (4.7)$$

whereas here in this work,

$$P(\mathcal{L}_i|\mathcal{Z}_q) + P(\bar{\mathcal{L}}_i|\mathcal{Z}_q) = 1. \quad (4.8)$$

Equation 4.7 is based on an underlying assumption that each location is only represented once, thereby assuming no loop closures will be missed, and that the map is accurate. As previously mentioned, there may in fact be more than one match to the query. This can happen when a previous loop-closure is missed – leaving two or more representations of the location in the covisibility map. In addition, images immediately surrounding the query do not need to be masked (removed from consideration) as commonly done ([Cummins and Newman, 2011, Angeli et al., 2008b]), since these local matches will not steal probability mass from others. Another benefit of this technique is that probabilities no longer need to be normalized over all locations in the map, leaving room for efficiency improvements over other techniques. It

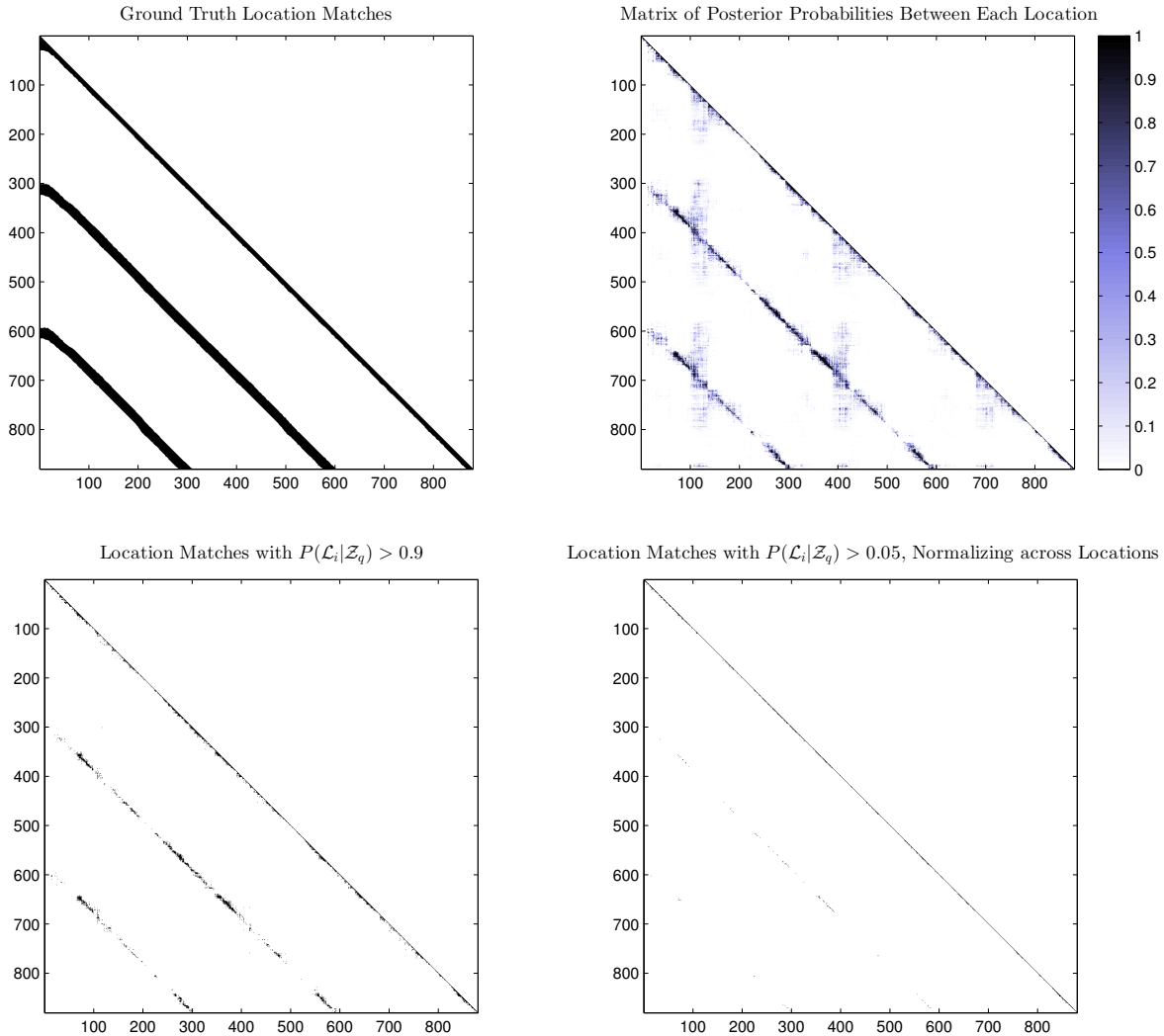


Figure 4.3: Depiction of place recognition results for the three-loop Begbroke dataset, using two different normalization methods. The first plot shows the ground truth location matches. The second and third plots show the results given by  $P(\mathcal{L}_i|\mathcal{Z}_q)|_{Eq.4.8}$ . In the second plot, probability scores are indicated by the colour bar, whereas in the third plot scores are thresholded at 0.9. The fourth plot shows the results given by  $P(\mathcal{L}_i|\mathcal{Z}_q)|_{Eq.4.7}$ , thresholded at 0.05. Note that when normalizing across all locations in the map (fourth plot), the probability mass is split accordingly, therefore generally resulting low values.

should be noted, however, that this method of normalization opens up the risk of obtaining more than one false-positive loop-closure per location, whereas in previous work, at most one loop-closure could be detected per location.

## 4.5.2 Empirical Evaluation of Normalization Models

In order to compare the effects of each normalization method, the FAB-MAP code [OxfordMRG, 2013] was augmented again, to incorporate the normalization method given by Equation 4.6.



Figure 4.3 shows the results for the Begbroke dataset using the two different normalization methods (again, refer to Appendix B and Appendix C for details about the datasets and test metrics). The first plot shows the expected ground truth based on GPS data, with location matches shown by black lines. The second plot shows the matrix of posterior probability values between each location under the marginalization scheme of Equations 4.8, with values shaded between 0 (white) and 1 (black). The third and fourth plots show the location matches resulting from thresholding the probabilities ( $P(\mathcal{L}_i|\mathcal{Z}_q)|_{Eq.4.8}$  and  $P(\mathcal{L}_i|\mathcal{Z}_q)|_{Eq.4.7}$ ) at 0.9 and 0.05 respectively, where the third plot is made using the normalization given by the marginalization of Equation 4.8 and the fourth plot is made using the marginalization of Equation 4.7. Results show that the chosen method allows for multiple location matches, whereas the other method has trouble dealing with many instances of the same location in the map. In these tests, the Naive-Bayes version of FAB-MAP was used, with the default sample data. However, no motion model was used, and there was no masking of recent images, in order to exclude effects of strong assumptions on motion.

Further investigation into the second plot of Figure 4.3, indicates that the areas where the diagonals are thinner correspond to locations where the robot turns corners (leaving fewer visual matches, given a forward-facing camera). In addition, the faint vertical streaks of confusion in the probability matrix correspond to locations which do not have a discriminative appearance. Figure 4.4, shows two examples of images from this region, where one can see many matched visual words (shown in red, rather than blue), despite showing different features. However, referring back to Figure 4.3, one can see that despite many matched words, the final match probability remains low, which can be seen as the streaks disappear completely after thresholding. This also serves as an illustration of how normalization can deal with the problem of perceptual aliasing.

Another implication of this normalization technique is shown in Figure 4.5. Here the matched visual words (shown in red rather than blue) and resulting posterior probabilities are shown for a series of images, as a location which matches the query is approached. Under the normalization scheme of Equation 4.7, at most one location in the map can have a significant probability mass, leaving the rest with low scores (in this scenario, the most recent location in the map has a score of  $P(\mathcal{L}_i|\mathcal{Z}^q) = 0.6$ ). When probabilities are no longer normalized across all locations in the map, the behaviour of probabilities becomes more intuitive, since now locations with similar appearances can have similar scores and there is a natural progression of probabilities as a location is approached.



Figure 4.4: Examples of images from the Begbroke sequence which correspond to the regions of confusion in first matrix of Figure 4.3, due to their non-discriminative appearance. The images contain many matched visual words (shown in red), despite depicting different locations. However, these locations tend to produce low matching probabilities regardless, which indicates a low confidence due to common appearance.

## 4.6 Location Priors

For the framework presented in this work, the location prior is estimated without the use of any motion prediction models. This is in part due to the fact that this work is ideally meant to remain robust to unpredictable movements and kidnapped robot situations. In practice, the effect of this prior is not especially strong, and it is therefore not a critical parameter. This is evident when comparing the order of magnitude of the observation likelihood (a product of probabilities over thousands of visual words) to that of a location prior. The weak influence of this term is also documented in Cummins and Newman [2008]. Therefore most of the prior probability is assigned to unobserved locations, conservatively favoring unobserved locations (to avoid false positives). Other cues could be used to more accurately estimate the location prior; such as global visual features or additional sensory information.

## 4.7 Sampling

As discussed in Section 4.5, the success of this framework relies on correct normalization of posterior probabilities, which is done here using sampling techniques. Sampling from such a high-dimensional space poses a variety of difficulties, including obtaining representative samples, modelling the sample locations properly, and working efficiently with the samples.

The normalization term in Equation 4.6 requires  $P(\mathcal{Z}_q | \bar{\mathcal{L}}_i)$  which is calculated from the

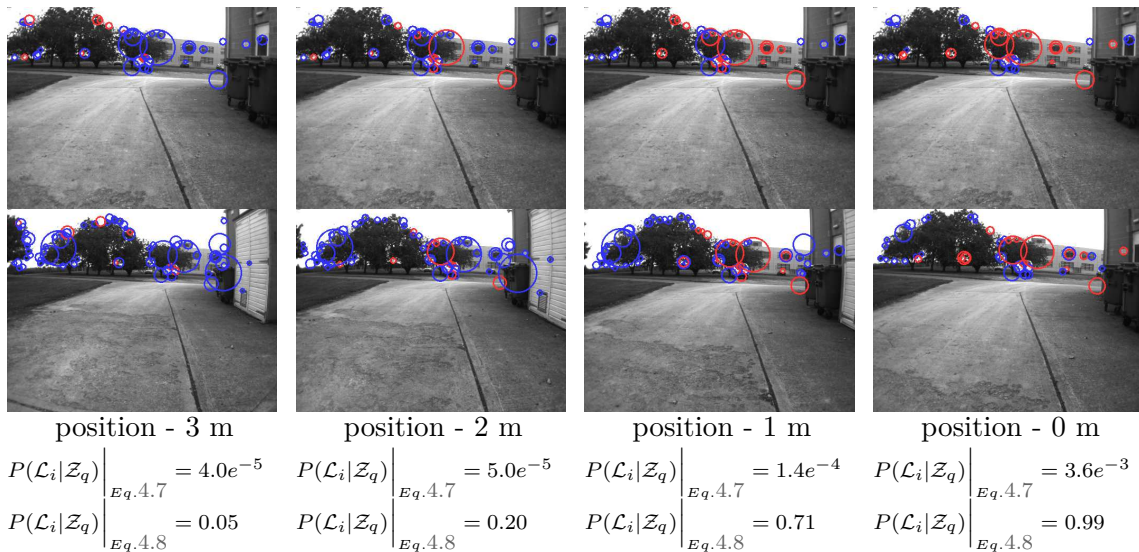


Figure 4.5: Demonstration of matched features and posterior probability values as a location which matches the query is approached. The query image is repeated in the top row, with four other locations preceding a match to the query shown in the second row. Matched visual words are shown in red, while unmatched visual words are shown in blue. The resulting match probabilities are shown below each image pair.

sample set as follows,

$$P(\mathcal{Z}_q|\bar{\mathcal{L}}_i) = \sum_{s=1}^{N_s} \frac{P(\mathcal{Z}_q|\mathcal{L}_s)}{N_s} \quad (4.9)$$

with  $N_s$  being the number of samples,  $\mathcal{L}_s$  being the  $s^{th}$  sample location, and  $P(\mathcal{Z}_q|\mathcal{L}_s)$  is subsequently given by Equations 4.2 and 4.3.

One straightforward way to generate sample locations is to collect images from the robot in the same manner as the typical use-case scenario. However, this may not always be possible, and it is often difficult to represent the full range of possible scenes. Therefore it may be necessary to combine sample images from a variety of experiments, as well as other resources such as online map repositories. This task benefits from prior knowledge about the types of environments that the robot will operate in, and should contain the scope of features which the robot is expected to encounter. These samples are what allows the system to understand which features are distinctive and avoid perceptual aliasing, by augmenting the impact of features based on how common they are in the data. Note that sample locations require no extra processing in comparison to the run-time virtual locations, since they require no supplementary labels or information.

For the experiments presented in this work, samples were taken from a variety of pub-

licly available datasets, as well as images obtained from Google Street View<sup>1</sup>. The same samples were reused across each test scenario (with the exclusion of images coming from the corresponding test set).

One of the difficulties in using samples to estimate the likelihood of unknown locations is knowing how many samples are required in order to get a reasonable estimate. Unfortunately this remains an open question. There are no guarantees on the quality of the estimate, and the number of samples required will vary, depending on the extent of the world which the robot will operate in. In general, the more samples that are available, the better.

Experience from the experiments presented here, as well as the work documented by Cummins and Newman [2008, 2011], shows that useful results can be obtained using a feasible number of sample locations, especially if some prior knowledge is known about the expected environment and therefore samples to use (*e.g.* urban, indoor, rural, etc.). For the results presented in this thesis, approximately 3000-5000 samples were used (the amounts vary across experiments because images from the current test set were removed from the sample set in each case).

## 4.8 Experimental Evaluation and Results

In order to analyse the performance of the proposed approach (hereby referred to as CovisMap), it was tested on each of the datasets described in Appendix B. The system is also compared to two widely known place recognition methods, FAB-MAP [Cummins and Newman, 2008] and SeqSLAM [Milford, 2013], to investigate the implications of the various assumptions and the relative behaviour of each system. To understand the details of test configurations, the implementation details are explained in Appendix A, the tested datasets are described in Appendix B, and precision-recall metrics are explained in Appendix C.

In this section, precision-recall characteristics are discussed in Section 4.8.1, with direct comparisons to FAB-MAP and SeqSLAM. This is then followed by detailed discussions on normalization methods, trajectory invariance, and location retrieval in Sections 4.8.2-4.8.4.

### 4.8.1 Comparison with State-of-the-Art

This section presents precision-recall results on five different datasets, for CovisMap, FAB-MAP, and SeqSLAM. As mentioned in Section 1.2, for robotic SLAM applications, the primary goal is to increase the achievable recall rate while maintaining perfect precision. The

---

<sup>1</sup><https://www.google.com/maps/views>

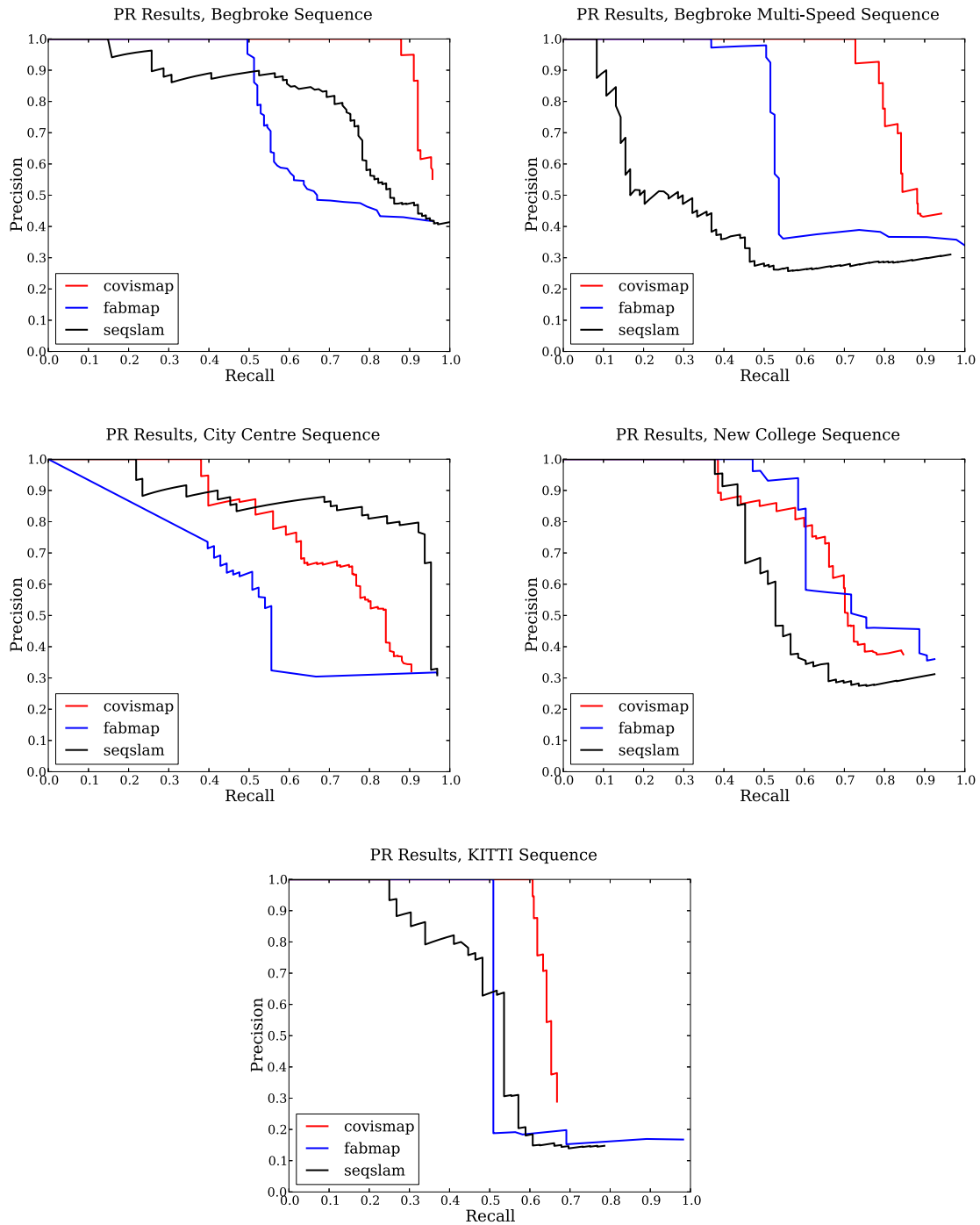


Figure 4.6: Precision-recall results for three methods (CovisMap, FAB-MAP, and SeqSLAM), tested on the Begbroke, Begbroke Multi-Speed, CityCentre, New College, and KITTI datasets.

details of the definition and calculation of precision-recall curves is given in Appendix C. Each result is discussed, and example images are given to provide insight about how each system performs.

Figure 4.6 shows the precision-recall curves for the Begbroke, Begbroke Multi-Speed,



Figure 4.7: A subset of images from a false positive sequence match example from SeqSLAM on the Begbroke dataset.



Figure 4.8: Example of a significant perspective changes in the Begbroke dataset. These perspective changes introduce difficulties for algorithms which rely on direct image comparisons, such as SeqSLAM.

CityCentre, New College, and KITTI datasets as the loop-closure threshold is varied. The details of the precision-recall evaluation are described in Appendix C. One can see that in general, the CovisMap framework achieves a higher recall, especially while maintaining 100% precision.

When comparing the two Begbroke datasets, recall results drop for each system on the Multi-Speed sequence. The drop in recall is most severe for the SeqSLAM dataset, which relies on consistent sequences to produce good scores. In both Begbroke datasets, SeqSLAM and FAB-MAP tend to suffer from perceptual aliasing more than CovisMap. In the case of FAB-MAP, perceptual aliasing arises because of the use of single-image locations which often have very similar appearance (see Figure 4.4 for example). SeqSLAM, on the other hand, has difficulty because many incorrect sequences can look similar on the level of global image comparisons (see Figure 4.7), while correct matches often look different due to changes in perspective (see Figure 4.8).

The City Centre dataset is also challenging for all three algorithms because of perceptual



Figure 4.9: Example of two scenes which cause false-positives due to perceptual aliasing in the City Centre dataset.



Figure 4.10: Due to the larger image spacing in the City Centre and New College datasets, there are often significant position offsets, creating difficulties for the SeqSLAM algorithm.

aliasing. Figure 4.9 shows the kind of extreme examples of perceptual aliasing that arise when using the bag-of-words methods in FAB-MAP and CovisMap which ignore position information from the features. Other than a few high-scoring false-positives, the SeqSLAM is able to obtain a very high recall rate at almost 90% precision because it does not suffer from such feature-based aliasing. However, these kinds of false-positives can generally be suppressed by post-processing matched locations with a check for geometrical consistency [Sivic and Zisserman, 2003, Cummins and Newman, 2011]. This is not done here, in order to maintain a clear analysis of the underlying approaches.

For both the City Centre and New College datasets, the relatively large image spacing means that the direct image comparisons used in SeqSLAM may give low scores because of position offsets. Refer to Figure 4.10 for an example from the New College dataset, where the same location can have strong image difference scores.

Performance on the New College Dataset is pretty similar for all three methods. CovisMap has difficulty distinguishing between different positions when there are scenes with a wide visibility and repetitive features. Figure 4.11 shows this kind of false-positive response

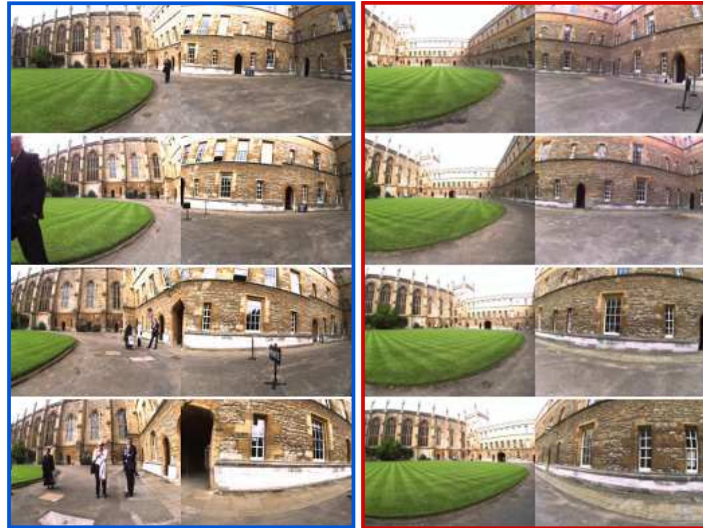


Figure 4.11: Locations with a wide visibility and repetitive features can cause false-negatives under the bag-of-words model, especially when considering larger sets of images. The loss of structure in the bag-of-words model reduces the distinctiveness of each location and causes perceptual aliasing. Here is a false-positive example given by the CovisMap algorithm on the New College dataset. The query location is shown on the left, and the retrieved location on the right (only a subset of images from each location are shown).

given by CovisMap. However, although during testing these are considered to be two different locations, there are indeed many common landmarks between the two scenes. This raises the question of how to best evaluate loop-closure, since some applications may require accurate position-based matches, while others only require matched landmarks. The loss of structure when using a bag-of-words model reduces the distinctiveness of each location, and increases the risk of false-positives. False-positives like that in Figure 4.11, or even more severe mismatches could possibly be avoided by including more structure during comparison. Ideas related to this are subject of current on-going work.

In the New College and the KITTI datasets, there are often relatively short sequences of loop-closures (unlike the Begbroke and City Centre datasets which mostly traverse the same sequence multiple times). This can result in recall problems for the SeqSLAM algorithm which relies on a fixed sequence length. Figure 4.12 shows an example of this from the KITTI dataset, where the middle portion of two sequences overlap, while the beginning and ends of the sequences vary, resulting in a relatively low score.

### 4.8.2 Investigating Relative Scoring Methods

The benefit of rigorous probabilistic methods are especially clear when looking at the thresholds used to determine matches. Figure 4.13 shows the scores from several passes by a query





Figure 4.12: Example of a loop-closure sequence which is shorter than the sequence length used in the SeqSLAM algorithm, resulting in relatively low scores and therefore missed loop-closures. The left column shows the image sequence from one pass, and the right shows the image sequence from a second pass (only a subset of images from the sequence are shown).

image from third loop: (current position)		image from third loop: (previous position)		image from second loop:		image from first loop:		image from a different location	
covismap	--	covismap	1.0	covismap	0.998	covismap	0.991	covismap	1e-05
fabmap	--	fabmap	0.865	fabmap	6e-07	fabmap	1e-05	fabmap	1e-05
seqslam	--	seqslam	0.819	seqslam	0.207	seqslam	0.186	seqslam	0.098

Figure 4.13: Example of a location from the Begbroke sequence which is passed several times, and the matching scores resulting from a query generated during the third pass. Scores are shown for the system described in this thesis (CovisMap), FAB-MAP [Cummins and Newman, 2008], and SeqSLAM [Milford, 2013]. The query location is shown on the left in grey, followed by the image just before the query location, an image from the previous pass, an image from the first pass, and a negative location example.

location in the Begbroke sequence from each of the tested methods. In this example, FAB-MAP struggles to recognize all instances of the location because the posterior probabilities must be normalized across all locations (refer to Section 4.5 for details). Therefore, most of the probability mass is assigned to the most similar location just before the query location, resulting in low probabilities being assigned to other instances of the same location. In the case of SeqSLAM, the scores are still normalized between 0 and 1, but do not represent probabilities. In the example in Figure 4.13, each other instance of the query location receives scores between 0.2 – 0.8, whereas the negative location example has a score of about 0.1. The generative model developed in Sections 4.1-4.7 allows the CovisMap algorithm to retrieve each instance of the query location with posterior probabilities above 0.99, while assigning a



Figure 4.14: Given the query shown in green, the most probable virtual locations from two separate traverses of a street are shown (with only three images displayed per location for clarity). Both the query and the first pass are traversed on the right side of the street, while the second pass is traversed on the left side. In addition, the images collected from the query were generated from a much slower and erratic traverse, resulting in more images representing the same traverse yet poses no problems for the system.

near zero probability to the negative location.

### 4.8.3 Investigating Trajectory Invariance

In this section, the system is investigated, to see whether it can handle changes in the motion between different traverses of the same area; namely offsets in position and variations in speed (or image frame-rate). This is done by traversing a section of street in several different ways: once quickly along the right side of the street, once quickly along the left side of the street, once down the right side of the street with inconsistent speed, and once down the left side of the street with inconsistent speed (including backtracking). The chosen street is completely surrounded by houses on either side, in order to present significant perspective changes between traverses on the right and left sides of the street. The images were collected using a basic hand-held camera, and therefore the images only roughly point in the same direction in each traverse, adding even more variation. Example images of the environment can be seen in Figures 4.14 and 4.15. In this scenario, the system is shown to implicitly handle variations in speed (since locations are created based on covisibility and not a predefined number of images or time scale), as well as perspective changes (when relying on feature based models).

Figures 4.14 and 4.15 show two examples of queries, along with the virtual locations



Figure 4.15: Given the query shown in green, the most probable virtual locations from three separate traverses of a street are shown (with only four images displayed per location for clarity). Both the query and the second pass are traversed on the left side of the street, while the first and third passes are traversed on the right side. In addition, the images collected from the query were generated from a much slower and erratic traverse (including backtracking), resulting in more images representing the same traverse yet poses no problems for the system.

deemed most probable from each previous traverse of the street (only a few representative images from the start, middle, and end of each location are shown for clarity). In both cases, the query locations were collected from more erratic traverses, where speed and sometimes direction was varied (although the camera was always facing forward). Note that there were no data associations between any separate traverses of the street, and therefore each individual sequence is independent from the others.

One can see that even though each instance of the same location can contain varying amounts of images, the scope remains consistent in each case. Additionally, the corresponding probabilities reflect the degree of similarity, while producing no false positives above 99%. The system defines locations based on covisible features, an attribute which should theoretically be independent of direction and speed (given that there remains enough overlapping features between images), allowing the system to implicitly cope with such variations. The algorithms

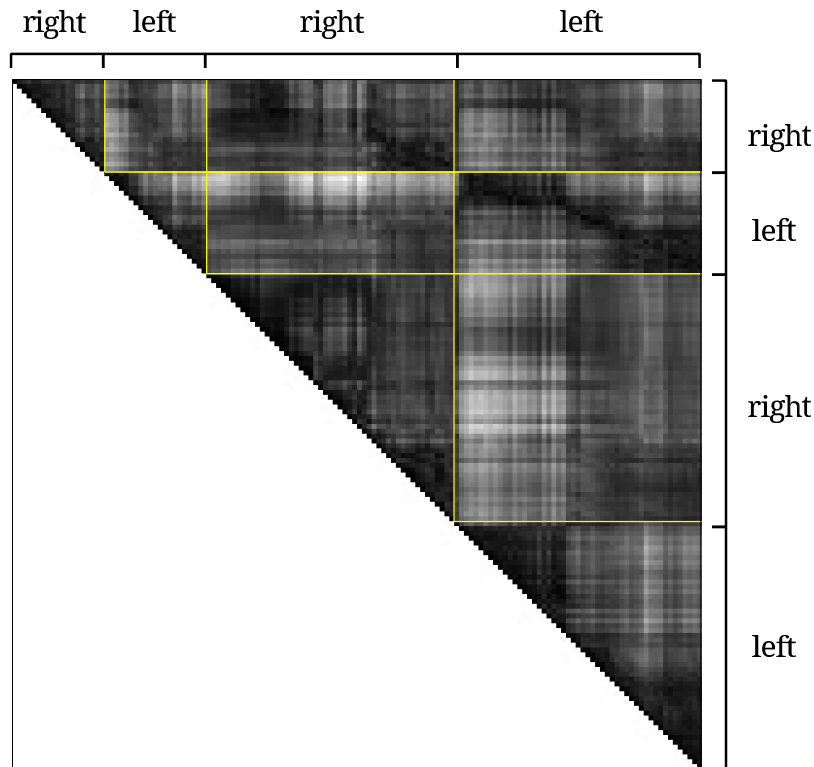


Figure 4.16: The confusion matrix when comparing differences (SAD) between all the images of the street sequence, with lighter pixels indicating larger differences. Yellow lines indicate the start of each new traverse down the street. Only half the matrix is shown because of symmetry. The SeqSLAM algorithm [Milford, 2013] searches for diagonals of low difference values in the matrix.

use of features also allows the system to deal (at least to some extent) with changes in viewpoint.

These points contrast algorithms such as SeqSLAM [Milford, 2013] which rely on temporal sequences of direct image difference values such as sum of absolute difference (SAD). Figure 4.16 illustrates the confusion matrix of the SAD values from each image in the four traverses of the same street. In the case of limited perspective change and consistent speed, repeated sequences should manifest themselves as diagonals of low difference scores in the matrix (dark pixels here). This image highlights where each traverse begins and ends with yellow lines and corresponding labels. It can be seen that very few of the blocks in the confusion matrix show strong diagonal features. The strongest diagonal appears when comparing the two traverses along the left side of the street, but even here, the diagonal does not have consistent slope, as a result of the speed variations in the second left pass. Although such algorithms suffer from these disadvantages, there are a number of alternative benefits; these include robustness to low quality and blurred images, reduced computational complexity, and no reliance on training samples.

Dataset Name	Fixed Covisibility	Clustering Approach
Begbroke	0.88	0.84
City Centre	0.38	0.32
New College	0.38	0.43
KITTI	0.61	0.62

Table 4.1: Maximum recall results at perfect precision for a version of the CovisMap algorithm which uses a fixed covisibility parameter and a version which uses a clustering algorithm rather than using parameters.

#### 4.8.4 Investigating Graph Clustering

Section 3.3.3 discussed how clustering landmarks based on the relative number of internal and external edges can be done to form virtual locations. In order to compare this method of clustering with that of expanding locations using a set covisibility parameter, precision-recall results were calculated for the four main datasets, and Table 4.1 shows the maximum recall results at perfect precision for each case. Based on these results, one can see that both methods maintain similar performance, and that compared to the other frameworks, the minor changes in recall at perfect precision is not enough to change the relative ranking of any frameworks (see Figure 4.6).

## 4.9 Outlook

This chapter has explored and evaluated several probabilistic models for visual place recognition, along with examining the merit of the covisibility framework for defining locations. A careful look into the observation likelihood model has led to a new parameterization and improved characteristics with respect to parameter sensitivity. In addition, a reformulation of the normalization model results in smoother and more intuitive posterior probabilities, sub-linear normalization complexity with respect to the size of the map, as well as adding the ability to handle multiple simultaneous loop closures that may occur due to redundant locations in existing maps. This normalization method relies on a set of sample locations to model the unexplored environment, which has been shown to be effective in our experiments, as well as experiments from related work such as Cummins and Newman [2011]. However, the applicability and creation of this sample set remains not well understood. Some prior knowledge regarding the types of expected environments is required to efficiently create and select the extent of a sample set, as computation scales with the number of sample locations. At the same time, such prior knowledge is feasible in most scenarios, and sample sets have been observed to perform well across a variety of datasets in practice. In addition, some

promising preliminary research has been done about using several different observation model parameters in order to cope with a variety of sources of sample locations and the types and quality of images that they are represented by.

In addition to these investigations into probabilistic models, the framework relies on the underlying covisibility structure proposed by Mei et al. [2010], which has also been examined here. This chapter has demonstrated the influence and benefits of using the covisibility graph in defining and extracting locations, by increasing the repeatability and relative context associated with a place in comparison to single-image location models, and increased trajectory invariance in comparison to sequential image sets. Nonetheless, these important structural cues from the covisibility graph are still being ignored in the probabilistic place recognition model which assumes conditional independence between word observations. As a result, one intriguing question to address, is how to incorporate more information from the covisibility graph into the observation model, which is the topic of the next chapter.



## Chapter 5

# Modelling Locations with Graphs of Words

### 5.1 Introduction

As seen in the previous chapter, a major difficulty of appearance-based place recognition techniques is maintaining robustness to perceptual aliasing. Approaches discussed thus far typically vary between local feature-based representations (such as comparison of sets of image interest points) and global image representations (such as comparison of image pixel intensities). Full feature-based comparisons can be computationally expensive, and therefore most of the underlying structure and geometry between features is generally ignored, such as in the FAB-MAP framework [Cummins and Newman, 2008] and the previously discussed methods presented in Chapter 4, therefore diminishing the distinctiveness of observations. As a result, such methods are prone to either false positive data associations or reduced recall, due to less discriminative observations. In the method of Chapter 4, a significant reduction in false positives due to perceptual aliasing is achieved by modelling the unknown locations with samples and including these unknown locations in the normalization of the posterior probability  $P(\mathcal{L}|\mathcal{Z})$ . In addition, many feature-based systems maintain storage of feature position information and rely on an extra ad hoc post-processing step on all returned location matches, by a geometric verification of the epipolar geometry. On the other hand, methods such as SeqSLAM [Milford, 2013] which use on global image representations, lack invariance and rely on long sequences of images in order to escape perceptual aliasing.

This chapter examines structured comparison of locations which are represented as graphs of visual landmarks from the underlying covisibility map of Chapter 3, under the assumption



that the relative arrangement of features is likely to play a key role in distinguishing places. This is in contrast to the previously presented bag-of-words models, where although locations are queried as graphs from the covisibility map, the inference is done using only unstructured sets of visual words. The reason behind such simplification techniques are to ease both the modelling task and the computation. In this chapter we aim to explore potential gains from considering the full graph structure of each location.

Graph theory is currently an active research area, with many applications ranging from bioinformatics, to network theory, to computer vision. We therefore look into the state-of-the-art in graph matching methods, and how they can be applied to visual place recognition. In general, solutions to the graph matching problem have only recently become feasible on large graphs (more than 100 nodes), and still remain difficult on dense graphs (containing many edges), graphs with more than several hundreds of nodes, or graphs with large label sets. As a result, some methods discussed here may not be able to run at a reasonable complexity for use in online navigation tasks yet, but we believe they remain as interesting discussion points for potential future application. The main contributions of this chapter are insights into graph kernels for visual place recognition, as well as a proposal and evaluation of an approach working with a simplified graph representation for efficient location comparisons.

## 5.2 On the Complexity of Graph Comparison

The introduction of visual bag-of-words techniques has allowed for efficient search and retrieval from vast amounts of images of scenes [Sivic and Zisserman, 2003]. However, bag-of-words representations lack distinctiveness, especially in the case of more general representations of locations no longer constrained to single images, where the structure of the location is completely lost. By working with covisibility maps, locations are directly represented as graphs of landmarks, and this therefore leads to the question of whether or not these location graphs can be compared in a more strict manner under computational constraints.

This is a difficult problem to approach, since in the general case, the graph matching of undirected graphs is NP-hard [Borgwardt, 2007]. Finding node and edge correspondence is a combinatorial problem which grows quickly with the number of nodes in the graph. Therefore, in order to simplify the task and incorporate error tolerance, one of many inexact graph matching approaches is typically used.

One traditional method is graph edit distance, which tries to compute the minimum cost based on edit operations (deletion, insertion, substitution) between two graphs [Bunke and Riesen,

2012]. However, edit distances rely on heuristic cost functions, and finding the minimal edit distance is still an NP-hard problem [Borgwardt, 2007]. Another, more efficient method, is to work with the graph spectra, rather than the graph itself, by decomposing a graph into the eigenvectors of the graph laplacian [von Luxburg, 2007]. However, spectral methods have trouble coping with structural noise because the eigendecomposition is sensitive to missing and spurious nodes [Bunke and Riesen, 2012]. More recently, graph kernels have also become a common method for comparing graphs, as less complex methods of computing graph kernels have been developed and graph structures are becoming more and more common in a variety of fields. Among the most common graph kernels are random walk kernels which have been shown to be able to be computed on the order of the number of nodes cubed  $O(n^3)$  [Borgwardt, 2007], and have been implemented for applications related to computer vision [Harchaoui and Bach, 2007] and scene characteristics [Fisher et al., 2011] (however with relatively small graphs of fewer than 100 nodes). Other types of graph kernels exist (both more and less complex), but kernels in the literature are limited by a complexity of at least on the order of the number of edges in the graph,  $O(E)$  (bounded by the number of nodes squared,  $O(n^2)$ ).

In order to give a better understanding of how this applies to the nature of our problem, we will now provide more insight into the size and structure of location graphs occurring in some of the relevant datasets.

### 5.3 Location Graphs of Landmarks

In this covisibility framework introduced in Chapter 3, nodes of the location graphs represent visual landmarks, while edges represent the co-observation of two landmarks, weighted by how often they are seen in the same observation. As this chapter works more closely with the graph structure, some general properties of location graphs are discussed here.

Dataset Name	<b>Begroke</b>	<b>City Centre</b>	<b>KITTI</b>
avg. # nodes	601	181	590
max. # nodes	1003	814	1274
avg. # edges (per graph)	47046	11075	51304
max. # edges (per graph)	90240	146248	160702
avg. # edges (per node)	75	47	84
max. # edges (per node)	112	179	138
avg. # degrees (per node)	105	93	119
max. # degrees (per node)	879	629	1149

Table 5.1: Location graph properties for three datasets.

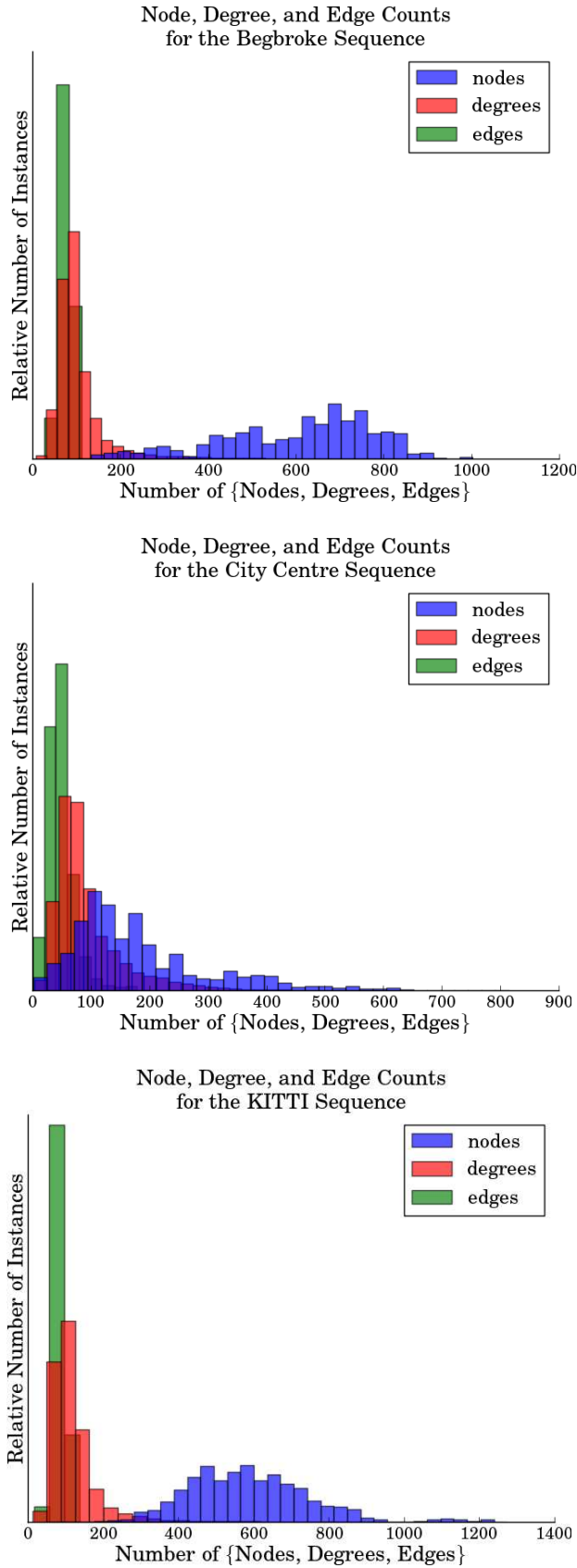


Figure 5.1: Summary of location graph properties for three datasets, showing the number of nodes, number of edges (normalized by the number of nodes), and the degree per node.

Figure 5.1, together with Table 5.1, illustrate the typical size and structural properties of graphs from virtual locations appearing in three datasets (the Begbroke sequence, the City Centre sequence, and the KITTI sequence). These figures provide intuition about the typical numbers of nodes, edges, and degrees (number of edges incident on a node). Here degree distributions are given per node, while edge distributions are given per graph, and also normalized by the number of nodes in each graph. From these figures, one can see that the average size of each virtual location is on the order of hundreds of nodes for all three datasets. Furthermore, the location graphs are densely structured, with each node being connected to roughly 100 nodes on average. In addition, the size of the label set associated to nodes in the graph corresponds to the size of the visual vocabulary used, in our case 10987 words. The size and structure of these graphs will be an important factor when considering the methods of analysis which can be applied. We can also see that even though the City Centre dataset contains much fewer nodes on average, the number of edges per node is almost as high as the other datasets. This is due to the fact that the dataset was created with the intention of having independent images (with larger distance between each image), resulting in locations consisting of small and dense cliques of only a few images each, and therefore resulting in a relatively unstructured location graphs. This sequence therefore represents a very challenging dataset for graph theoretical analysis.

## 5.4 Graph Kernels

### 5.4.1 Primer on Graph Kernels

This section introduces the basics of a few different graph kernels, based on several references which are suggested reading for more details.

The motivation behind using graph kernels for location comparison is twofold: it can provide a means of comparing the structure of location graphs in a fairly efficient way, and furthermore opens the door for more analysis of locations by allowing direct use of the graph information in most existing machine learning and pattern recognition algorithms [Borgwardt, 2007].

A graph kernel function  $k(G, G') = \langle \phi(G), \phi(G') \rangle$  defined between two graphs,  $G$  and  $G'$ , effectively maps the graphs into a linear feature space, and can act as a similarity measure. Kernels can be defined in a number of different ways, and kernel choice is often important for achieving useful results, as it acts as an information bottleneck. In kernel selection, prior knowledge about data types and domain patterns is valuable. The work of Borgwardt [2007]

highlights the characteristics required of a kernel for practical application:

- the kernel should be able to capture a non-trivial measure of similarity between graphs, expressively capturing the discriminative information of graph topology and labels,
- the kernel function must be symmetric  $k(G, G') = k(G', G)$ ,
- the kernel function should be positive definite in order to be applicable in methods which require convex optimization (such as support vector machines),
- the kernel should be computationally efficient enough to use in practice,

These concepts will additionally apply to our discussion of graph kernels.

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. Again, comparing all subgraphs between two graphs is an NP-hard problem, and therefore the types of subgraphs considered are generally limited. Examples of this include random walks, random paths, and graphlet kernels (typically enumerating subgraphs of three to five nodes). These few examples of graph kernels will now be described in more detail, in order to provide a more thorough understanding of what is often done in the field.

**Walk and Path Kernels:** Perhaps the most frequently used graph kernel in the literature is the random walk kernel. Essentially, these kernels perform walks through the graphs, where the first step can be sampled using a probability distribution over nodes, node transitions can be defined through probabilities based on edge weights, and the length of walks can be determined by one of several stopping functions which increase the likelihood of ending the walk after each transition. By then comparing walks as their sequence of node labels and/or edge labels, the graph kernels can be established (such as by simply counting how many times each walk exists in each graph being compared). The complexity of calculating random walk kernels is on the order of  $O(n^6)$ , but can be sped up to  $O(n^3)$  (by defining the problem using Kroeneker products), where  $n$  is the number of nodes. Additional speed-ups can be achieved by limiting the length of walks. [Borgwardt, 2007]

Some common bottlenecks to applying walk kernels in practice include computation speed on larger graphs, and something known as tottering, where walks repeatedly transition back and forth between the same nodes. As a result, kernels are often designed on paths, rather than walks, meaning that repetition of nodes is not allowed and transitions can only be made

to each node once. Of course, computing paths is more difficult than walks, and therefore kernels do not compute the entire set of paths in a graph, but are rather based on shortest paths between nodes. Even still, the complexity of such graph kernels are currently on the order of at least  $O(n^4)$ . [Borgwardt and Kriegel, 2005, Vishwanathan et al., 2010]

**Graphlet Kernels:** As mentioned before, graphlet kernels examine the existence of small subgraphs of a particular size (e.g., three, four, and five nodes). In order to remain computationally efficient, a fixed number of graphlets are sampled from the graph, rather than enumerating every such subgraph. Computing graphlet kernels using these sampling techniques is on the order of  $O(nd^{k-1})$ , where  $d$  is the largest degree in the graph (number of edges coming and going from a node), and  $k$  is the size of the graphlets. Based on this, it is clear that this particular kernel is well suited to graphs where the number of nodes is much less than the degree for each graph,  $n \ll d$ . [Shervashidze et al., 2009]

**Weisfeiler-Lehman Kernels:** Another trick to capture more structural information of a graph without needing to explicitly examine all subgraphs, is to include information about the neighbourhood of a node within its label. This is used in Weisfeiler-Lehman (WL) kernels, where node labels are updated to include the labels of their neighbours in an iterative scheme. The concept is illustrated in Figure 5.2, which shows the result of one iteration. By augmenting node labels in this way, the WL kernel can achieve practical results by simply comparing sets of node labels (similar to the bag-of-words vectors of Chapter 4), and therefore scales only linearly in the number of edges in the graph, thus making it much more efficient than the previously mentioned graph kernels for graphs with many nodes. [Shervashidze et al., 2011]

### 5.4.2 Applications to Location Graphs

**Walk and Path Kernels:** Random walk kernels have been previously applied to several visual recognition tasks ([Harchaoui and Bach, 2007, Bach, 2008]), however using much smaller graphs with much lower degrees of connectivity. Considering location graphs of sometimes more than 1000 nodes, one must consider the  $O(n^3)$  complexity becoming difficult to cope with, even more so for path kernels of  $O(n^4)$ . In addition, the high degree of location graphs means that sampling matching random walks becomes more difficult, as well as the likelihood of tottering and repeated cycles. Finally, high levels of noise in the node labels from the vocabulary clustering additionally make comparing label sequences from walks more chal-

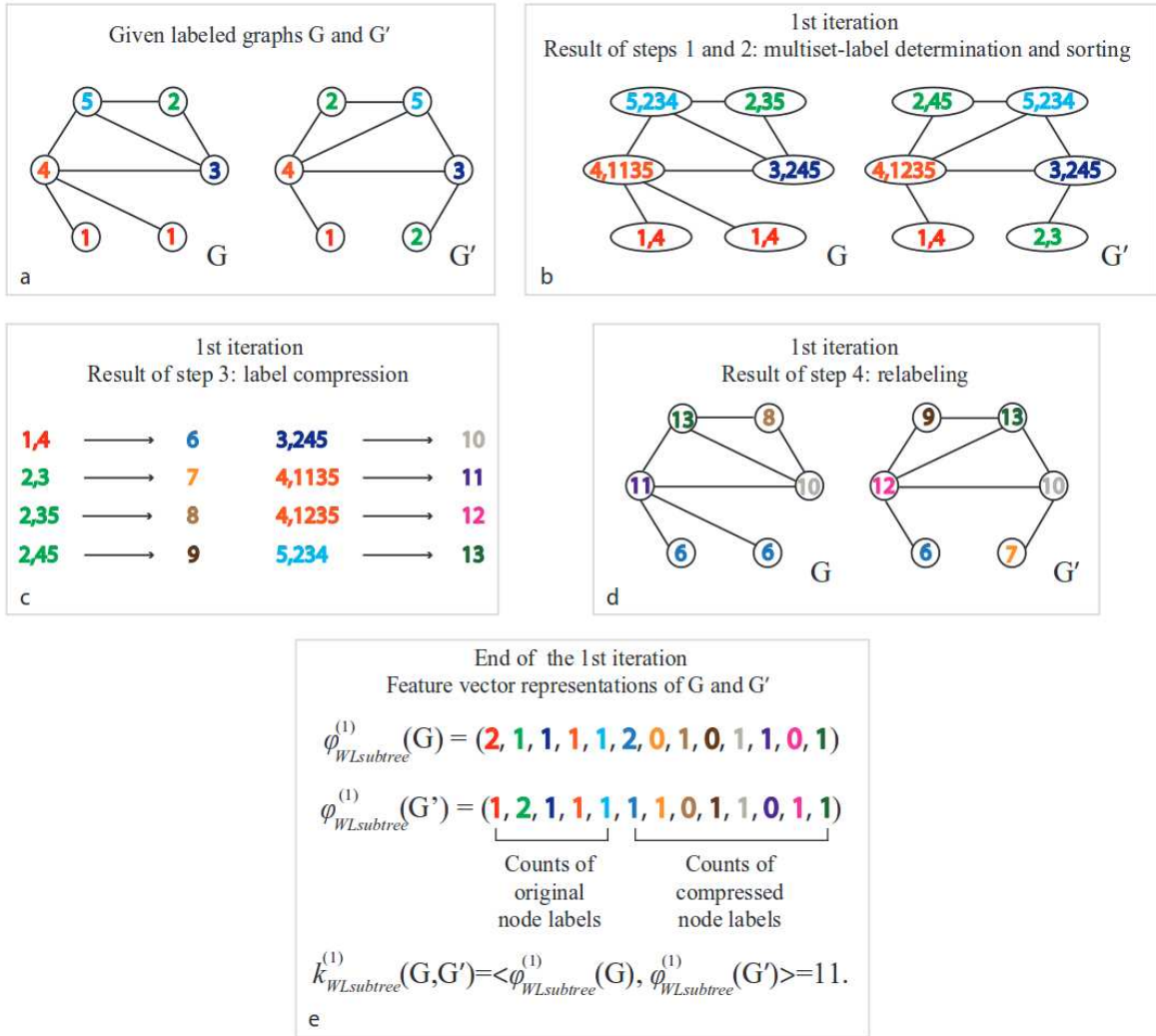


Figure 5.2: Illustration of the vector representation used in Weisfeiler-Lehman kernels, before and after one iteration of the node relabelling procedure. [Shervashidze et al., 2011]

lenging. However, by constraining the length of the random walks, the random walk kernel can be made more applicable for the task of visual place recognition, as will be discussed in Section 5.5.5.

**Graphlet Kernels:** Given the high-connectivity of location graphs, graphlet kernels do not lend themselves well to the task of place recognition. The reason is twofold: firstly because of the high complexity of the approach which scales by several times the degree of the graph,  $d^{k-1}$ , where  $k$  is the size of the graphlet. Secondly, this kernel works on the basis of identifying the connectivity structures of a few nodes, which will not perform well in the case of highly connected graphs.

**Weisfeiler-Lehman Kernels:** Weisfeiler-Lehman kernels have the lowest complexity of the discussed kernels, making them better suited to the large and highly connected location graphs. In addition, the kernel can be defined over edges, allowing it to incorporate edge weights defined by things such as covisibility counts. However, one difficulty in applying WL kernels to location graphs is due to the large vocabulary and noisiness in visual word assignment, meaning that the node-relabeling step will result in a vast number of new labels (increasing the functional vocabulary) and a low number of matching label counts, especially as the number of iterations increases.

Taking inspiration from the discussion on graph matching and graph kernels, as well as the discussion about the structure and behaviour of the given location graphs, we aim to identify a means of efficient graph comparison for visual place recognition, which will be introduced and evaluated in the following subsections.

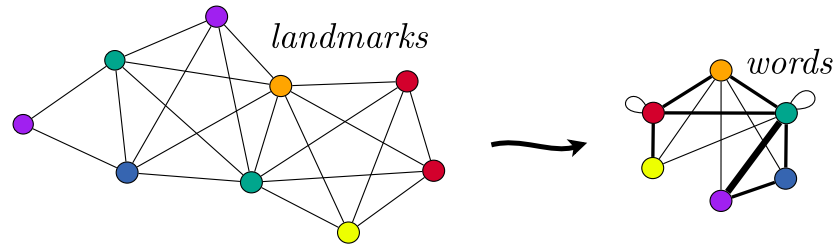
## 5.5 Location Graphs of Visual Words

Reflecting on the analysis of the previous sections 5.2 and 5.4, one realizes the trade-off between the complexity and the discriminative structure of the observations. Graph matching is an inherently difficult problem because of the fact that the so-called alignment, or node-to-node correspondences are not generally known, resulting in a combinatorial optimization to solve the task exactly. As a result, this intuitively leads to the question of how to simplify the problem by constraining node correspondences. In this section, efficient comparison is achieved by working with graphs of visual words rather than landmarks, circumventing the graph alignment problem and exploiting sparseness in the vocabulary when evaluating matches, while maintaining much more information than traditional bag-of-words techniques.

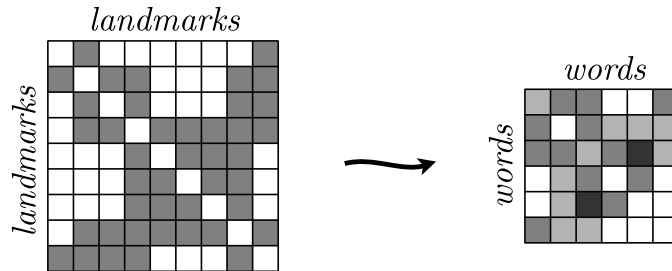
Once virtual locations are retrieved as clusters of landmarks from a covisibility map, the locations are represented as a simplified graph of visual words (rather than landmarks), where nodes are connected based on covisibility. Sparse adjacency matrices then provide a representation of locations which can be easily compared word-to-word.

The complexity of the developed method is therefore on the order of the number of common word connections between two locations  $O(E)$ , which is always bounded by the number of (unique) observed words squared  $O(n^2)$ . Essentially, locations are represented by statistics on the pair-wise observations of words which are connected in the covisibility graph, rather than statistics on individual words. This leads to a novel observation likelihood





(a) A location: from a graph of landmark co-visibility to a weighted graph of words, retaining structural information while easing comparison



(b) Representing the location by its landmark adjacency matrix and word adjacency matrix

Figure 5.3: Aiming to reduce the information encoded in images while preserving the most important visual and structural cues, each location, initially considered as a co-visibility graph of visual landmarks, is converted to a weighted co-visibility graph of visual words, where nodes correspond to unique visual words and edge weight to word co-visibility count. In (5.3a), node colour represents the associated visual word, and the thickness of edges represents the relative weighting; (5.3b) shows the equivalent adjacency matrix representation, where cell shading represents the relative weighting. The word and landmark ordering used is arbitrary.

formulation which is evaluated on some of the datasets of Appendix B.

The following subsection will outline how visual-word-based location graphs are obtained from landmark-based location graphs, and then the next subsection explains how this representation can be used to calculate observation likelihoods for locations. The two subsequent subsections then provide a discussion of how the method compares to both  $tf \times idf$  and graph kernel techniques.

### 5.5.1 Reduction to Visual Word Representation

Visual word location graphs approximate landmark-based location graphs by grouping corresponding nodes labelled by the same visual words together. This means that the graph consists of nodes representing visual words from the dictionary, rather than landmarks from the map (whose labels correspond to their visual words). Working in the space of visual words, rather than landmarks, allows the algorithm to bypass the alignment problem when comparing locations, as nodes can be easily matched one-to-one.

Figure 5.3 depicts the difference between graphs of landmarks and graphs of visual words. Edges in the word graph are weighted according to the connectivity count between words in

**Algorithm 1** Conversion between landmark and word adjacency matrices

---

```

norm = 0
n = 0
for i in range(num_landmarks):
    for j in range(i):
        # store row, column, and data values in vectors
        # for efficient sparse matrix creation
        # (only fill half sym. matrix)
        if landmark_adj[i, j] != 0:
            row[n] = min(landmark_words[i], landmark_words[j])
            col[n] = max(landmark_words[i], landmark_words[j])
            data[n] = landmark_adj[i, j]
            norm += landmark_adj[i, j]
            n += 1
# create sparse matrix
# (which sums duplicate entries)
word_adj = sparse(row, col, data)
# normalize:
word_adj /= norm

```

---

the landmark graph. The corresponding word adjacency matrices are implemented as sparse matrices, and Algorithm 1 provides details about the conversion process as pseudo code.

In addition, Figure 5.4 illustrates the effect that this alternative representation in the space of visual words, rather than landmarks, has on the overall size and structure of the graphs. From these plots, one can see that there is logically a reduction in the size of the graphs, however, the number of edges typically does not change by much (20% on average), implying that the structure of the graph and information provided by the graph is not significantly modified.

### 5.5.2 Estimating Observation Likelihoods

As in the previous chapter, a probabilistic framework is used here to evaluate place recognition, and the posterior probability of being in a location given the observation, is therefore given by Bayes' rule as follows,

$$P(\mathcal{L}|\mathcal{Z}) = \frac{P(\mathcal{Z}|\mathcal{L})P(\mathcal{L})}{P(\mathcal{Z}|\mathcal{L})P(\mathcal{L}) + P(\mathcal{Z}|\bar{\mathcal{L}})P(\bar{\mathcal{L}})} \quad (5.1)$$

where  $P(\mathcal{L})$  represents the prior probability of being in a given location,  $P(\mathcal{L}|\mathcal{Z})$  represents the likelihood of an observation given a location,  $P(\mathcal{Z}|\bar{\mathcal{L}})$  represents the likelihood of the observation coming from any other location, and  $P(\bar{\mathcal{L}})$  represents the prior probability of

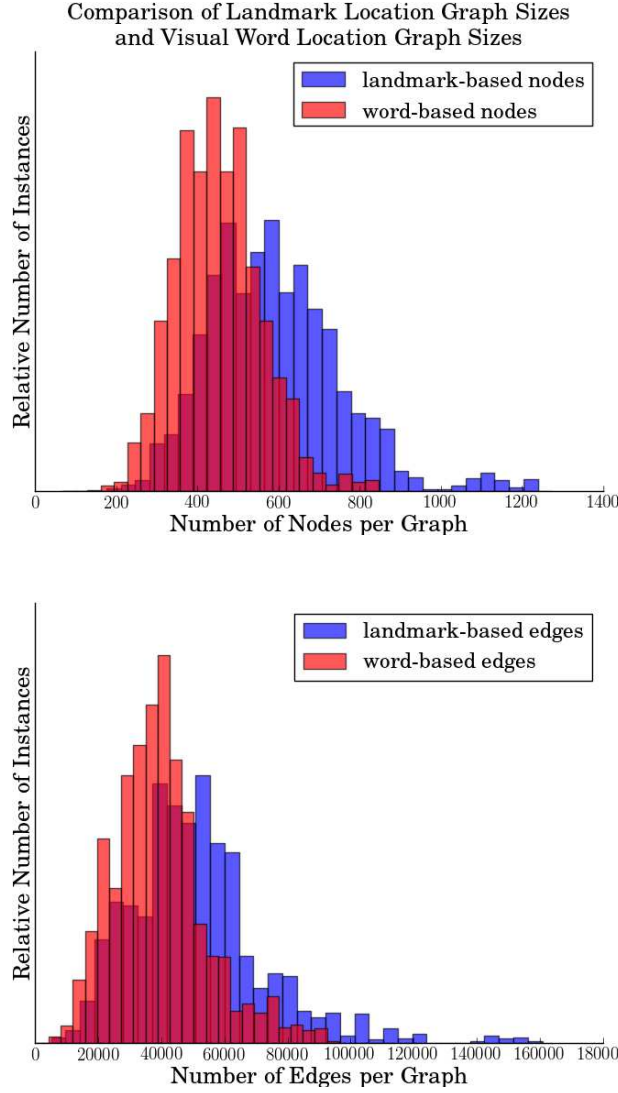


Figure 5.4: Comparison of landmark-based location graph sizes and word-based location graph sizes for the KITTI dataset. The first plot shows the number of graph nodes in each location, while the second shows the number of graph edges in each location.

being in any location other than  $\mathcal{L}$ .

Provided query and candidate virtual locations which are represented by their visual word graphs described in Section 5.5.1, likelihood values can be formulated in a relatively straightforward manner. For instance, here the normalized cross-correlation between two adjacency matrices is assumed to be proportional to the observation likelihood  $P(\mathcal{Z}|\mathcal{L})$ , as shown in Equation 5.2.

$$P(\mathcal{Z}|\mathcal{L}) = \frac{\sum_{\{\mathcal{E}\}} E_{uv}^{\mathcal{Z}} \cdot E_{uv}^{\mathcal{L}}}{\sqrt{\sum_{\{\mathcal{E}\}} (E_{uv}^{\mathcal{Z}})^2 \sum_{\{\mathcal{E}\}} (E_{uv}^{\mathcal{L}})^2}} \quad (5.2)$$

Where  $E_{uv}^{\mathcal{Z}}$  and  $E_{uv}^{\mathcal{L}}$  represent the edge weights between words  $w_u$  and  $w_v$  from the query

observation and candidate location respectively, and  $\{\mathcal{E}\}$  is the set of possible edges.

Due to the sparsity of visual words in each location, computing cross-correlation scores requires relatively few calculations, as only words common to both locations are involved in the numerator of Equation 5.2. This implies that the complexity of this computation is typically substantially less than  $O(n^2)$  as it depends on the number of common edges in the graphs. However, even though only a subset of words from each location are involved in this sum, *all* words in a location have an impact on the final score since edge weights are always normalized across the location (see Algorithm 1).

### 5.5.3 Sampling

Like in the previous chapter, the likelihood of the query coming from another location  $P(\mathcal{Z}|\bar{\mathcal{L}})$  can be calculated analogously to Equation 5.2, using a set of sample locations as follows,

$$P(\mathcal{Z}|\bar{\mathcal{L}}) = \sum_{s=1}^{N_s} \frac{P(\mathcal{Z}|\mathcal{L}_s)}{N_s} \quad (5.3)$$

with  $N_s$  being the number of samples, and  $\mathcal{L}_s$  being the  $s^{th}$  sample location. Note that the terms in the denominator of Equation 5.2 only need to be computed once for each location, which is meaningful for the query and sample locations which are typically reused in many calculations.

### 5.5.4 Relation to $tf \times idf$

As some words and edges occur more commonly than others, they provide different amounts of information about the location. This concept is well recognized in the text analysis field, and therefore terms are generally weighted according to a prior on their frequency, which is provided by known documents. As a result, more common terms tend to have less impact on the final results than rare terms. In the context of place recognition, an intuitive example is provided by comparing a brick wall to a statue. Since bricks are seen throughout cities, bricks would have a relatively low weighting, as they do not provide much context about the location, whereas the statue is unique and provides much more contextual information.

In the example of the commonly used  $tf \times idf$  scoring method, each word is given by a value proportional to the number of times it was seen in that document (*term frequency*) and inversely proportional to the number of other documents which contained the same word (*inverse document frequency*), as shown in Equation 2.5. Each document is then represented by a vector of all its  $tf \times idf$  word values, and similarity is given by the dot product between

document vectors [Sivic and Zisserman, 2003]. In this work, the word adjacency matrices can be viewed analogously to  $tf \times idf$  vectors, comparing word connectivity (edges), rather than individual words. Note that working with this word connectivity, rather than a traditional bag of words, is the important factor for retaining structural information encoded in the observations. Each word adjacency matrix is then weighted according to the relative information content  $-\ln P(E_{ij})$  provided by each edge, which is precalculated based on prior probabilities from a set of sample locations. Further study on the full probabilistic interpretation behind  $tf \times idf$  weighting can be found in [Hiemstra, 2000].

### 5.5.5 Relation to Graph Kernels

The observation likelihood formulation of Section 5.5.2 can additionally be re-interpreted in the form of a graph kernel. The work of Mohan et al. [2015], derive an analogous similarity score to be the same as a length-one random walk graph kernel on a weighted graph. This can be intuitively seen by the fact that a 1-walk kernel simply iterates through all edges in two graphs and counts how many edges start and finish with the same nodes in each graph respectively. The likelihood formulation can furthermore be viewed in a similar way to a Weisfeiler-Lehman kernel over edges, without any relabeling iterations.

These links to graph kernels interestingly open the door for further research towards other machine learning and pattern recognition tools for analyzing locations, as well as investigating and developing other types of graph kernels for such tasks. We have begun a more in-depth look into graph kernels for visual place recognition, and some preliminary insight will be provided in Section 5.7.

## 5.6 Experimental Evaluation of Visual Word Location Graphs

In this section we evaluate the visual word graph methodology of Section 5.5 empirically, and illustrate the functionality of the approach with a number of intuitive examples. For clarity, from here on the methods developed in this chapter will be referred to as *Graph CovisMap*, while those developed in the previous chapter will be referred to as *Naive-Bayes CovisMap*.

### 5.6.1 Comparison with State-of-the-Art

During testing, each dataset is incrementally traversed, building a map over time and using the most recent location as a query on the current map, with the goal of retrieving any previous instances of the query location from the map. Precision-recall results for the three

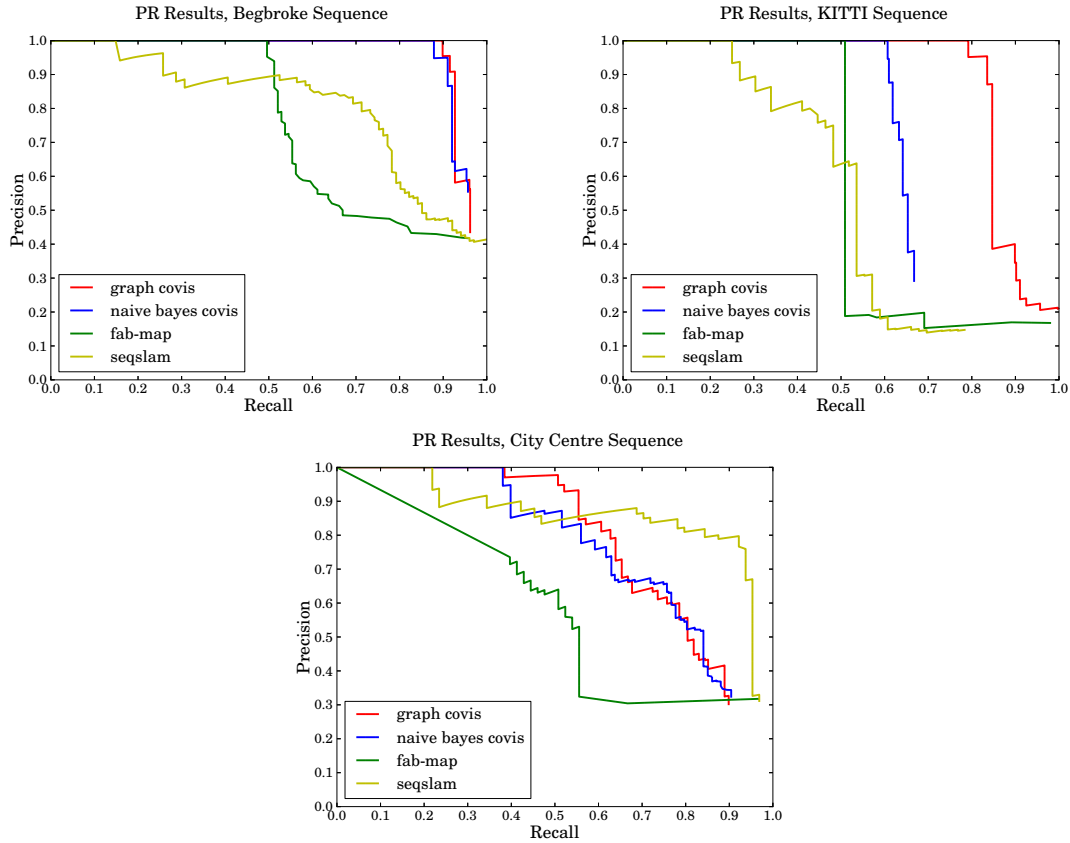


Figure 5.5: Precision-recall results for the Graph Covis framework presented in this chapter, the Naive-Bayes Covis method [Stumm et al., 2015a], and the FAB-MAP method [Cummins and Newman, 2008], on three different datasets.

datasets are shown in Figure 5.5, comparing the method described in this chapter (Graph CovisMap), to that of Chapter 4 (Naive-Bayes CovisMap) which uses covisibility for virtual location extraction but discards structure during comparison, and that of FAB-MAP which works with single-image locations and no location graphs [Cummins and Newman, 2008]. Note that the results from the FAB-MAP framework are included for completeness here, but that the probability normalization model differs compared to the other tested frameworks, reducing recall significantly in the presence of multiple instances of the same location in the map (as explained in Chapter 4 or [Cummins and Newman, 2008]). In this work, positive loop-closures are given by locations, which contain landmarks from within a given radius of the query location. The radius used for evaluation was set to  $8m$ , as errors in ground truth labels can reach several meters, and image spacing is frequently as far as  $2m$  apart. Each framework was provided with the same set of sample locations, which consist of images from Streetview locations and other datasets (excluding the tested dataset). In addition, as is typically done during testing, no data associations were made based on loop closures [Cummins and Newman, 2008, Stumm et al., 2015a].

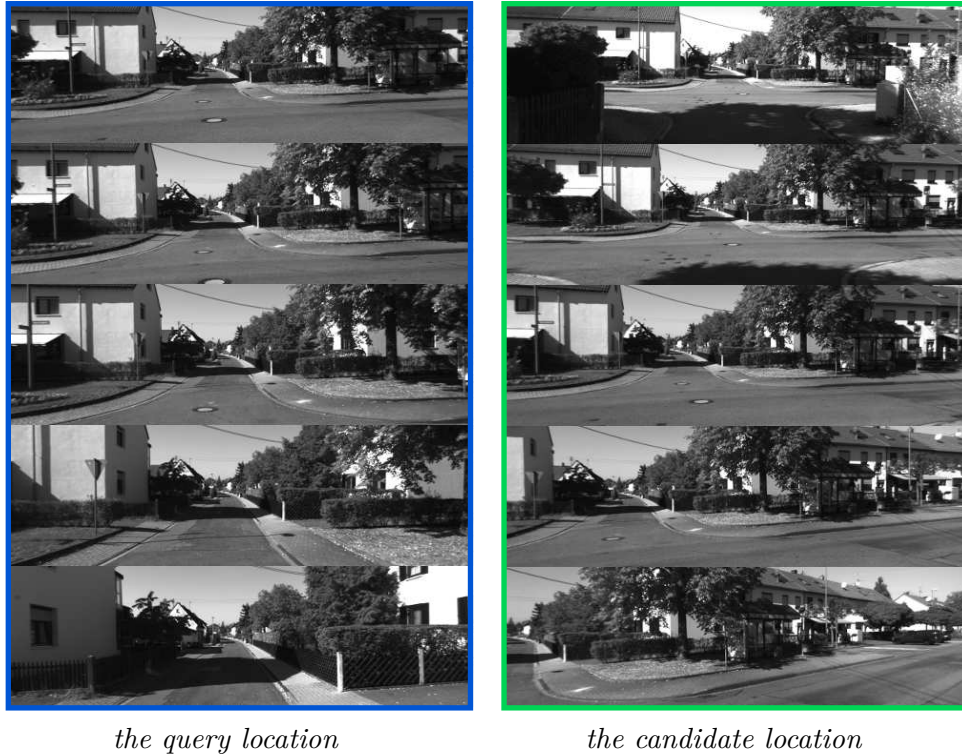


Figure 5.6: A query and a candidate location from the KITTI dataset shown by five representative images. The two locations are a true match, obtained from the same intersection; the query passing straight through and the candidate location turning right. Our graph-based covisibility framework assigns a match probability of 0.91, while the unstructured framework assigns a probability of 0.02.

From Figure 5.5, one can see a general improvement from utilizing the structural information for both location extraction and comparison. Improvements are minor compared to the Naive-Bayes CovisMap framework on the Begbroke sequence, since the recall is near perfect already. Results are more significant in the other two datasets, where the Graph CovisMap framework provides a more significant boost in recall rates. Note that performance is lower on the City Centre dataset, as images tend to contain less overlap in features, reducing the quality of the covisibility map built from them, and limiting the improvements from the proposed method. In addition, the City Centre sequence generally contains more variations and ground truth errors than the other two datasets.

A representative example of the improved recall of our method can also be seen in Figure 5.6, where a query and candidate location are compared using both the structured Graph CovisMap and unstructured Naive-Bayes CovisMap frameworks. The two locations represent the same intersection, only traversed in different ways. This difference in traversal introduces enough differences to the word sets of each location for the unstructured method to assign a low match probability, while word connectivity remains consistent enough for the structured

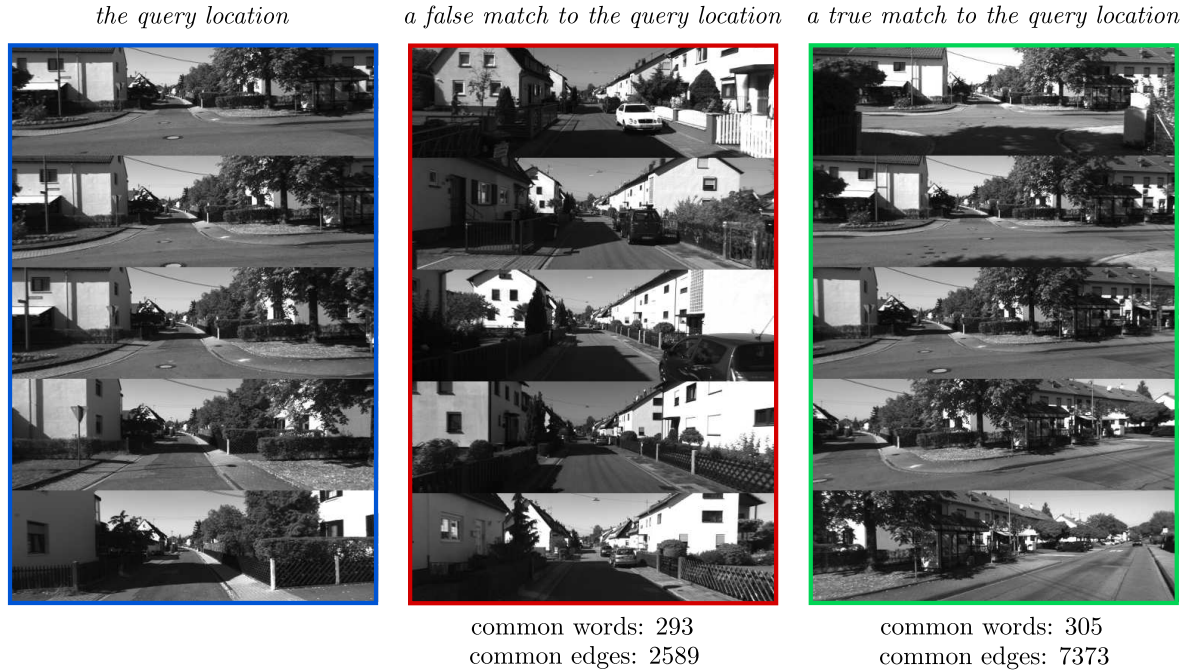


Figure 5.7: A query and two candidate locations from the KITTI dataset, shown by five representative images. The two candidate locations demonstrate the importance of structural information in the observation, as both locations share approximately 300 visual words with the query location, the true match shares more than twice as many edges with the query location.

method to provide a relatively high probability.

Looking more closely at this intersection, one can see how the observation is affected when including structure in the model. Figure 5.7 shows the same query location, with both a false match and a true match. Although both candidate locations share approximately 300 visual words with the query location, the false candidate shares only 2589 edges with the query while the true candidate shares 7373 edges with the query.

### 5.6.2 Investigating Weighting Schemes

The importance of edge weighting is clearly illustrated in Figure 5.8, which shows a query and two candidate locations (one matching and one false), along with the posterior match probabilities provided from using both unweighted and weighted word adjacency matrices as described in Section 5.5.4. In this case, the unweighted probabilities remain fairly high for both locations, as they have a similar appearance and share a similar number of common edges with the query. When weighting the word adjacency matrices based on the edge frequencies in sample locations, however, the importance of frequently occurring edges is down-weighted, reducing the probability of the false location.





*the query location*



common edges: 4390, unweighted prob: 0.91, weighted prob: 0.9  
*a true match to the query location*



common edges: 3451, unweighted prob: 0.83, weighted prob: 0.66  
*a false match to the query location*

Figure 5.8: Example of a query and two retrieved locations from the KITTI dataset. Each location is depicted by a single image central to the location. In this case, both of the retrieved location graphs share many common edges with the query. As a result, the match probabilities when using unweighted term frequencies are high for both locations. However, when using term frequencies, which are weighted by the relative document frequencies from sample locations, the match probability of the false candidate drops significantly.

### 5.6.3 Investigating Behaviour with Respect to Noise

Furthermore, in order to investigate the robustness of the location graph models, noise is incrementally added to locations and the behaviour is shown in Figure 5.9a. Taking independent locations from the KITTI dataset and adding varying amounts of noise, the noisy version is compared to the original location, plotting the resulting boxplots of the posterior match probabilities. Noisy locations are created by corrupting a certain percentage of the words associated to the location's landmarks, randomly swapping them with another word

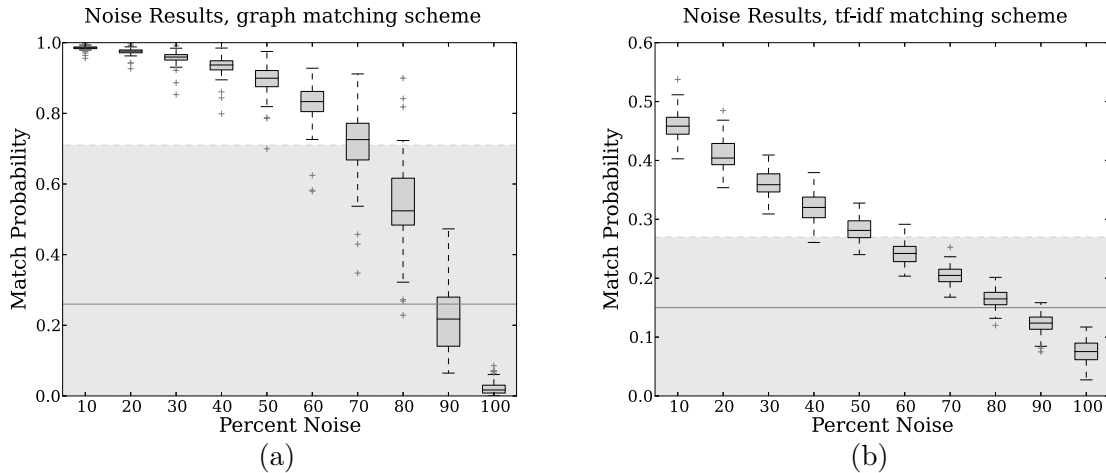


Figure 5.9: Statistics for location similarity as the visual words of locations are corrupted by noise and compared to their original state. The grey shaded areas show the scores where the first false-positive from other retrieved locations occur, and the black lines show the median scores for these false-positives.

from the dictionary. This process implicitly alters the edge structure of the corresponding word adjacency matrices. Figure 5.9a also shows the highest posterior match probability achieved by a false loop-closure from the same dataset with grey shading, indicating the line above which perfect precision would be maintained. Figure 5.9b shows the results of the same experiment run using a traditional  $tf \times idf$  comparison method, which uses words with no added edge structure. Note that the probabilities are significantly lower in this plot because locations are less discriminative under a structureless model, reducing scores normalized using sample locations as in Equation 4.1. Together, these plots illustrate the benefit of structured comparison. While the boxplot whiskers remain above the false-positive threshold up to 60% of added noise when using structured comparisons, this is only the case up to 30% of added noise when using unstructured comparisons.

## 5.7 Outlook

In the analysis of structured comparison of locations extracted as graphs from the covisibility map, we have examined several different approaches ranging in complexity. The work of Chapter 4 looks at highly simplified graphs in the form of a bag-of-words models which ignore all the information in the graph edges, while this chapter attempts to incorporate varying levels of structural information from edges, settling on visual-word based location graphs as a representation which allows for a compromise between the complexity of high-level graph comparison techniques and bag-of-words techniques. Empirical analysis suggests that such a representation captures the available information well, while being robust to noise present

in the observations. In fact, preliminary tests with several more complex graph kernels have been conducted, and these indicate a degradation in performance due to their lack of ability to handle observation noise, especially in the case of mislabeled nodes (a common problem due to poor clustering in the definition of the visual dictionary). In general, selecting and applying graph kernels to problems such as visual place recognition is not an easy task. Each type of graph kernel captures different aspects of the graph structures and properties, and knowing which ones efficiently pick out the discriminative nature of graphs for a particular application requires significant insight.

Taking into account the expected nature of observations in the form of covisibility graphs, we have implemented another type of graph kernel, which we will refer to here as the neighbourhood kernel. This kernel is similar to a single iteration WL kernel but better suited towards graphs with noisy node labels. Effectively, as with the WL kernel, this compares the direct neighbourhood (all nodes within a distance of one edge), but as a sparse vector

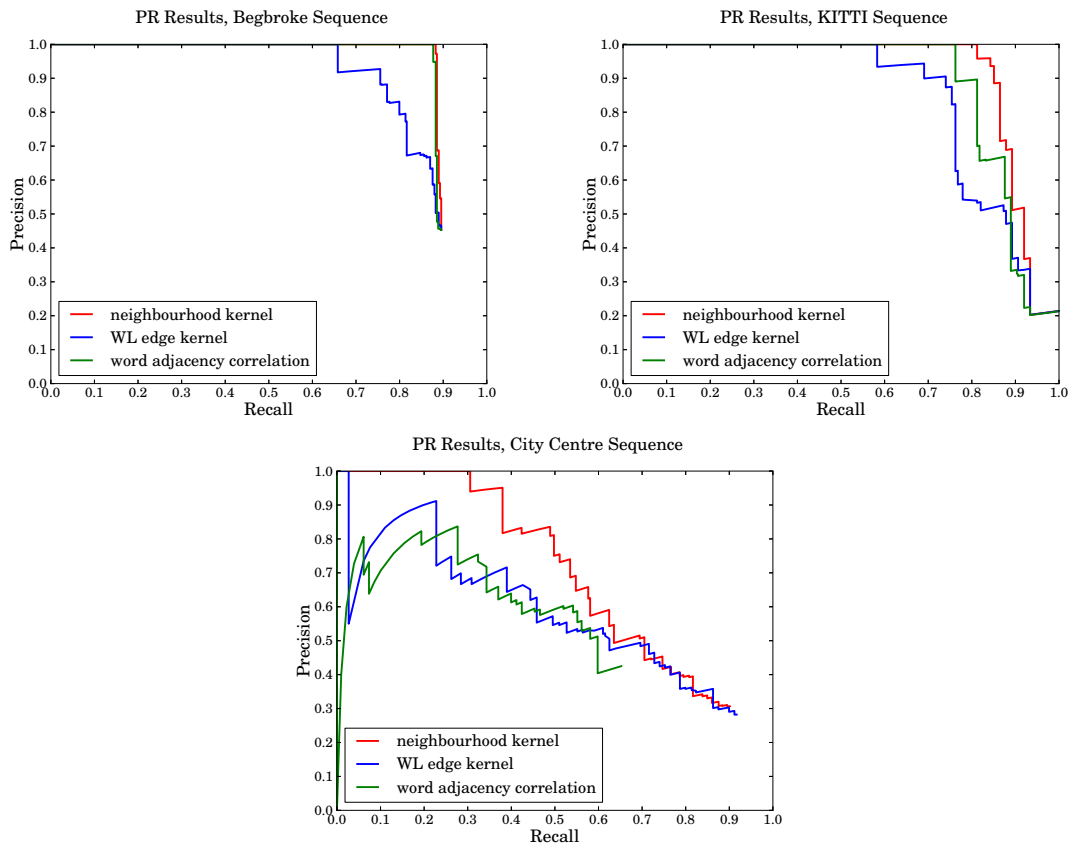


Figure 5.10: Precision-recall results for observation likelihood values from the proposed neighbourhood kernel, the single iteration WL kernel, and the word-adacency correlation method of Section 5.5.2 on three different datasets. Note that the volatile nature of the PR curves for the City Centre sequence are due to a few high scoring false-positives which lead to low precision until the number of true positives balance them out as the threshold is lowered.

of the length of the vocabulary rather than by relabeling the root node with the neighbourhood label sequence. In order to remain efficient, this neighbourhood is only compared for root nodes with the same label (taking the best match in the case of multiple nodes with the same label), and therefore depends on the set of words observed in both locations and the degree of each node (number of edges associated to a node). Scores are summed across the set of observed words. The metric is symmetric and results in a positive-definite kernel matrix. The resulting complexity 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. Note that this complexity is only slightly higher than the method of Section 5.5.2, as edges are essentially double counted when working with the degree of nodes.

Figure 5.10 shows the results of some preliminary tests which compare the neighbourhood kernel to a single iteration WL kernel and the word-adjacency correlation metric of Section 5.5.2. These plots show the precision-recall results when thresholding the likelihood score of each method, and demonstrate that the neighbourhood kernel is able to produce good results without normalization. These results are somewhat intuitive, as the neighbourhood kernel utilizes information-rich and discriminative observations. However, as a result of the same phenomenon, precision-recall results actually decrease after normalization using sample locations which are no longer able to simulate the unknown world well. Another observed result of working with the neighbourhood kernel, is that it tends to assign high scores within a tighter metric radius than the other methods, meaning that it could be better suited for applications of loop-closure or metric localization.

From this analysis, the trade-off between incorporating more or less structural information for more discriminate representations of locations is better understood, but still requires clarification through further investigation. On one hand, methods such as the neighbourhood kernel have the potential advantage of bypassing the expensive normalization step completely, but at the loss of the easy probabilistic interpretation that comes with properly normalized posterior probabilities.



## Chapter 6

# Closing Remarks

### 6.1 Conclusion

This thesis has studied the task of appearance-based place recognition; primarily investigating various representations of places and the corresponding probability models based on visual observations. In order to group landmarks together in a relevant way, a covisibility graph is created, and clusters within the graph represent locations. This method is shown to be inherently able to cope with variations in robot trajectories, including irregular changes in speed, direction, and viewpoint. A detailed analysis of probabilistic observation models was used to improve robustness to errors and parameter sensitivity. The resulting generative model provides the posterior probability of being in a certain location given a particular observation, in a way which does not require normalizing over the entire map and is able to find multiple instances of a location. The framework is useful for loop-closure detection, recovery from the kidnapped robot problem, map fusion, and topological mapping. Therefore the methodology is appropriate for applications where the robot travels in unconstrained environments, or when using unconventional modes of travel such as elevators or trains, where sensing egomotion becomes very difficult (equivalent to the kidnapped robot problem).

In addition, in order to better take advantage of the graph-based covisibility representation of locations, efficient approaches of graph matching have been explored for comparing locations. Notably, this thesis has developed an observation likelihood formulation which exploits the covisibility of landmarks to account for geometric structure on top of appearance, and is demonstrated to outperform state of the art methods in place recognition on several datasets. While only relatively weak geometric and structural information is encoded in the covisibility graph, this is shown to be enough to disambiguate across appearance-only

matches, helping tackle perceptual aliasing, which is a common problem in existing methods. An evaluation across a variety of environments and variable presence of noise reports increased recall at perfect precision. The added complexity of incorporating geometric cues is minimized by employing efficient algorithms inspired by both graph-matching and place recognition literature.

## 6.2 Extensions and Open Questions

Reflecting on insights gained during work on this thesis, this section highlights some ideas for extensions and improvements to the developed methods, in the hope of inspiring future progress in understanding and improving location models for visual place recognition.

### Visual Vocabulary

The work in this thesis relies on high dimensional descriptor quantization via a trained visual vocabulary of roughly 10000 words. Finding a partitioning of such a high-dimensional and complex descriptor space which can produce repeatable word associations is a difficult task. However, experience often shows that starting with a well-trained vocabulary is not necessarily important for achieving practical results [Cummins, 2009]. For example, the same visual vocabulary has been used across all the datasets in this work, as well as drastically different environments such as indoor imagery along with birds-eye imagery from a flying vehicle. At the same time, a poor partitioning of the descriptor space is perhaps one of the largest sources of noise in the system, due to false data associations in the word labelling, which is consistently seen throughout all experiments. One important question is understanding the extent to which this affects the results, and how to improve behaviour with respect to poor quantization.

Improvements can be made on both sides; to the visual word association, as well as to how observation likelihoods are modelled. Some methods of improving vocabulary creation and word assignments have already been investigated in work such as [Jegou et al., 2008] where hamming distance is used to refine the quantization of features, [Angeli et al., 2008b] where the dictionary is maintained incrementally, and [Cummins and Newman, 2011] where the size of the vocabulary is simply increased to improve results. Methods have been introduced which allow for fast search and retrieval with large amounts of feature vectors, such as work on efficient nearest neighbour search by Jegou et al. [2011] and inverted multi-index of Babenko and Lempitsky [2015], which could be used to increase the size and accuracy of

the feature associations. In addition, the choice of the underlying local feature detector and descriptor is important for repeatability and ease of clustering, of which many alternatives exist [Calonder et al., 2010, Leutenegger et al., 2011, Zitnick and Ramnath, 2011, Alahi et al., 2012], as well as combining two or more in the same framework [Sivic and Zisserman, 2003, Angeli et al., 2008b].

On the modelling side, one can imagine adjusting observation likelihoods and detection probabilities to cope with errors in the quantization. This could arguably be more useful than improvements to the descriptor quantization, as the same vocabulary is typically used within a variety of environments, therefore motivating robustness to vocabulary choice. As an example, in the model of Equation 4.4, word existence probabilities can vary from word to word, based on clustering properties. Additionally, relationships between visual words can be incorporated, based on neighbouring clusters in the descriptor space (similar to [Philbin et al., 2008]), although at the cost of decreases in sparseness.

Moreover, due to the general nature of the approach, the vocabulary could likewise be replaced by higher level features, or even features provided by sensors other than standard cameras. For example, a vocabulary of objects or contextual features may prove to be more robust in certain scenarios, especially for applications where semantic clues might be useful. The works of Singh et al. [2012] and Doersch et al. [2012] learn mid-level features which are representative of certain places or things. As another example, useful features could be learnt using techniques such as convolutional neural networks and training data [Oquab et al., 2014, Chen et al., 2014, Sünderhauf et al., 2015b]. In the same way, certain environments may be better suited towards different sensing modalities, in which case the visual vocabulary could be swapped with, or used alongside, one from another sensor (e.g. using point clouds from a lidar or RGBD camera [Bo et al., 2011, 2013]).

### **Normalization by Sample Locations**

Similar to the visual vocabulary, choosing a set of sample locations for normalization of probabilities is an uncertain task. In order to avoid perceptual aliasing and also estimate word observation likelihoods, samples are used to model the frequency of small and large scale features of the expected environment. However, in many applications, the true nature of the environment is not known ahead of time, and if it is not well approximated by the sample locations then false-positive data associations can easily occur. As a result, more work should be done on understanding these choices, and how to evaluate the extent of the samples.



Some possible extensions have already been partially explored, in order to make better use of the given set of sample locations. This includes increasing the efficiency associated to using the sample set, such that the size of the sample set can be increased without reaching computational limitations. This can be done by retrieving relevant samples from a more complete sample set (similar to virtual location retrieval) and only using that subset explicitly in the calculations. When increasing the size of the sample set, a variety of different types of imagery often arise (from different cameras, for example), which can be compensated by different observation and detection probabilities corresponding to each type of imagery. Further extensions also include evaluating and seeking out an improved sample set over time, similar to work by Paul and Newman [2013]. Of course, there is still no fundamental guarantee that false positives will not occur. For example, in the case of changing environments, new and repetitive features could be suddenly introduced, causing perceptual aliasing. Understanding these limitations, the system should strive to be robust to as many modes of false positives as possible. However, methods of dealing with the false positives that do occur should also be explored, such as [Latif et al., 2013, Sünderhauf and Protzel, 2012].

### **Additional Sensory Information**

Throughout this work, little emphasis has been put on the role of the location prior. This is to some extent because it has been shown to have relatively little impact on the results [Cummins and Newman, 2008], and also because this work was intended to remain independent to types of motion, with little or no assumptions about how the robot is expected to move. Commonly, place recognition systems update location priors based on the current position estimate and an assumed motion model, such as by increasing the probability of nearby locations in the map. However, in order to refine results, a number of different contextual cues could be used to better estimate a location prior. Information from complementary sources of visual information could be used, such as global image features, including colour histograms [Angeli et al., 2008b], scene texture [Torralba et al., 2003], gist descriptors [Oliva and Torralba, 2006], or reduced resolution pixel intensities [Milford, 2013]. Taken further, semantic reasoning on scenes (such as [Pronobis et al., 2006, Quattoni and Torralba, 2009]) can be used to evaluate which types of places are typically correlated and adjust the priors accordingly.

Moreover, observation models could be updated to incorporate an array of information from other sensors such as inertial measurement units, GPS, thermal imagery, WiFi networks, existing map data, etc. Some interesting related work is given in [Newman et al., 2009,

Biswas and Veloso, 2010, Maddern et al., 2012, Brubaker et al., 2013]. This is perhaps more easily done within in a larger estimation framework, which then passes a prior probability to the place recognition system, bearing in mind that place recognition is often intended to be a relatively independent source of loop closures.

### **Graph Analysis**

The graph comparison work of Chapter 5 has led to a number of interesting questions. Preliminary work suggests that certain types of graph kernels are able to capture much more discriminative information than previously existing techniques. This gives the potential to substantially reduce false positives in environments with similar visual features. However, due to the more precise nature of observations, building a representative sample set for normalization becomes a challenge. More work should be done to see if there is an effective representation of sample locations for this task, or an alternative method of normalization should be used. In certain more contained applications, it may even be appropriate to forgo normalization and rather resort to training a discriminative model or tuning a decision threshold.

In addition, future work includes further investigation into which graph properties are well suited to place recognition. For example, recent work by Bai et al. [2014] highlights some of the problems with many existing graph kernels, and proposes alternatives based on information theoretic measures. Furthermore, work from other related fields could provide ideas about subgraph retrieval from large databases [Khan et al., 2013], incorporating non-rigid geometric constraints in graphs with iterative closest point [Zhou and De la Torre, 2013], and SLAM graph connectivity properties [Khosoussi et al., 2014]. Additionally, related to work by [Singh et al., 2012, Doersch et al., 2012], analysis could be used to identify frequently occurring and discriminative graphlets (small cliques in the covisibility graph) representing mid-level features which are informative in recognizing places.

### **Evaluation and Metrics**

An underlying difficulty in developing reliable place recognition algorithms is defining test metrics for evaluating system performance. One main reason for this is that the desired nature of the place recognition is largely application dependent. For example, if using place recognition as a means of metric localization, it is fairly easy to define true positives as falling within a tight radius of the given location. However, if the idea is to simply provide data associations between landmarks in the environment, defining such a radius is not possible,

as the system could match distant features correctly. In addition, it is generally difficult to obtain a diverse collection of datasets, with accurately labeled position information. As another example, the intended behaviour will also depend on the application. While some systems may want to maximize the recall at full precision, others may be more interested in minimizing the distance or time between loop closures. In addition, some applications may demand high levels of robustness towards unexpected, dynamic, and noisy observations, while others may be put to use in much more constrained environments. As a result, this motivates things such as model stability, parameter sensitivity, and sample selection very differently.

### **Semantic Recognition and Hierarchy**

Semantic recognition of scenes is a natural extension of the current work, making use of the proposed location models to then classify scenes and/or objects. We see the graph similarity measures and statistics on structural properties as being especially well suited towards this task. More specifically, the discussed graph kernel methods can be directly applied to existing machine learning algorithms, such as the commonly used support vector machines. This kind of semantic labelling could be used for applications such as task planning, human-robot interaction, augmented reality, and even enhance place recognition [Pronobis et al., 2006, Oliva and Torralba, 2006, Sünderhauf et al., 2015a].

Further interest lies in achieving a kind of hierarchical, or variable resolution localization, combining semantic information with covisibility information. As an example, depending on the outcome of a place recognition query, the system may be able to provide varying levels of localization. The concept of places can then vary in scope, ranging from things like being in a certain city or building, to being in a specific location in the city or building. If the system cannot decisively pick out the exact location it might still be able to provide useful higher level information. This is related to concepts from the works of Doersch et al. [2012] and Mohan et al. [2015] which can pick out which features are common to a specific environment, but not others. Using the covisibility graph, different resolutions can be extracted by looking at larger and smaller clusters of features.

### **Long-Term Autonomy**

As appearance-based place recognition has become more and more reliable over the last few years, a current trend is to work towards place recognition systems that function over larger time scales. Related challenges include how to handle vast amounts of data that is collected over time, and how to deal with drastic shifts in the appearance of a scene due to seasonal

changes and dynamic objects.

Currently, there seems to be a dichotomy in approaches to long-term place recognition: between maintaining several observations of each place under different appearances, and alternatively utilizing a model which can recognize places despite appearance changes. Along the lines of the first approach, the works of Churchill and Newman [2013], Linegar et al. [2015] extend local feature-based place recognition frameworks by recording various “experiences” of locations over time and maintaining several appearance modes, which can then be used to perform visual place recognition across each mode. Alternatively, using sequences of image pixel values as a starting point, other systems rely on the invariance of global image properties to allow for place recognition under lighting and seasonal changes [Milford, 2013, Sünderhauf et al., 2013, Arroyo et al., 2015]. Some work has also been done on predicting how image superpixels can be transformed to account for seasonal changes [Neubert et al., 2015]. Somewhat similarly, models can be used to learn which local features remain stable across changes in appearance [Johns and Yang, 2014]. In terms of efficiency, in order to deal with large maps, work by MacTavish and Barfoot [2014] and Mohan et al. [2015] rely on hierarchical techniques to efficiently conduct place recognition.

Interesting prospective work would be to look into how to extend the work from this thesis in similar ways, and possibly make use of the structural covisibility graph information for use in this task. For example, the graph-based comparisons proposed in this work have already been shown to be more robust to noise, and therefore may be well suited to place recognition with minor appearance changes. Based on the suggested future work regarding semantic recognition or vocabulary extensions, one can further imagine the covisibility graph including higher level features such as objects or characteristics which are more likely to remain constant over time. Furthermore, in terms of hierarchy, it is easy to imagine how covisibility can aid in defining a hierarchy of location size, again with potential benefits from semantic information.



# Appendix A

## Implementation

### A.1 CovisMap framework

The CovisMap framework presented in this thesis essentially processes a stream of images, testing for place recognition using the current location, and updating the covisibility map at each time step. However, in order to better understand and evaluate the performance of each method, no data association from loop-closures is actually done during testing (similarly to the tests in [Cummins and Newman, 2011]). Figure A.1 provides an overview of this process. The inputs to the system, shown by grey, square boxes, are the image sequence provided by the mobile robot, and a set of sample images as described in Section 4.7. Data structures are shown in yellow, rounded boxes, which include the covisibility graph, and all of the locations (query location, virtual locations, and sample locations). Then, the algorithmic blocks are shown by blue ovals, with relevant sections listed alongside. The main contributions of this

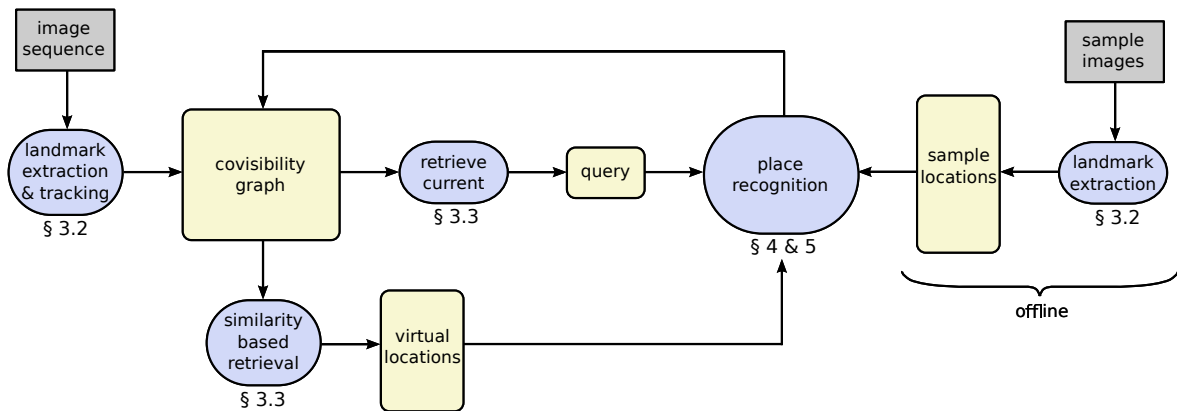


Figure A.1: General scheme of the implementation framework, with inputs shown by grey boxes, data structures shown by yellow boxes, and algorithmic blocks shown by blue ovals. References to relevant sections are provided below the algorithmic blocks.

thesis lie within the place recognition block, described in Chapters 4 and 5, and Appendix D. While the image stream from the robot must be processed during run-time, the processing of the sample locations (feature detection, extraction, etc.) can be done ahead of time. Also note that rather than using single-image queries, query locations are expanded analogously to the way in which virtual locations are formed, as described in Section 3.3. This query expansion process provides more context and suppresses false positives, similar to concepts used in text and image retrieval [Chum et al., 2007], but can take advantage of covisibility to greatly simplify the expansion process.

The bulk of the computation lies in the place recognition block, since location retrieval can be done efficiently using the inverted index. However, the computation does not grow directly with the number of locations (like most other systems), since normalization does not require all locations in the map, and in general, only a small subset of areas in the covisibility map are retrieved as candidate virtual locations.

The procedural steps at each iteration of an online place recognition framework can be summarized as follows:

- (i) process each new image
  - (a) extract landmarks from the current image
  - (b) extract landmark descriptors and visual words for the current landmarks
  - (c) perform local matching between current landmarks and those from the previous time-step
  - (d) update landmark and covisibility statistics according to this local data association
- (ii) generate the current query graph via query expansion from the currently observable set of landmarks
- (iii) retrieve candidate locations from the covisibility graph
  - (a) retrieve list of relevant observation cliques (images) via the inverted index
  - (b) expand cliques into local clusters of landmarks to form virtual locations
- (iv) evaluate each candidate location with respect to the query
  - (a) compare each candidate location to the query using one of the proposed place recognition methods
  - (b) apply a threshold on the given posterior probabilities to get a list of matching locations
- (v) incorporate any resulting data associations
  - (a) perform matching between current landmarks and those from the matched location
  - (b) update landmark and covisibility statistics according to these new data associations
- (vi) repeat for each time-step

## A.2 Various Parameter Settings

While evaluating each system on several datasets, one important point is that the system parameters are able to produce results consistently across different environments. As a result, during each set of experiments, the only varying condition across the datasets is the set of sample locations that is used (this does not apply to SeqSLAM, as it does not use any samples). This is because, in each case, the samples include images from the other datasets, but images from the tested dataset are never included in the samples. However, since Naive-Bayes CovisMap, Graph CovisMap, and FAB-MAP require the use of sample locations, the same sample set is always used for all methods, with a different sample set for each dataset.

In the case of FAB-MAP, the implementation by the original developers [OxfordMRG, 2013] was used for testing, while in the case of SeqSLAM, a modified (in order to match the descriptions by Milford [2013]) version of OpenSeqSLAM was used [Sünderhauf, 2013]. For both FAB-MAP and SeqSLAM, parameters settings are given by the descriptions in [Cummins and Newman, 2008] and [Milford, 2013] respectively. Therefore in FAB-MAP, the detection probabilities are set as  $P(z_n|e_n) = 0.39$  and  $P(z_n|\bar{e}_n) = 0.0$ . However, in order to maintain consistency across different methods, no motion model or image masking was used. In addition, the naive-bayes implementation was used, rather than that which uses the Chow-Liu tree. The exception to these settings are for those experiments presented in Appendix D where both the motion model and Chow-Liu tree are used in FAB-MAP comparisons (further parameter settings are listed therein). As in [Cummins and Newman, 2008], FAB-MAP, Naive-Bayes CovisMap, and Graph CovisMap use U-SURF features.

For the SeqSLAM tests, the sequence length was set to 50 frames and the image resolution was always kept well above the documented threshold for performance degradation. In addition, the difference matrix was locally normalized using a radius of 20 frames, as documented. Finally, in order to be able to cope with the speed or frame-rate variations in the datasets, the slope for sub-route searches was varied between 0.25 and 4 for all tests.

Using the test results from Section 4.4, the detection probabilities for the Naive-Bayes CovisMap implementation were set to  $P(e_n|z_n) = 0.78$  and  $P(e_n|\bar{z}_n) = 0.32$ . For both Naive-Bayes CovisMap and Graph CovisMap, the covisibility parameter was set to 5%, and the percentage of observed words for candidate virtual location retrieval was set to 4% (although the system does not appear to be critically sensitive to these parameters).





# Appendix B

## Datasets

A total of six different datasets of image sequences labeled with ground-truth position estimates were used in order to analyse the behaviour of the systems. Table B.1 provides a summary of each of these datasets, with a few representative example images shown in Figure B.1.

The two Begbroke sequences listed in Table B.1 actually contain images from the same dataset, but using different subsampling of a high-framerate image stream. Both of these sequences are good for investigating place recognition, as the robot made three loops around a path, passing each location three times, therefore giving three instances of each place. These sequences consist of images from a forward-facing camera totalling approximately 1 km. Some challenges involved in the Begbroke dataset include a high repetition of scene elements (trees, bushes, grass, paved paths), blurred images in some locations, and ‘speed’ variations in the case of the Begbroke Multi-speed sequence which uses a different framerate for each loop.

The City Centre and New College datasets originate in [Cummins and Newman, 2008], and consist of two different parts of a university campus, each with fairly varied terrain (roads, gardens, paths). Both these datasets consist of images from two cameras mounted on a mobile robot, angled slightly to the left and right, and each traversing approximately 2 km. These two datasets are challenging due to many dynamic elements such as cars and pedestrians. In addition, these datasets are non-ideal for the covisibility framework presented in this thesis, because images were collected in an attempt to be independent from each other, providing relatively little connectivity between frames and therefore proving to be even more challenging.

The KITTI dataset is provided by Geiger et al. [2013], and consists of roughly 1.6 km of forward-facing images taken from the dashboard of a car. More specifically, it is the fifth

sequence from the odometry benchmark sequences. This particular sequence was chosen because it includes both interesting loop-closures and accurate ground truth coordinates. Some challenges involved with this dataset are dynamic objects, speed variations, and some relatively short loop-closure sequences.

Finally, the Ruelle dataset is provided by a handheld point-and-shoot camera, with images from a narrow alleyway. The street which is traversed is relatively short, but is traversed several times from different view-points and different speeds (namely image spacing). The difficulties in this dataset lie in the fact that loop-closure sequences are inconsistent in length and order, as well as view-point.

The first five datasets from Table B.1 are used to compare precision-recall characteristics for different place recognition frameworks in Section 4.8.1, while the last, shorter dataset is only used for illustrative purposes in Section 4.8.3.

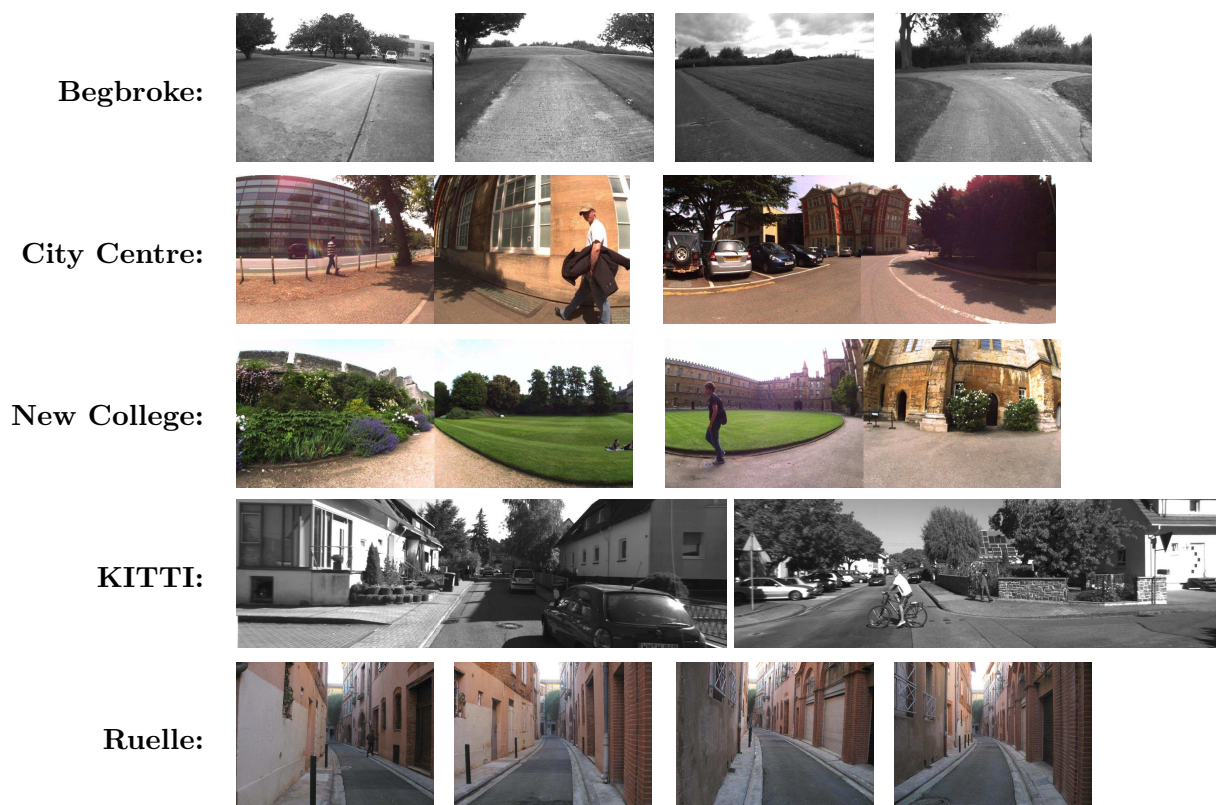


Figure B.1: Example images from each of the datasets described in Table B.1.

Dataset Name	Description	Sequence Length	Image Spacing	Image Specs
Begbroke	3 loops around a path surrounded by fields, trees, buildings, and cars.	approx. 1km, 1000 images	approx. 1m	forward facing, greyscale, $512 \times 384$ px
Begbroke Multi-speed	3 loops around a path surrounded by fields, trees, buildings, and cars. Each loop is given by a different framerate.	approx. 1km, 1000 images	approx. 0.5m, 1.0m, 2.0m	forward facing, greyscale, $512 \times 384$ px
City Centre	University campus with many buildings, cars, roads, gardens, and people.	approx. 2km, 1200 images	approx. 1.6m	left and right, colour, $640 \times 480$ px each
New College	University campus with many buildings, cars, roads, gardens, and people.	approx. 2km, 1200 images	approx. 1.6m	left and right, colour, $640 \times 480$ px each
KITTI	Urban dataset containing mostly roads, houses, trees, and cars.	approx. 2.3km, 1400 images	approx. 1.6m	forward facing, greyscale, $1226 \times 370$ px
Ruelle	Urban dataset of a narrow alleyway surrounded by houses. Several traverses are made with variations in the trajectory.	approx. 200m, 242 images	varied	forward facing, colour, $3264 \times 2448$ px

Table B.1: Overview of datasets used for testing.



## Appendix C

# Precision-Recall Metrics

Precision-recall characteristics are commonly used as a tool for system evaluation. Precision relates the number of correct matches to the number of false matches, whereas recall relates the number of correct matches to the number of missed matches. More formally, precision is the ratio of true positives over true positives plus false positives:

$$P = \frac{tp}{tp + fp}$$

And recall is the ratio of true positives over true positives plus false negatives:

$$R = \frac{tp}{tp + fn}$$

Curves can then be plotted, giving precision versus recall as the scores output by the algorithm are thresholded. A perfect system would return a result where both precision and recall have a value of one. When this is not achievable, the goal is to come as close to this as possible, possibly giving preference to maintaining certain levels of precision or recall, depending on the application.

For the tests in this work, a dataset of images is incrementally traversed, creating a map of locations over time and uses the most recent location as a query on the current map, with the goal of retrieving any previous instances of the location. A true positive is therefore defined as any returned location containing images that were taken within a certain radius of the query location (images are tagged with ground-truth position from GPS measurements). Similarly, a false positive is defined as a returned location which lies outside of the same radius, and a false negative is a location which lies within the radius but was not returned. When comparing different state-of-the-art methods, defining true and false matches in an

unbiased way can be difficult. For example, defining all locations within the given radius as positive would be unfair towards algorithms such as FAB-MAP [Cummins and Newman, 2008] since it can only return one image per query. As a result, the tests shown throughout this work use binary values of true positive, false positive, and false negative for each location. This additionally helps to avoid having to define the exact extent of a specific location which should be returned, as many cases can be ambiguous. The radius used for evaluation is set to 8 m unless otherwise stated. This choice is related to errors in the GPS ground truth data, which can reach up to several meters, and the fact that images can be spaced upwards of 2 m in some datasets. In this work systems strive for achieving as high a recall as possible while maintaining a precision value of 1.0. This preference towards perfect precision is due to the severe impact which false data-associations can have on maps.

## Appendix D

# Preliminary Version of CVPR16

## Paper on Graph Kernels

This final appendix includes work done during and after the preparation of the manuscript. The work has resulted in a paper appearing in CVPR 2016, and the following pages contain a preliminary version of the paper.



## 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, 10, 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



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.

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. 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.

The presented approach additionally 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 [11, 21, 33]. However, the recent trend is to rely on localizing within a prior map, or relying heavily on training and sample data, as in the works of [12, 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 [11, 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. 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], 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 pre-determined, 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 10000 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,

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

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,  $\bar{\mathcal{L}}_i$ :

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

resulting in the following equation for the posterior probability:

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

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,  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$ , 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,

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

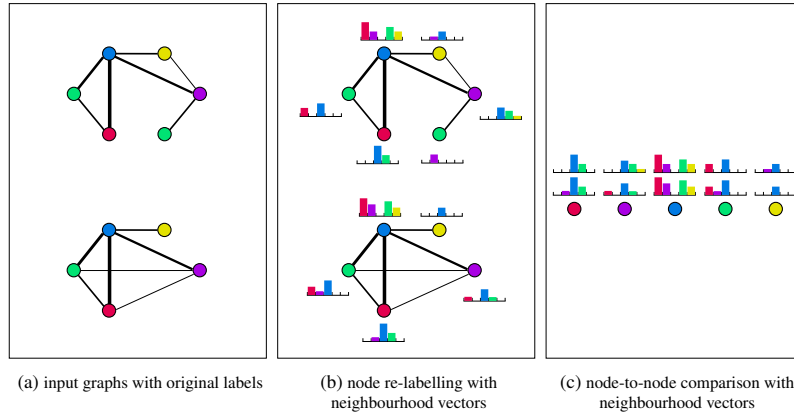


Figure 2: Illustration of the graph comparison process. The input graphs with node labels are shown, followed by the re-labelled graphs including each corresponding neighbourhood vector, and a node-to-node comparison of neighbourhood vectors from each graph. Colours in the node labels represent elements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

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 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 en-



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

sure 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.

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]. 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 [11], 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 to 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 KITTI 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 [11, 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 meth-

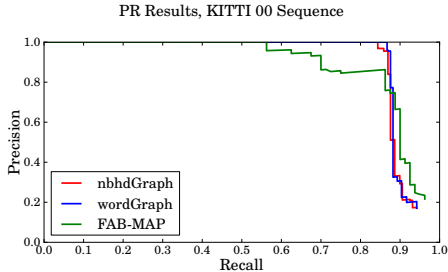


Figure 4: Precision-recall results on the KITTI 00 sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [11].

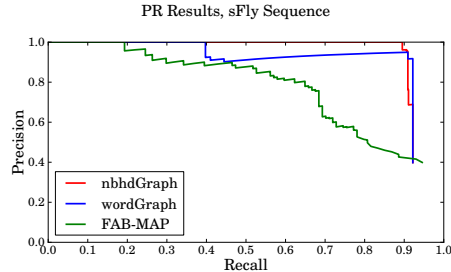


Figure 6: Precision-recall results on the sFly sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [11].

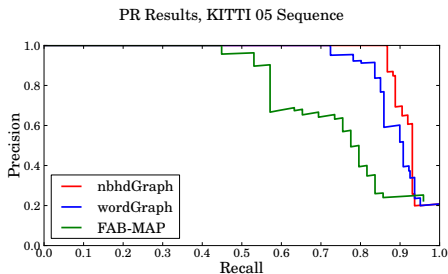


Figure 5: Precision-recall results on the KITTI 05 sequence for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [11].

ods, 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 implementation of nbhdGraph and wordGraph, the same covisibility clustering parameter of 0.05 was used [32]. The effective  $P(\mathcal{Z}_q|\mathcal{L}_i)$  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 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 [11] (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.

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

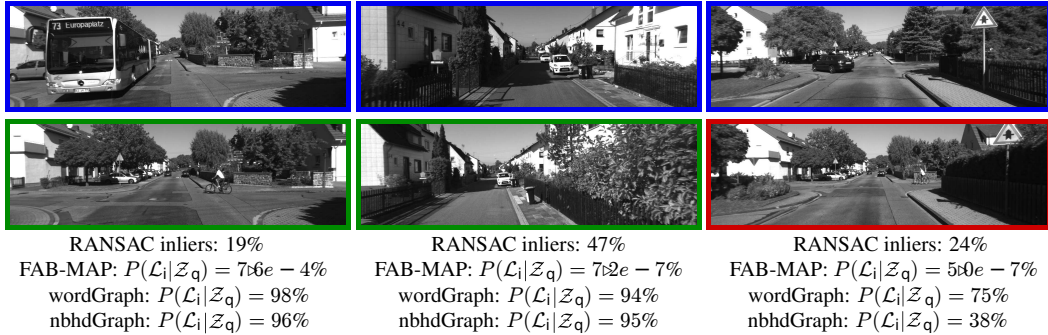


Figure 8: Example true and false-positive matches from the KITTI 05 dataset. Each column shows one example, where the query locations are shown in the top row in blue, with a candidate location below. True matches are designated in green, while false matches are designated in red. These examples represent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

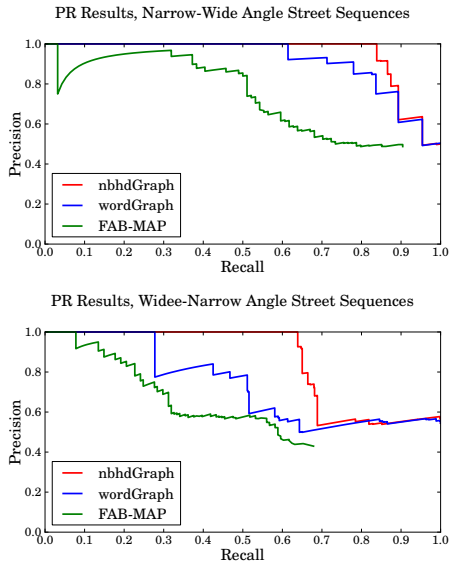


Figure 7: Precision-recall results on the Narrow-Wide and Wide-Narrow Angle sequences for the proposed method (nbhdGraph), the wordGraph method of [32] and the FAB-MAP framework of [11].

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.

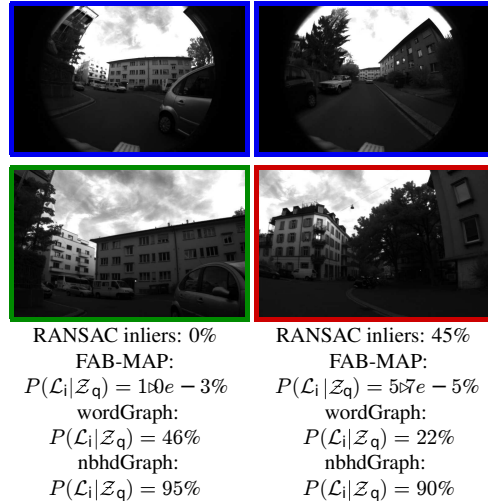


Figure 9: Example true and false-positive matches from the Narrow-Wide dataset. Each column shows one example, where the query locations are shown in the top row in blue, with a candidate location below. True matches are designated in green, while false matches are designated in red. These examples represent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

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

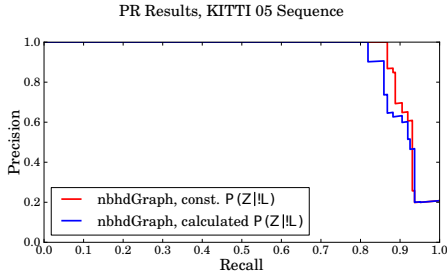


Figure 10: Precision-recall results on the KITTI 05 sequence, comparing the results using a constant value for  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$ , and one which calculated  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  using the dataset ground-truth.

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 [11]), 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  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  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,  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  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  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  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,  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  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  $P(\mathcal{Z}_q|\bar{\mathcal{L}}_i)$  can be ap-

proximated 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.

## 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), pages 2651–2652, 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)*, pages 510–517, 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)*, pages 25–32, New York, NY, United States, 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] M. Cummins. Probabilistic localization and mapping in appearance space. PhD thesis, University of Oxford, Balliol College, October 2009.
- [11] 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.
- [12] M. Dymczyk, S. Lynen, T. Cieslewski, M. Bosse, R. Siegwart, and P. Furgale. The gist of maps-summarizing experience for lifelong localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2767–2773, 2015.
- [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)*, pages 1–8, 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)*, pages 304–317. 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)*, pages 1169–1176, Miami Beach, FL, United States, 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. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3489–3496, 2011.
- [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)*, pages 90–97, Seattle, WA, United States, May 2015.
- [22] D. G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision (ICCV)*, pages 1150–1157, Corfu, Greece, 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)*, pages 3738–3744, Taipei, Taiwan, 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)*, pages 5487–5494, Seattle, WA, United States, 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)*, pages 2649–2656, 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)*, pages 3829–3836, 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)*, pages 1470–1477, Nice, France, 2003.
- [31] E. Stumm, C. Mei, and S. Lacroix. Building location models for visual place recognition. *The International Journal of Robotics Research (IJRR)*, April 2015.
- [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)*, pages 5475–5480, Seattle, WA, United States, 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)*, Karlsruhe, Germany, 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), pages 883–890, Portland, OR, United States, 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), pages 2922–2929, Portland, OR, United States, 2013.



## Appendix E

# French Summary of Manuscript

The following pages include a selective translation of parts of the manuscript into French.

THÈSE

Université de Toulouse,  
Université Paul Sabatier

and

LAAS-CNRS

Laboratoire d'analyse et d'architecture des systèmes

---

**Modèles probabilistes  
pour la reconnaissance  
visuelle de lieux**

*Auteur :*

Elena Stumm

*Directeurs :*

Simon Lacroix,  
Christopher Mei

*Equipe de recherche :*

Robotics & Interactions (RIS)

*Institut :*

LAAS-CNRS, Toulouse

*Rapporteurs :*

David Filliat,  
José Neira

*Jury :*

Ingmar Posner,  
Patrick Danès,  
Christopher Mei,  
Simon Lacroix

avril 2016

## Résumé

Cette thèse traite de la cartographie et de la reconnaissance de lieux par vision en robotique mobile. Les recherches menées visent à identifier comment les modèles de localisation peuvent être améliorés en enrichissant les représentations existantes afin de mieux exploiter l'information visuelle disponible. Les problèmes de la cartographie et de la reconnaissance visuelle de lieux présentent un certain nombre de défis : les solutions doivent notamment être robustes vis-à-vis des scènes similaires, des changements de points de vue de d'éclairage, de la dynamique de l'environnement, du bruit des données acquises. La définition de la manière de modéliser et de comparer les observations de lieux est donc un élément crucial de définition d'une solution opérationnelle. Cela passe par la spécification des caractéristiques des images à exploiter, par la définition de la notion de lieu, et par des algorithmes de comparaison des lieux.

Dans la littérature, les lieux visuels sont généralement définis par un ensemble ou une séquence d'observations, ce qui ne permet pas de bien traiter des problèmes de similarité de scènes ou de reconnaissance invariante aux déplacements. Dans nos travaux, le modèle d'un lieu exploite la structure d'une scène représentée par des graphes de covisibilité, qui capturent des relations géométriques approximatives entre les points caractéristiques observés. Grâce à cette représentation, un lieu est identifié et reconnu comme un sous-graphe.

La reconnaissance de lieux exploite un modèle génératif, dont la sensibilité par rapport aux similarités entre scènes, aux bruits d'observation et aux erreurs de cartographie est analysée. En particulier, les probabilités de reconnaissance sont estimées de manière rigoureuse, rendant la reconnaissance des lieux robuste, et ce pour une complexité algorithmique sous-linéaire en le nombre de lieux définis. Enfin les modèles de lieux basés sur des sacs de mots visuels sont étendus pour exploiter les informations structurelles fournies par le graphe de covisibilité, ce qui permet un meilleur compromis entre la qualité et la complexité du processus de reconnaissance.



# Table des matières

<b>1</b>	<b>Prémisse</b>	<b>1</b>
1.1	Motivations . . . . .	1
1.2	Vue d'ensemble . . . . .	3
1.3	Contributions . . . . .	7
1.4	Publications et présentations . . . . .	8
<b>3</b>	<b>Carte de co-visibilité (résumé)</b>	<b>11</b>
<b>4</b>	<b>Modélisation de lieux utilisant des sacs de mots (résumé)</b>	<b>13</b>
<b>5</b>	<b>Modélisation de lieux utilisant the graphes de mots (résumé)</b>	<b>17</b>
<b>6</b>	<b>Conclusion</b>	<b>21</b>
	<b>Bibliography</b>	<b>23</b>





# Chapitre 1

## Prémisse

### 1.1 Motivations

La navigation à long terme dans des environnements inconnus devient de plus en plus importante pour une grande variété de robots à plateforme mobile ainsi que diverses applications. Dans ce but, la plateforme de navigation doit être robuste vis-à-vis des erreurs, avec une localisation qui fonctionne même dans des cas non prévus, dynamiques et possiblement similaires. Un des plus importants des prérequis qui permettent d'obtenir une fiabilité de maintien de l'erreur de positionnement d'un robot durant la localisation et la cartographie simultanées (SLAM) est la reconnaissance de lieux visuel, pour réaliser une fermeture de boucle, à cause de la capacité du robot à naviguer dans une grande variété d'environnement [Cummins and Newman, 2011, Maddern et al., 2012]. Sans fermeture de boucle, la compréhension du monde par le robot diverge très rapidement de l'état réel à cause des petites erreurs accumulées au fil du temps, rendant la navigation impossible (la figure 1.1 présente un exemple du problème) [Cummins, 2009]. En plus de la fermeture de boucle, la reconnaissance de lieux peut être une fonction importante, comme tremplin, pour la cartographie sémantique et la compréhension de scène, ainsi que la fusion de carte ou la navigation multi-robots. Une reconnaissance de lieux incorrecte peut causer de grandes erreurs de cartographie en SLAM ou en fusion de carte, ainsi que des raisonnements complètement faux basés sur des interprétations incorrectes de scènes, ce qui souligne l'importance d'éviter de considérer les faux-positifs. Il en résulte que le but de notre tâche est de détecter le plus d'associations correctes de lieux, sans retourner de faux-positifs. La difficulté de ce problème réside dans la représentation de lieux d'une manière décorrélée de la trajectoire et des changements de l'environnement, mais suffisamment discriminante pour distinguer les caractéristiques répéti-

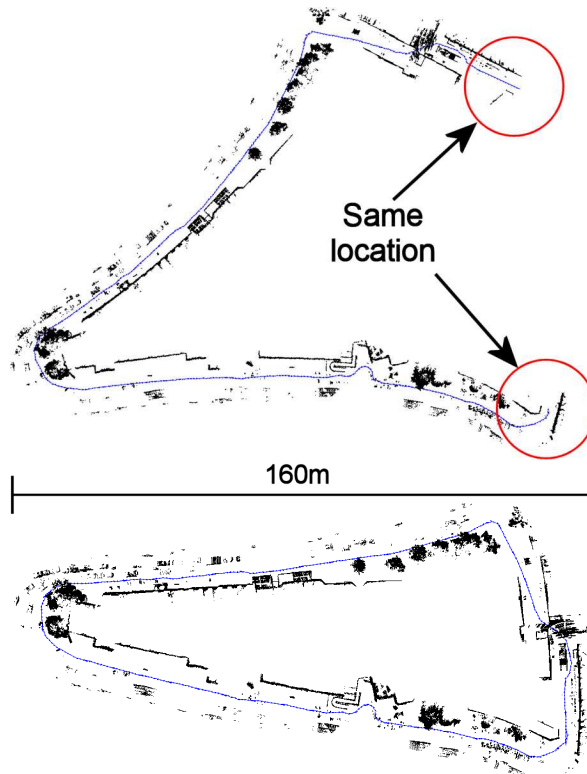


FIGURE 1.1: Plus les petites erreurs s'accumulent, plus la cartographie diverge de l'état réel et a besoin d'être corrigée grâce à une fermeture de boucle basée sur l'association de données.

tives, et permettre des inférences efficaces. Quelques exemples des types de difficulté visuelle qui peuvent arriver sont présentés dans les figures 1.2 et 1.3.

La problématique traitée dans cette thèse est d'évaluer si un robot mobile est en train de revisiter un lieu où il a déjà été ou s'il visite un nouveau lieu, ceci en analysant l'apparence visuelle de la scène courante. Au bout du compte, cette analyse doit être effectuée avec des hypothèses minimales concernant le comportement de l'environnement et du robot, dans le but de rester applicable dans le cas général. De plus, ce travail est basé uniquement sur des informations visuelles enregistrées par des caméras. Différentes manière de modéliser les observations visuelles pour la reconnaissance de lieux existents. Cependant, chaque model comporte ses propres simplifications et hypothèses, amenant divers avantages et inconvénients. Durant cette thèse nous avons étudié le comportement de divers models couramment utilisés, ainsi que certains, originaux, de notre conception. En faisant un choix prudent du modele pour représenter un lieu, les difficultés susmentionnées, lié à la prise de décision fiable concernant les associations de données dans le cas de scènes répétitives et similaire (appelé aliasing perceptuel), mais aussi de changement d'apparence due à la dynamique des éléments qui le composent, à la variation du point de vue et au changement de luminosité, sont simplifiées.

Dans ce travail, nous proposons un Framework bayésien qui utilise des modèles de locali-



FIGURE 1.2: Exemple de tâches d'identification difficiles du a des changements environnementaux.

sations basé sur des caractéristiques, construit sur la base de série d'éléments co-visible d'une scène pour évaluer la reconnaissance de lieux. Les modèles de localisation sont construits en groupant les caractéristiques basées sur la connectivité visuelle, permettant l'utilisation de contexte plutôt que d'images solitaire, tout en restant décorrélées des variations de trajectoires. De plus, l'utilisation de modèles basés sur les caractéristiques apporte une robustesse innée concernant les points de vue et les variations de luminosité grâce à l'utilisation de description invariante de caractéristiques locales. Intégrer de tel modèle de localisation dans un Framework bayésien permet de gérer, d'une manière intégrée et probabilistique, l'aliasing perceptuel, la dynamique des éléments et les incertitudes de détections.

La prochaine section propose une vue plus détaillée des concepts présents dans cette thèse ainsi que les différentes contributions.

## 1.2 Vue d'ensemble

Cette thèse porte principalement sur la reconnaissance de lieux visuel pour des robots mobiles, basé sur les concepts introduit par [Mei et al., 2010, Cummins and Newman, 2008] en établissant des modèles de localisation generatifs utilisant des cartes de co-visibilité. Cette section propose une brève vue d'ensemble de la structure de la thèse et des fondamentaux de la reconnaissance de lieux. Pour une meilleure compréhension du contexte et de la terminologie, veuillez vous référer au chapitre 2.

Les lieux sont représentés d'ordinaire par des repères visuels qui sont détectés dans la



FIGURE 1.3: Exemple de tâches d'identification difficiles du a des similitudes structurelle de l'environnement (aliasing perceptuel).

scène. Ces repères sont décrits grâce à des mots visuels, qui sont des versions quantifiées des descripteurs des caractéristiques originales, permettant de faciliter l'analyse et de réduire les besoins en mémoire ainsi que la complexité. Il en résulte que les lieux peuvent être retrouvé et comparé en utilisant les techniques du domaine de la récupération de documents texte [Sivic and Zisserman, 2003, Manning et al., 2008, Cummins and Newman, 2008, Angeli et al., 2008b, Botterill et al., 2011]. Par conséquent, des modèles probabilistes relativement simples, mais valables, peuvent être créés et utilisés pour réussir à réaliser une reconnaissance de lieux efficace, comme dans le travail de Cummins and Newman [2008]. L'idée est la suivante : en modélisant chaque lieu par une liste de mots visuels, la représentation reste invariante, jusqu'à un certain degré, aux changements de luminosité et de points de vue, tout en incorporant, grâce au modèle probabiliste, les notions de bruits des observations et de la dynamique des éléments. Cependant, les travaux précédents ont du mal à définir l'étendue d'un lieu, ils se basent généralement sur une discrétisation arbitraire de l'espace, soit défini par une seule image, soit par une séquence d'image à longueur prédéfinie. De plus, ces modèles n'incorporent pas les informations 3D concernant les repères du lieu à cause de la difficulté d'utilisation des informations additionnelles de manière efficace. Les références citées fonctionnent à l'aide de modèles sac-de-mots qui ne prennent pas en considération les relations structurelles entre les repères.

Dans ce travail, nous insistons sur l'importance de ces relations structurelles à plusieurs niveaux. Premièrement, ces relations sont utiles pour définir l'étendue d'un lieu donné, et

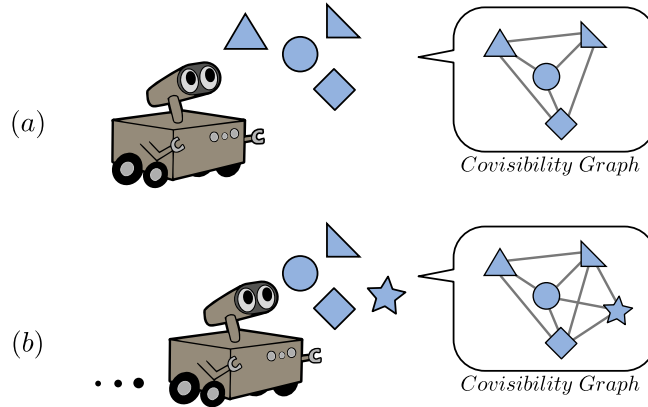


FIGURE 1.4: En se déplaçant, le robot fait des observations, détecte des points de repère et note dans une structure de graphe ceux qui ont été vus ensemble. Vous pouvez voir ici un exemple simple de deux étapes d'un robot qui se déplace vers l'avant dans l'environnement ainsi que le graphe de covisibilité résultant.

peuvent être utilisées pour retrouver des lieux [Stumm et al., 2013]. Deuxièmement, ce sont des sources importantes d'informations discriminantes, qui aident à diminuer l'aliasing perceptuel et les faux-positifs [Stumm et al., 2015b]. Nous soutenons que ces relations peuvent être intégrées de manière simple et intuitive, ce qui permet d'obtenir une représentation de l'environnement plus proche de la réalité et plus continue. Ceci est réalisé en utilisant des cartes de co-visibilité qui sont construites pendant que le robot explore son environnement, en notant quel sont les repères observés en même temps dans un graphe structure [Mei et al., 2010]. Les concepts de la cartographie de base sont décrits dans la figure 1.4. Le graphe de co-visibilité est capable de capturer de manière implicite la structure sous-jacente de l'environnement sans avoir à modéliser toutes les informations de positionnement 3D ; comme les repères appartenant à la même structure seront co-observés plus souvent que ceux appartenant à des structures différentes. À partir de la carte de co-visibilité, les « lieux virtuels » pertinents, qui ressemblent à une requête donnée, peuvent alors être retrouvés en tant que clusters [Stumm et al., 2015a]. Les lieux sont maintenant définis en utilisant des propriétés directes de l'environnement (repères), et deviennent moins dépendants des variations de trajectoires tout en éliminant le problème de la sélection d'une position.

Une fois les lieux virtuels retrouvés, un framework probabiliste est utilisé pour identifier toute correspondance possible entre la requête et les lieux observés préalablement. Le développement d'un modèle génératif propre est un facteur clé afin de produire des résultats utilisables, spécialement dans les environnements difficiles, ce qui en fait l'objectif principal de cette thèse. Par exemple, une méthode probabiliste rigoureuse permet des seuils de confiance inhérents et peut gérer des situations problématiques telles que l'aliasing perceptuel en

comprenant la probabilité des éléments de la scène. Ceci se base sur un traitement prudent de la normalisation de la probabilité, fait en utilisant un ensemble d'échantillon de lieux qui sont utilisés pour modéliser le monde (inconnu). La méthode présentée dans cette thèse permet au système de chercher toutes les correspondances de lieux, plutôt que la plus probable, donnant la possibilité de faire face aux cartes erronées qui peuvent contenir plus d'une instance du même lieu (par exemple, quand les fermetures de boucles sont manqué). Le modèle développé améliore aussi la stabilité en ce qui concerne les paramètres de sélections, par rapport aux travaux précédent. La probabilité postérieure résultante représente une mesure intuitive de similarité de lieu, avec des valeurs variant doucement lors d'un passage dans un lieu requis. Cette thèse développe différentes manière de calculer les probabilités d'observation pour les lieux, utilisant des montants variables d'information structurelle et spatiale provenant du graphe de co-visibilité tout en examinant les résultats ainsi obtenu.

Le manuscrit est divisé en cinq chapitre : le chapitre 2 donne une vue global du contexte lié au recherche mené durant cette thèse, le chapitre 3 introduit la notion de carte de co-visibilité et comment les lieux sont retrouvé en tant que graph de repère co-visible. Les chapitre 4 et 5 analyse et développe plusieurs modèles génératif et probabiliste pour la reconnaissance de place visuel, d'abords en utilisant une représentation sacs-de-mots puis en utilisant des représentations basé graphe. Finalement, le chapitre 6 fournit une perspective de l'impact de ce travail et présente différentes piste pour les travaux futurs. De plus, comme les développements théoriques de cette thèse sont largement soutenus par des évaluations expérimentales, plusieurs résultats de tests seront présentés tout au long de la thèse pour plus de clarté. Dans le but de comprendre l'implémentation et la procédure de test, les annexes présentes les choix d'implémentation (Annexe A), les ensembles de données utilisées (Annexe B) et un rappel de la mesure de la précision (Annexe C).

## 1.3 Contributions

Le travail accompli durant cette thèse a apporté les contributions suivantes au domaine :

- présentation et analyse d'un framework unifié pour définir, retrouver et reconnaître des lieux de manière robuste
- des modèles génératif amélioré pour une reconnaissance de lieux basés sur l'apparence, incluant :
  - une stabilité améliorée en ce qui concerne les réglages des paramètres des modèles d'observation visuel
  - un nouveau schéma de normalisation permettant des lieux redondant dans une carte et une performance sous linéaire en ce qui concerne les lieux d'une carte
  - l'introduction de technique de comparaison efficace et structuré pour les graphes de localisation, diminuant les effets de l'aliasing perceptuel
- une évaluation et une analyse approfondie du comportement et des performances caractéristiques comparées à l'état de l'art



## 1.4 Publications et présentations

- E. Stumm, C. Mei, S. Lacroix, “Recognizing Places using Covisibility Maps,” in IEEE International Workshop on Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM), 2013
- E. Stumm, C. Mei, and S. Lacroix, “Probabilistic place recognition with covisibility maps,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 2013. [Stumm et al., 2013]
- E. Stumm, C. Mei, and S. Lacroix, “Building location models for visual place recognition,” in The International Journal of Robotics Research (IJRR), 2015. [Stumm et al., 2015a]
- E. Stumm, C. Mei, S. Lacroix, and M. Chli, “Location Graphs for Visual Place Recognition,” IEEE International Conference on Robotics and Automation (ICRA), Seattle, USA, 2015. [Stumm et al., 2015b]
- E. Stumm, C. Mei, S. Lacroix, M. Hutter, J. Nieto and R. Siegwart, “Robust Visual Place Recognition with Graph Kernels,” Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA, 2016. [Stumm et al., 2016]





## Chapitre 3

# Carte de co-visibilité (résumé)

Pour une requête de localisation donnée (qui correspond généralement à la position actuelle du robot), l'idée est que le système puisse évaluer si et où un même emplacement a été vu auparavant. Les principales approches développées dans cette thèse se fondent sur des descriptions de localisation comprenant des ensembles de mots visuels [Sivic and Zisserman, 2003], ce qui permet de comparer efficacement la requête avec un ensemble d'emplacements candidats extraits de la carte actuelle. Ce chapitre décrit la façon dont l'environnement est représenté par un graphe de ces caractéristiques visuelles, où la covisibilité définit la connectivité [Mei et al., 2010] et sur lequel des modèles de localisation probabiliste sont ensuite construits (ces modèles probabilistes seront décrits en détails dans les chapitres 4 et 5). Des mots visuels quantifiés sont utilisés pour décrire les caractéristiques fournies par chaque point de repère (caractéristiques visuelles distinctes dans l'image). La carte est ensuite construite comme un graphe de covisibilité non orienté, dont les noeuds sont ces points de repère et dont les arrêtes représentent le fait que deux repères connectés ont été vus ensemble. Au moment de la requête, on peut rechercher dans le graphe des groupes de points de repère qui partagent une forte similitude avec la requête tout en extrayant des sous-graphes qui représentent des candidats d'emplacements virtuels pour une analyse plus approfondie. Ces lieux virtuels s'adaptent dynamiquement à la scène de la requête, et sont intrinsèquement moins dépendants à la trajectoire d'observation du robot. La section 3.2 explique comment le graphe de covisibilité est construit et maintenu au fil du temps, tandis que la section 3.3 explique comment récupérer les lieux virtuels pertinents.



## Chapitre 4

# Modélisation de lieux utilisant des sacs de mots (résumé)

Ce chapitre vise à développer un modèle probabiliste d'observation des lieux, afin d'évaluer si l'un d'un ensemble des lieux de candidats correspondant à un lieu de la requête. Les grappes de mots visuels détectés qui ont été introduites dans la section 3.3 représentent les observations de lieux, et voici, une approche probabiliste sac-de-mots est utilisé pour comparer la requête à l'ensemble des lieux virtuels candidats.

La probabilité d'un emplacement générer l'observation de requête donnée peut être trouvée en utilisant le théorème de Bayes :

$$P(\mathcal{L}_i|\mathcal{Z}_q) = \frac{P(\mathcal{Z}_q|\mathcal{L}_i)P(\mathcal{L}_i)}{P(\mathcal{Z}_q)} \quad (4.1)$$

$\mathcal{L}_i$  est un lieu virtuel particulier, et  $\mathcal{Z}_q$  est l'observation de la requête donnée par un ensemble de mots visuels  $\{z_1, z_2, \dots, z_N\}$ . Le développement de chaque terme de l'équation 4.1 est donnée tout au long de ce chapitre. Section 4.2 explique comment l'existence et l'observation des éléments visuels de la scène sont modélisés, suivie par l'introduction d'un modèle nouveau pour l'observation probabilité donnée d'un lieu à la section 4.3, une discussion des techniques de normalisation dans la section 4.5, un aperçu de l'échantillonnage procédure à la section 4.7, et une description de la façon dont le lieu périeurs sont estimés à la section 4.6.

Les contributions de ce chapitre sont les suivantes : Un regard attentif dans le modèle de probabilité d'observation a conduit à un nouveau paramétrage et caractéristiques améliorées par rapport à la sensibilité des paramètres. En outre, une reformulation des résultats

des modèles de normalisation plus lisses et plus intuitives probabilités a posteriori, la complexité de normalisation sous-linéaire par rapport à la taille de la carte, ainsi que l'ajout de la capacité à gérer plusieurs fermetures de boucles simultanées qui peuvent survenir en raison de licenciés lieux dans les cartes existantes. Cette méthode de normalisation repose sur un ensemble de lieux d'échantillonnage de modéliser l'environnement inexplorée, qui a été montré pour être efficace dans nos expériences, ainsi que des expériences de travail connexe, comme Cummins and Newman [2011]. Cependant, l'applicabilité et la création de cet échantillon ensemble demeure pas bien compris. Une certaine connaissance préalable concernant les types d'environnements attendus est nécessaire pour créer efficacement et sélectionner la mesure d'un ensemble de l'échantillon, que le calcul évolue avec le nombre de points d'échantillonnage. Dans le même temps, une telle connaissance préalable est réalisable dans la plupart des scénarios, et de séries d'échantillons ont été observés à bien performer à travers une variété d'ensembles de données dans la pratique. En outre, des recherches préliminaires prometteurs a été fait sur l'utilisation de plusieurs paramètres du modèle d'observation différentes afin de faire face à une variété de sources des lieux d'échantillonnage et les types et la qualité des images qu'ils sont représentés par. En plus de ces études dans des modèles probabilistes, le cadre repose sur la structure sous-jacente de covisibilité proposé par Mei et al. [2010], qui a également été examiné ici. Ce chapitre a démontré l'influence et avantages de l'utilisation du graphe de covisibilité dans la définition et l'extraction des lieux, en augmentant la répétabilité et le contexte relatif associé à un lieu par rapport aux modèles de localisation d'une image unique et l'augmentation de l'invariance de trajectoire par rapport aux ensembles d'images séquentielles.

image from third loop: (current position)	image from third loop: (previous position)	image from second loop:	image from first loop:	image from a different location
covismap --	covismap 1.0	covismap 0.998	covismap 0.991	covismap 1e-05
fabmap --	fabmap 0.865	fabmap 6e-07	fabmap 1e-05	fabmap 1e-05
seqslam --	seqslam 0.819	seqslam 0.207	seqslam 0.186	seqslam 0.098

FIGURE 4.13: Exemple d'un lieu de la séquence Begbroke qui est passé à plusieurs reprises, et les scores de concordance résultant d'une requête générée au cours de la troisième passe. Les scores sont présentés pour le système décrit dans cette thèse (CovisMap), FAB-MAP, et SeqSLAM. Le lieu de la requête est affiché sur la gauche en gris, suivie par l'image juste avant le lieu de recherche, une image de la passe précédente, une image de la première passe, et un exemple négatif de lieu.



FIGURE 4.15: Compte tenu de la requête en vert, lieux virtuels les plus probables de trois parcours distincts d'une rue sont présentés (avec seulement quatre images affichées par lieu pour plus de clarté). La requête et la seconde passe sont traversés sur le côté gauche de la rue, tandis que les premier et troisième passages sont traversés sur le côté droit. En outre, les images recueillies à partir de la demande de recherche ont été générées à partir d'un déplacement bien plus lent et erratique (y compris les retours en arrière), résultant en plus d'images représentant la même traverse encore ne pose aucun problème pour le système.





## Chapitre 5

# Modélisation de lieux utilisant the graphes de mots (résumé)

Comme vu dans le chapitre précédent, la difficulté majeure de la technique de reconnaissance basée sur l'apparence est de conserver la robustesse de l'aliasing perceptif. Les approches discutées diffèrent principalement entre les représentations relatives aux caractéristiques (comme la comparaison entre des ensembles de points d'images) et les représentations relatives aux images globales (comme la comparaison entre l'intensité des pixels dans une image). Les comparaisons de l'intégrité des caractéristiques peuvent être coûteux en ressources informatiques, and c'est pourquoi la plupart des caractéristiques sous-jacentes liées à la structure et la géométrie sont généralement ignorées, tel que dans le cadre du FAB-MAP [Cummins and Newman, 2008] qui, par ce fait, diminue la spécificité des caractères distinctifs des méthodes discutées précédemment dans les chapitre 4. Par conséquent, telles méthodes sont sujette à générer des associations de données à faux résultat positif ou à des biais de rappel qui sont causés par un manque de observations discriminatoires. Dans la

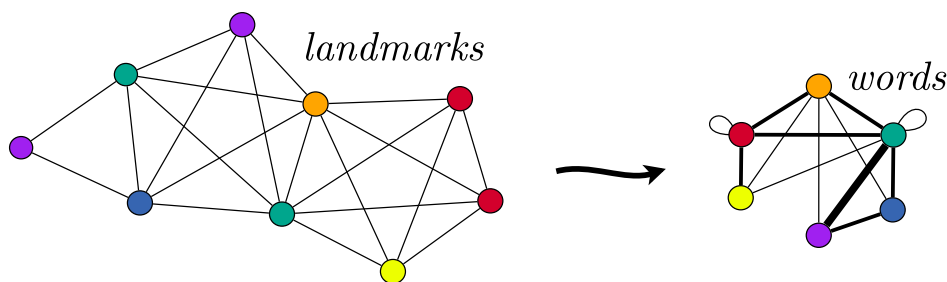


FIGURE 5.3: Un lieu : à partir d'un graphe de visibilité de repère dans un graphe pondéré de mots, tout en conservant des informations structurales tout en louant la comparaison

méthode du chapitre 4, une réduction des faux positifs par l'aliasing perceptif est accompli grâce à la modélisation des positions inconnues des échantillons en les comprenant dans une normalisation de la probabilité postérieure  $P(\mathcal{L}|\mathcal{Z})$ . De plus, beaucoup de systèmes basés sur les caractéristiques retiennent l'information sur les caractéristiques des positions et se fondent sur une étape additionnelle de hoc post-processing appliqué à toutes les positions correspondantes possibles par une vérification géométrique de la géométrie épistolaire. D'autre part, les méthodes telles que SeqSLAM [Milford, 2013] qui utilisent les représentations d'image globale, manquent d'invariance et se fondent sur de longues séquences d'images afin d'éviter l'aliasing perceptif.

Ce chapitre examine la comparaison structurée des lieux représentés en tant que graphes contenant des repères visuels de la carte de covisibilité du chapitre 3, avec considérant le postulat disant que l'arrangement relatif aux caractéristiques joue potentiellement un rôle fondamental dans la distinction des lieux. Cela contraste avec le modèles de sacs de mots présentés précédemment, dans lesquels l'interférence est faite en utilisant des ensembles non-structurés de mots visuels, bien que les lieux sont recherchés en tant que graphe basé sur la carte de covisibilité. La raison de cette simplification est de faciliter la tâche de modélisation autant que le calcul. Dans ce chapitre, nous avons pour but d'explorer les gains potentiels en considérant l'intégralité de la structure du graphe pour chaque lieu.

La théorie des graphes est actuellement un domaine de recherche actif, avec de nombreuses applications allant de la bioinformatique, de réseauter théorie, à la vision par ordinateur. Nous attendons donc dans l'état de l'art dans les méthodes graphique correspondant, et comment ils peuvent être appliqués à la reconnaissance visuelle de lieu. En général, les solutions au problème d'appariement de graphes ont récemment devenu possible sur de grands graphiques (plus de 100 nœuds), et encore rester difficile sur les graphes denses (contenant de nombreux bords), des graphiques avec plus de plusieurs centaines de nœuds, ou graphiques avec grande jeux d'étiquettes. En conséquence, certaines méthodes discutées ici peuvent ne pas être en mesure de fonctionner à une complexité raisonnable pour l'utilisation dans les tâches de navigation en ligne encore, mais nous croyons qu'ils demeurent des points de discussion intéressants pour l'application potentielle future. Les principales contributions de ce chapitre sont aperçus sur les noyaux de graphes pour la place la reconnaissance visuelle, ainsi que d'une proposition et l'évaluation d'une approche de travail avec une représentation graphique simplifiée pour les comparaisons efficaces de localisation. Les résultats expérimentaux démontrent la validité des améliorations de la démarche et de démontrer plus de l'état de l'art.

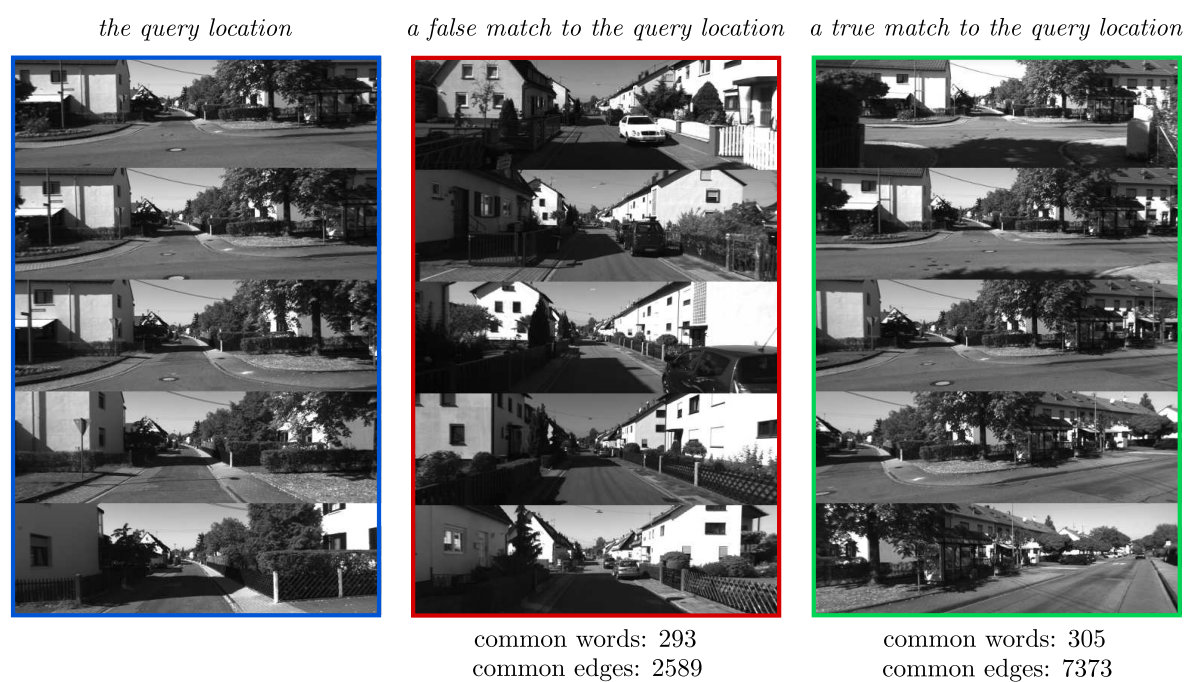


FIGURE 5.7: Une requête et deux lieux de l'ensemble de données Kitti candidats, représenté par cinq images représentatives. Les deux sites candidats démontrent l'importance de l'information structurelle dans l'observation, que les deux sites partagent environ 300 mots visuels avec le lieu de la requête, les véritables actions de match plus de deux fois plus nombreux bords avec le lieu de la requête.



## Chapitre 6

# Conclusion

Cette thèse a porté sur la reconnaissance de lieu basé sur leur apparence ; en considérant tout d'abord divers représentations de localisations et la probabilité correspondante des modèles basé sur l'observation visuelle. Afin de regroupé des repères de manière cohérente, un graphe de covisibilité est créé, et partitionné dans le graphe de representation des emplacements. Cette méthode s'est avérée être capable de s'adapté de manière inhérente aux variations de trajectoires du robot, comprenant des changements irrégulié de vitesse, direction, et point de vue. Une analyse détaillée des modèles probabilistiques d'observation a été utilisé pour améliorer la robustesse aux erreurs et la sensibilité aux paramètres. Le modèle génératif qui en résulte fourni la probabilité à posteriori d'être dans un certain endroit étant donné une observation particulière, d'une manière ne nécessitant pas la normalisation de la carte toute entière et est capable de trouver plusieurs instances d'un même endroit. L'infrastructure logicielle est utile pour la détection de fermeture de boucle, le rétablissement succédant le problème du robot kidnappé, la fusion de carte et la correspondance topologique. C'est pourquoi la méthodologie est appropriée pour les applications où le robot se déplace dans un environnement non contraint, ou lors de l'utilisation de modes non conventionnels de déplacement, tel que les ascenseurs ou les trains, où repérer l'egomotion devient très difficile (équivalent au problème du robot kidnappé). De plus, afin d'exploiter pleinement la representation par graphe de covisibilite des lieux, des techniques efficaces de ont ete exploree pour comparer les lieux. Notamment, cette these developpe un indice de probabilite d'observation qui exploite le covisibilites des reperes pour prendre en compte les structures geometrique en plus de l'apparence, ila ete demontre que cette methode surpasse les methodes de l'etat de l'art dans la reconnaissance de lieux sur plusieurs collectes de données. Alors que les données encodées sont relativement faible en terme d'information géométrique et structurale dans le

graphe de covisibilité, cela apparaît être suffisant pour différencier les correspondances basées uniquement sur l'apparence, ce qui aide à résoudre l'aliasing perceptuel, qui est un problème commun dans les méthodes existantes. Une évaluation réalisée sur des environnements variés et avec un niveau de bruit variable montre une augmentation du rappel à une précision parfaite. L'ajout de complexité lors de l'incorporation d'indices géométriques est minimisé par l'usage d'algorithmes efficaces inspirés par à la fois la correspondance de graphe et la reconnaissance de lieux issue de la littérature.







## Acknowledgements

I would like to thank the people and organizations that have made this work possible. This includes my local and distant supervisors, especially Christopher and Simon, and also Margarita and Marco. I am grateful to have gotten financial support from France, as well as LAAS, Université Paul Sabatier, Ecole Doctorale Systèmes, and the Swiss National Science Foundation for allowing me to travel and collaborate with other labs. Also, thank you to all my reviewers, for taking the time to give me feedback and improve my work. Finally, I would like to thank all my friends, family, colleagues, turkeys, and moose for all the help and adventures.





# Bibliography

- Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Building Rome in a day. *Communications of the Association for Computing Machinery (ACM)*, 54(10):105–112, 2011.
- Alexandre Alahi, Raphael Ortiz, and Pierre Vanderghenst. FREAK: Fast retina keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 510–517, 2012.
- Adrien Angeli, Stéphane Doncieux, J.-A. Meyer, and David Filliat. Incremental vision-based topological SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1031–1036, September 2008a.
- Adrien Angeli, David Filliat, Stéphane Docieux, and Jean-Arcady Meyer. A fast and incremental method for loop-closure detection using bags of visual words. *IEEE Transactions on Robotics (T-RO), Special Issue on Visual SLAM*, 24(5):1027–1037, October 2008b.
- Adrien Angeli, Stéphane Doncieux, J-A Meyer, and David Filliat. Visual topological SLAM and global localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4300–4305, 2009.
- Roberto Arroyo, Pablo F. Alcantarilla, Luis M. Bergasa, and Eduardo Romera. Towards life-long visual localization using an efficient matching of binary sequences from images. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6328–6335, Seattle, WA, United States, May 2015.
- Artem Babenko and Victor Lempitsky. The inverted multi-index. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(6):1247–1260, June 2015.
- Francis Bach. Graph kernels between point clouds. In *International Conference on Machine Learning (ICML)*, pages 25–32, New York, NY, United States, 2008.

- Lu Bai, Luca Rossi, Horst Bunke, and Edwin R Hancock. Attributed graph kernels using the Jensen-Tsallis  $q$ -differences. In *Machine Learning and Knowledge Discovery in Databases*, pages 99–114. Springer, 2014.
- Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics Automation Magazine*, 13(3):108–117, September 2006.
- Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.
- Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science, New York, NY, United States, 2007.
- Joydeep Biswas and Manuela Veloso. WiFi localization and navigation for autonomous indoor mobile robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4379–4384, 2010.
- Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Depth kernel descriptors for object recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 821–826, 2011.
- Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Unsupervised feature learning for rgb-d based object recognition. In *Experimental Robotics*, pages 387–402. Springer, 2013.
- Karsten M. Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In *IEEE International Conference on Data Mining (ICDM)*, pages 74–81, 2005.
- Karsten Michael Borgwardt. *Graph Kernels*. PhD thesis, Ludwig-Maximilians-Universität München, July 2007.
- Tom Botterill, Steven Mills, and Richard Green. Bag-of-words-driven, single-camera simultaneous localization and mapping. *Journal of Field Robotics (JFR)*, 28(2):204–226, March/April 2011.
- Marcus A. Brubaker, Andreas Geiger, and Raquel Urtasun. Lost! leveraging the crowd for probabilistic visual self-localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3057–3064, Portland, OR, United States, 2013.
- Horst Bunke and Kaspar Riesen. Towards the unification of structural and statistical pattern recognition. *Pattern Recognition Letters*, 33(7):811–825, 2012.

- César Cadena, Dorian Gálvez-López, Juan D. Tardós, and José Neira. Robust place recognition with stereo sequences. *IEEE Transactions on Robotics (T-RO)*, 28(4):871–885, August 2012.
- Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary robust independent elementary features. In *European Conference on Computer Vision (ECCV)*, pages 778–792. Springer, 2010.
- Song Cao and Noah Snavely. Graph-based discriminative learning for location recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 700–707, Portland, OR, United States, June 2013.
- Zetao Chen, Obadiah Lam, Adam Jacobson, and Michael Milford. Convolutional neural network-based place recognition. In *ARAA Australasian Conference on Robotics and Automation (ACRA)*, December 2014.
- Margarita Chli and Andrew J. Davison. Automatically and efficiently inferring the hierarchical structure of visual maps. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 387–394, Kobe, Japan, May 2009.
- Siddharth Choudhary and P.J. Narayanan. Visibility probability structure from sfm datasets and applications. In *European Conference on Computer Vision (ECCV)*, pages 130–143. Springer, 2012.
- Ondřej Chum, James Philbin, Josef Sivic, Michael Isard, and Andrew Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *International Conference on Computer Vision (ICCV)*, pages 1–8, Rio de Janeiro, Brazil, 2007.
- Winston Churchill and Paul Newman. Experience-based navigation for long-term localization. *The International Journal of Robotics Research (IJRR)*, 32(14):1645–1661, December 2013.
- Mark Cummins. *Probabilistic localization and mapping in appearance space*. PhD thesis, University of Oxford, Balliol College, October 2009.
- Mark Cummins and Paul Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research (IJRR)*, 27(6): 647–665, June 2008.

- Mark Cummins and Paul 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.
- Frank Dellaert and Michael Kaess. Square root SAM: Simultaneous localization and mapping via square root information smoothing. *The International Journal of Robotics Research (IJRR)*, 25(12):1181–1203, 2006.
- Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes Paris look like Paris? *Association for Computing Machinery (ACM) Transactions on Graphics*, 31(4), 2012.
- Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping (SLAM): part I. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.
- Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision (ECCV)*, pages 834–849. Springer, 2014.
- Susana Eyheramendy, David D. Lewis, and David Madigan. On the naive bayes model for text categorization. In *International Workshop on Artificial Intelligence and Statistics*, Key West, FL, United States, 2003.
- Matthew Fisher, Manolis Savva, and Patrick Hanrahan. Characterizing structural relationships in scenes using graph kernels. *Association for Computing Machinery (ACM) Transactions on Graphics*, 30(4):34, 2011.
- Dorian Gálvez-López and Juan D. Tardós. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics (T-RO)*, 28(5):1188–1197, October 2012.
- Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research (IJRR)*, 32(11):1231–1237, 2013.
- Arren Glover, William Maddern, Michael Warren, Stephanie Reid, Michael Milford, and Gordon Wyeth. OpenFABMAP: An open source toolbox for appearance-based loop closure detection. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4730–4735, St. Paul, MN, United States, 2012.

- Zäid Harchaoui and Francis Bach. Image classification with segmentation graph kernels. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.
- Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, volume 15, page 50. Manchester, United Kingdom, 1988.
- Djoerd Hiemstra. A probabilistic justification for using  $\text{tf} \times \text{idf}$  term weighting in information retrieval. *International Journal on Digital Libraries*, 3(2):131–139, 2000.
- Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision (ECCV)*, pages 304–317. Springer, 2008.
- Herve Jegou, Matthijs Douze, and Cordelia Schmid. On the burstiness of visual elements. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1169–1176, Miami Beach, FL, United States, June 2009.
- Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Improving bag-of-features for large scale image search. *International Journal of Computer Vision (IJCV)*, 87(3):316–336, 2010.
- Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(1):117–128, 2011.
- Edward Johns and Guang-Zhong Yang. Feature-co-occurrences maps: Appearance-based localisation throughout the day. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3212–3218, Karlsruhe, Germany, 2013.
- Edward Johns and Guang-Zhong Yang. Generative methods for long-term place recognition in dynamic scenes. *International Journal of Computer Vision (IJCV)*, 106(3):297–314, 2014.
- Arijit Khan, Yinghui Wu, Charu C Aggarwal, and Xifeng Yan. Nema: Fast graph search with label similarity. *Proceedings of the Very Large Databases (VLDB) Endowment*, 6(3):181–192, 2013.
- Kasra Khosoussi, Shoudong Huang, and Gamini Dissanayake. Novel insights into the impact of graph structure on SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2707–2714, September 2014.



- Kurt Konolige and Motilal Agrawal. FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics (T-RO)*, 24(5):1066–1077, 2008.
- Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3607–3613, 2011.
- Yasir Latif, César Cadena, and José Neira. Robust loop closing over time for pose graph SLAM. *The International Journal of Robotics Research (IJRR)*, 32(14):1611–1626, 2013.
- Stefan Leutenegger, Margarita Chli, and Roland Yves Siegwart. BRISK: Binary robust invariant scalable keypoints. In *International Conference on Computer Vision (ICCV)*, pages 2548–2555, 2011.
- Chris Linegar, Winston Churchill, and Paul Newman. Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 90–97, Seattle, WA, United States, May 2015.
- Ming Liu and Roland Siegwart. Topological mapping and scene recognition with lightweight color descriptors for an omnidirectional camera. *IEEE Transactions on Robotics (T-RO)*, 30(2):310–324, 2014.
- David G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision (ICCV)*, pages 1150–1157, Corfu, Greece, 1999.
- Kirk MacTavish and Timothy D. Barfoot. Towards hierarchical place recognition for long-term autonomy. In *Workshop on Visual Place Recognition in Changing Environments, at IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, 2014.
- Will Maddern, Michael J. Milford, and Gordon F. Wyeth. CAT-SLAM: probabilistic localisation and mapping using a continuous appearance-based trajectory. *The International Journal of Robotics Research (IJRR)*, 31(4):429–451, April 2012.
- Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, United States, 2008.
- Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.

- Christopher Mei, Gabe Sibley, and Paul Newman. Closing loops without places. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3738–3744, Taipei, Taiwan, 2010.
- Krystian Mikolajczyk and Cordelia Schmid. Scale & affine invariant interest point detectors. *International Journal of Computer Vision (IJCV)*, 60(1):63–86, 2004.
- Michael J. Milford. Vision-based place recognition: how low can you go? *The International Journal of Robotics Research (IJRR)*, 32(7):766–789, June 2013.
- Michael J. Milford and Gordon F. 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, St. Paul, MN, United States, 2012.
- Mahesh Mohan, Dorian Gálvez-López, Claire Monteleoni, and Gabe Sibley. Environment selection and hierarchical place recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5487–5494, Seattle, WA, United States, 2015.
- Michael Montemerlo, Sebastian Thrun, Daphne Koller, and Bernard Wegbreit. Fast-SLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1151–1156, 2003.
- Ana C. Murillo and Jana Kosecka. Experiments in place recognition using gist panoramas. In *IEEE International Conference on Computer Vision, Workshops (ICCV Workshops)*, pages 2196–2203, 2009.
- José Neira, Juan D. Tardós, and José A. Castellanos. Linear time vehicle relocation in slam. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 427–433, 2003.
- Peer Neubert, Niko Sünderhauf, and Peter Protzel. Superpixel-based appearance change prediction for long-term navigation across seasons. *Robotics and Autonomous Systems*, 69: 15–27, 2015.
- Paul Newman, Gabe Sibley, Mike Smith, Mark Cummins, Alastair Harrison, Chris Mei, Ingmar Posner, Robbie Shade, Derik Schroeter, Liz Murphy, et al. Navigating, recognizing and describing urban spaces with vision and lasers. *The International Journal of Robotics Research (IJRR)*, 28(11-12):1406–1433, 2009.

- David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2161–2168, 2006.
- Aude Oliva and Antonio Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in Brain Research*, 155:23–36, 2006.
- Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1717–1724, 2014.
- Oxford Mobile Robotics Group - OxfordMRG. Open source FabMap 2.0 code, 2013. URL <http://www.robots.ox.ac.uk/~mjc/Software.htm>. [Online; accessed 2013].
- Rohan Paul and Paul Newman. Fab-map 3d: Topological mapping with spatial and visual appearance. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2649–2656, 2010.
- Rohan Paul and Paul Newman. Self-help: seeking out perplexing images for ever improving topological mapping. *The International Journal of Robotics Research (IJRR)*, 32(14):1742–1766, December 2013.
- Edward Pepperell, Peter Corke, and Michael Milford. Towards persistent visual navigation using SMART. In *ARAA Australasian Conference on Robotics and Automation (ACRA)*, 2013.
- James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.
- James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.
- Andrzej Pronobis, Barbara Caputo, Patric Jensfelt, and Henrik I. Christensen. A discriminative approach to robust visual place recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3829–3836, 2006.
- Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 413–420, June 2009.

- Ananth Ranganathan and Frank Dellaert. Bayesian surprise and landmark detection. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2017–2023, Kobe, Japan, 2009.
- Ananth Ranganathan and Frank Dellaert. Online probabilistic topological mapping. *The International Journal of Robotics Research (IJRR)*, 30(6):755–771, May 2011.
- Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *European Conference on Computer Vision (ECCV)*, pages 752–765. Springer, 2012.
- Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, August 2007.
- Karl-Michael Schneider. On word frequency information and negative evidence in naive Bayes text classification. In *Advances in Natural Language Processing*, volume 3230, pages 474–486. Springer, 2004.
- Nino Shervashidze, Tobias Petri, Kurt Mehlhorn, Karsten M. Borgwardt, and S. Vichy N. Vishwanathan. Efficient graphlet kernels for large graph comparison. In *International Conference on Artificial Intelligence and Statistics*, pages 488–495, 2009.
- Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. Weisfeiler-Lehman graph kernels. *The Journal of Machine Learning Research*, 12:2539–2561, 2011.
- Roland Siegwart, Illah Reza Nourbakhsh, and Davide Scaramuzza. *Introduction to Autonomous Mobile Robots*. MIT Press, 2nd edition, 2011.
- Saurabh Singh, Abhinav Gupta, and Alexei Efros. Unsupervised discovery of mid-level discriminative patches. *European Conference on Computer Vision (ECCV)*, pages 73–86, 2012.
- Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, pages 1470–1477, Nice, France, 2003.
- Randall Smith, Matthew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous Robot Vehicles*, pages 167–193. Springer New York, 1990.

- Hauke Strasdat, Andrew J. Davison, J.M. Martínez Montiel, and Kurt Konolige. Double window optimisation for constant time visual SLAM. In *International Conference on Computer Vision (ICCV)*, pages 2352–2359, 2011.
- Elena Stumm, Christopher Mei, and Simon Lacroix. Probabilistic place recognition with co-visibility maps. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4158–4163, Tokyo, Japan, 2013.
- Elena Stumm, Christopher Mei, and Simon Lacroix. Building location models for visual place recognition. *The International Journal of Robotics Research (IJRR)*, April 2015a.
- Elena Stumm, Christopher Mei, Simon Lacroix, and Margarita Chli. Location graphs for visual place recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5475–5480, Seattle, WA, United States, 2015b.
- Elena Stumm, Christopher Mei, Simon Lacroix, Marco Hutter, Juan Nieto, and Roland Siegwart. Robust visual place recognition with graph kernels. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, United States, 2016.
- Niko Sünderhauf. OpenSeqSLAM open source code, 2013. URL <https://openslam.org/openslam.html>. [Online; accessed 2013].
- Niko Sünderhauf and Peter Protzel. BRIEF-Gist – closing the loop by simple means. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1234–1241, 2011.
- Niko Sünderhauf and Peter Protzel. Switchable constraints for robust pose graph SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1879–1884, Oct 2012.
- Niko Sünderhauf, Peer Neubert, and Peter 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)*, Karlsruhe, Germany, 2013.
- Niko Sünderhauf, Feras Dayoub, Sean McMahon, Ben Talbot, Ruth Schulz, Peter Corke, Gordon Wyeth, Ben Upcroft, and Michael Milford. Place categorization and semantic mapping on a mobile robot. *under review*, 2015a.

- Niko Sünderhauf, Sareh Shirazi, Adam Jacobson, Feras Dayoub, Edward Pepperell, Ben Upcroft, and Michael Milford. Place recognition with ConvNet landmarks: Viewpoint-robust, condition-robust, training-free. *Proceedings of Robotics: Science and Systems*, 2015b.
- Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision (IJCV)*, 7(1):11–32, 1991.
- Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT Press, 2005.
- Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 883–890, Portland, OR, United States, June 2013.
- Antonio Torralba, Kevin P Murphy, William T Freeman, and Mark A Rubin. Context-based vision system for place and object recognition. In *International Conference on Computer Vision (ICCV)*, pages 273–280, 2003.
- Iwan Ulrich and Illah Nourbakhsh. Appearance-based place recognition for topological localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1023–1029, 2000.
- S. Vichy N. Vishwanathan, Nicol N. Schraudolph, Risi Kondor, and Karsten M. Borgwardt. Graph kernels. *The Journal of Machine Learning Research*, 11:1201–1242, 2010.
- Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.
- Stephan Weiss, Markus W. Achtelik, Simon Lynen, Michael C. Achtelik, Laurent Kneip, Margarita Chli, and Roland Siegwart. Monocular vision for long-term micro aerial vehicle state estimation: A compendium. *Journal of Field Robotics (JFR)*, 30(5):803–831, 2013.
- Brian Williams, Mark Cummins, José Neira, Paul Newman, Ian Reid, and Juan Tardós. A comparison of loop closing techniques in monocular slam. *Robotics and Autonomous Systems*, 57(12):1188–1197, 2009.
- Sukjune Yoon, Soonyong Park, Sung Hwan Ahn, Hyoseok Hwang, and Kyung Shik Roh. Robust place recognition by spectral graph matching using omni-directional images. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1527–1532, July 2014.

Hong Zhang, Bo Li, and Dan Yang. Keyframe detection for appearance-based visual SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2071–2076, 2010.

Feng Zhou and Fernando De la Torre. Deformable graph matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2922–2929, Portland, OR, United States, 2013.

C. Lawrence Zitnick and Krishnan Ramnath. Edge foci interest points. In *International Conference on Computer Vision (ICCV)*, pages 359–366, 2011.